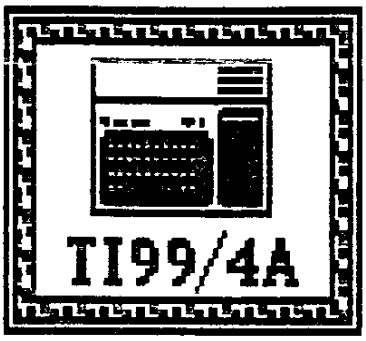
TI-D-BITS

JANUARY

1988



A NEW TEAM

The Philadelphia Area TI-99/4A Users' Group meets twice a month. On the first Saturday of any given month, we meet at the Bucks County Youth Development Center, (YDC, which is next to Neshaminy Mall), Administration Building, beginning at 10:00 am. On the third Saturday of each month, we meet at LaSalle University, 20th Olney, in room H-329 located in the Science Building. Membership to The Philadelphia Area TI-99/4A Users' Group is available to all. We invite anyone that is interested in the TI-99/4A to visit us. Stop in and see what is available to you for your TI and how membership can benefit you!

Current executive board consists of:

PRESIDENT.....	Don Arsenault.....	215-368-0446
VICE PRESIDENT.....	Allan Silversteen.	215-885-7910
SECRETARY.....	Mark Wannop.....	609-365-1776
TREASURER.....	Tom D'Annunzio....	215-947-7353

Committees consists of:

TI-d-BITS	Ralph Field.....	215-362-2534
	Don Arsenault.....	215-368-0446
	Bill Hughes	
	Rice Hall	
LIBRARY	George Kulp.....	215-489-2418
	Newton Stallman	
	Charles Campbell	
MEMBERSHIP ...	Bill Hughes	
	Bob Lundin	
ASSISTANT TREASURER.	Frank Passini	
EDUCATION	Barry Traver	
	Bud Shapiro	
	Carlo Angelico	
EQUIPMENT	Rice Hall	

REMEMBER to be considerate when calling any of the above people. Limit your calls to the early evening hours. (6pm to 9pm)

The opinions expressed herein are those of the individual authors are not necessarily those of the Philadelphia Area TI-99/4A Users' Group or its officers. Nor is the Philadelphia Area TI-99/4A Users' Group or any of its officers responsible for any damage, inconvenience, or loss which may result as a consequence of the use of any written material herein.

TI-d-Bits is published monthly by the Philadelphia Area TI-99/4A Users' Group, P.O. Box 18486, Philadelphia, PA 19120. All material herein may be reprinted freely by other non-profit User Groups, (unless otherwise stated), as long as proper credit is given to both source and author. Contributions are encouraged, but no payment is made. Editorial, advertising, and classified, copy MUST be in by the LAST day of the previous month. You can either mail your copy to: TI-d-Bits, c/o The Philadelphia Area TI-99/4A Users' Group, P.O.Box 18486, Philadelphia, PA 19120 or send it via modem by contacting Don Arsenault at (215)-368-0446. If your piece contains any diagrams, charts, or code, send a paper copy AT FINAL PUBLICATION SIZE.

Classified ads are printed in blocks. A block consists of 3 lines, 55 characters wide, or any increment of 3 lines. Classified advertising is accepted from members at NO CHARGE for a one block ad, per issue. Additional ads from members may be placed at cost of $1.00 per block. Non-members may place classified ads at a cost of $2.00 per block. All advertisements MUST be paid for in advance.

Commercial advertising is accepted for publication at the following rates:

Quarter page	$ 5.00
Half page	$ 8.00
Full page	$15.00

Commercial advertisements will be placed in the next available issue. All advertisements MUST be paid for in advance.

The editor of TI-d-Bits or the executive board of The Philadelphia area TI-99/4a Users' Group reserve the right to reject any material submitted for publication for any reasons.

The Philadelphia Area TI-99/4A Users' Group's program library is available to all active members at NO CHARGE. A catalog of the library's contents is given to all new members upon request and updates will appear in this publication from time to time. To obtain material from the library, contact the librarian for the best procedure to obtain your requests.

THE NEXT LA SALLE MEETING IS JANUARY 16 1988

SECRETARY'S NOTES

By P. Mark Mannop

HAPPY NEW YEAR!!!

Another year has passed by, and we can all look forward to new and better things for ourselves, our families, and our machines! The Geneve 9640 looks like it is making itself at home in the TI-99/4A community, plus there are more and more items coming out for all of us. (There's some new things coming from Mechatronics which I'll note a little later, as well as some other nifty new items.)

As usual, the first parts of this column are basically the same as the PACS DATA BUS column (except for the correction of one glaring grammatical error that wasn't caught until Too Late...), and the "new" stuff begins with NOTES FROM HERE AND THERE... So if you've read the 'BUS column, jump on down...

DECEMBER MEETING NOTES

Attendance was lighter than usual, about 30 members, due to the closeness of the holidays; still, we had a varied and interesting meeting.

The TI SIG is now the proud owner of a Geneve 9640 computer from Myarc. This is the current-production 640K machine (TI-99/4A compatible) that many users are upgrading to. With purchase of the 9640, we can serve the needs of owners of that machine while continuing our support of the original 99/4A.

Bud Shapiro was named an honorary member of the SIG; Bud was an active member of our education committee, giving classes in Extended BASIC as well as running the "one-liner" and "five-liner" contests. We all wish Bud well, and hope he drops in when he can to see us.

PE CARD FOR SPEECH SYNTHESIZER

Dr. Eric Bray demonstrated a new product from Rave 99 (the firm which also makes an enhanced keyboard for the TI), that allows the user to place the circuit board from the TI Speech Synthesizer inside the PE Box, instead of daisy chained on the peripheral port. This daisy-chain connection can be a possible cause of intermittancy on the bus between the computer and the PE Box; this card eliminates that problem.

While similar to Corcomp's Triple-Tech card, the Mechatronics offering lacks the printer spooler and clock circuit of the Corcomp unit; The Mechatronics card is, of course, less expensive. This is of note for the Geneve 9640 owner, as that computer already has printer spooler capacity as well as a clock; buying the Corcomp unit to gain speech capability would be a tad redundant.
(the 9640 has no provision for the normal connection of the Synthesizer, although it does support it electronically.) The Mechatronic unit fills the need for a simpler unit.

Eric demonstrated several programs (in 99/4A mode) that utilize speech; the unit worked well. Use of the card requires the user to remove the Speech Synthesizer circuit board from its case, and plug it into the PE board. This is an easy operation that can be accomplished by anyone who can handle a screwdriver.

A CHRISTMAS ANIMATED CARTOON

Barry Traver demonstrated a real Christmas treat for us all, an animated cartoon that runs on both the 99/4A and the 9640. It features Woodstock, Snoopy's little bird friend from the "Peanuts" strip. After a bit of music while the opening screens are shown, Woodstock flies in to his nest in the tree next to Snoopy's doghouse. After he has "successfully" landed, he notices the present Snoopy has left him on top of the doghouse. He tries to get it to his nest, but is hampered by Killer Snowflakes. He finally gets it there, and opens it to find... but that would be telling.

The program, written by Ray Kasmir, is quite amusing and features some nice programming tricks. When Woodstock chirps, a "translation" appears in a window at the bottom of the screen. While this is common on Apples (being very easy to do on that machine), this is not often found in TI programs. Another feature of the program is the instantly appearing graphics screen; usually these screens are "constructed" as the veiwer looks on. This program is worth having, not only as an amusing demonstration, but as an interesting program to "disect" and pick up some valuable programming techiques.

A VISIT FROM STEVE LONGO

In response to requests from the SIG, Steve Longo came to talk to us about the grumblings reported in previous columns. Dr. Longo explained the rationale for the room change, explaining how he arrives at the room assignations. The main point of contention centers around the elevator, which requires a key to operate; Dr. Longo expressed sympathy for the situation, and explained that the University is reluctant to distribute keys due to security reasons; Dr. Longo said he would try to investigate the matter further.

Dr. Longo mentioned that he has proposed a "Small SIG Division" and a "Large SIG Division" within PACS; this would allow the small SIGS to get together to express

their needs to the PACS organization. Dr. Longo also mentioned that PACS would welcome volunteers to help out with PACS functions, such as the upcoming Computer Festival, and the Trenton Computer Flea Market (which PACS helps organize).

Also discussed was the PACS BBS; the BBS will have new phone numbers shortly (check the PACS BBS column) that do not go through the LaSalle switchboard. This will hopefully cut down the line noise that many users have had to contend with.

We were glad to have Dr. Longo at the meeting to answer our questions, and thank him for his visit.

SPACE STATION PHETA

John Calvin Traver, assisted by his father Barry, demonstrated a unique game called Space Station Pheta. The game is similar to many "hop and jump" style games, but has some nice twists to it. The object of the game is to run, hop, and climb through a maze, collecting keys in order to open the door. This gets you to the next screen, where you again collect keys, and so on. There are several odd features of these mazes, however, including certain sections that will spring you to the top of the screen, or cause you to disappear from one point and re-appear in another.

Although the program comes with 79 screens, it also comes with an easy-to-use editor allowing the user to create his own screens. Existing screens can be made easier or harder to the user's taste and ability, and entirely new screens may be created. Game components can be placed and moved at will with the editor, and there are many nifty "devices" that can be used, such as "temporary floors" and "invisible ladders".

On the 79 screens provided with the program, the computer will display the proper movement solution upon request. That way, if the player gets stuck on one particular screen, he can see the solution and go on with the game.

This is a commercial program, and is available from TI Software, 109 Tee Circle, Salem VA 24153. Phone number is (703) 387-2581 - call at reasonable hours...

NOMINATIONS AND ELECTION

Barry Traver, as spokesman for the Nominating Committee, reported on the Committee's recomendations for SIG officers for 1988. Citing the tendency for our SIG Presidents to "burn out" from overwork (Barry mentioned his own case as an example), the committee suggested that Don Arsenault be considered for the Presidential position. The committee suggested that the other positions be retained by those filling them, if they so wish, with the exception of the Vice-President, who is selected by the President-Elect. Don indicated that he would be happy if Allan Silversteen would continue as Vice-President. All 1987 officers indicated they would be willing to continue in their positions. Finishing the Nominating Committee report, nominations were solicited from the floor; there being none, it was moved to accept the recommendations of the Committee, and to place those named in office. The 1988 SIG officers are:

President - Don Arsenault
Vice President - Allan Silversteen
Treasurer - Tom D'Annunzio
Secretary - Mark Wannop

The current SIG officers will try to make the meetings and services of the SIG as varied, helpful, and relevant to the members as we can. We would also like to thank Ted Cheney for his leadership during the past year. Ted indicated he would be available for help with the SIG, and would be interested in helping members with Multiplan.

MUSIC PRE-PROCESSOR CHRISTMAS DISK

Norm Sellers, programmer of the Music Pre-Processor, showed off his disk of Christmas music to be used with that system, and demonstrated various features of the Music Pre-Processor program. The program creates a system to key in music easily, without having to program innumerable CALL SOUND statements. The information read by the program is placed in REM statements, which the computer reads (!) and converts into music. The measure being read is displayed on the screen; if this is not wanted (it wasn't used on the Christmas disk, as that featured a graphic display) one can program a D OFF statement; D ON turns the measure counter on again. Norm's interest is with Barbershop Quartets, and he wanted four-part harmony in his program; the TI sound chip provides only three tone channels (and one "noise" channel). To solve the problem, Norm has one of the sound channels warble between the desired two harmony parts, to get four voices out of three. Keychanges can be introduced anywhere in the music.

LONG-DISTANCE ON THE PACS BBS REVISITED

As I went on about in the last column, the PACS BBS can be used to communicate with computer users all over the country, if you know the various "nodes" to go through to reach the networking BBS near that person. I used Mike Riccio as an example, as I had a node address that would reach him in Pittsburg. Mike advised the group that he has found a shorter address than the one published; to send Mike a message through the PACS BBS private (M)ail, enter this at the "who is the message for" prompt (all in lower case):

teav>'%pa!rutgers!andrew.cmu.edu!ar2u

This will cut a couple hours off the transmission time from the last version of this address, as it bounces about the countryside a tad less. (For example, it has a couple less nodes, and goes to Rutgers rather than Harvard.)

It would be nice if we could come up with a directory to reach TI people around the country; that way we could communicate with each other for the cost of a local call. This would be a boon to those of us who cannot afford the commercial services like Compuserve!

NOTES FROM HERE AND THERE

NEW ITEMS FROM MECHATRONICS

The following was gleaned from an "advertising feature" of International TI-LINES (Oxford, England); The International TI User Group is acting as a sales agent to help British TI'ers find peripherals. The "advertising feature" makes for interesting reading, as some of the items haven't yet made an appearance in the States (or not much of one...).

MECHATRONICS 80 COLUMN STANDALONE CARD

This card is a "choo choo train" peripheral because it has to be - there is insufficient space within the console, and cannot be made for the PE Box due to timing and signal strength considerations.

Mack McCormick (familiar to many of us from the Seminar he gave at PATIUG a couple of years ago, as well as his columns in MICROpendium) rated the card thusly: Ease of use: A; Documentation: B; Performance: A; Value: A; OVERALL RATING: A+. The following is quoted directly from Mack's review as printed in TI-LINES.

"The Mechatronic 80 Column Card is the final answer for the TI-99/4A. In fact, if you already have a RAVE 99 keyboard and a GRAM KRACKER or GRAM Card, you already have the equivalent of the new Myarc 9640 micro. [Almost - You still wouldn't be able to access the large amount of user RAM that the 9640 can. - P.M.W.] My system running this configuration and the 9640 O/S (which executes perfectly from an Editor/Assembler cartridge with RAM at >6000) was only about 5% slower then the 9640. This is because much of the 9640's speed derives from the 9938 VDP chip which is also at the heart of the Mechatronic 80 column card."

"**Ease of use:** The card is transparent except when running standard BASIC when a micro switch on the card must be pressed so the O/S can find the extra 128K of VDP RAM provided. Installation requires the removal of the old TI VDP chip from within your console. Usually the VDP chip is socketed, so work with a soldering iron is unnecessary. Mechatronics provides pictorial step by step instructions. Total installation time is about ten minutes."

"To fully benefit from this card you really need an RGB monitor, since a standard TV may not be able to cope with the high resolution graphics." [That should be an ANALOG RGB, not a TTL RGB, monitor. The TTL RGB cannot show more than 16 colors... You'd want to use the SAME monitors recommended for use with the Geneve 9640 - P.M.W.]

"**Documentation:** The manual is short but adequate. All DIP switches are fully documented and the RGB output connections are clear. The manual goes a good job of documenting how you get 80 column output from within your current TI programs by printing to the card, which is treated like a peripheral."

"The company provides 80 column versions of TI-Writer and MultiPlan with each purchase" [Hopefully, this will hold true in the States... - P.M.W.]

"**Performance:** The card performs exactly as expected, is well behaved, and I have never had it crash a program. With the extra 128K RAM you can run the 9938 chip in all modes including super high resolution bit map mode. Converting Extended BASIC programs to give displays in 80 columns is a snap. You simply print to the card using a special print sequence which also specifies colours and other necessary information. From TI-Writer you can also see your print in 80 columns on screen before printing it."

"The improvement in screen resolution and being able to have the TI work in 80 column mode is fantastic. This is the last piece of hardware you need to upgrade your machine to the equivalent of many other newer systems on the market."

"**Overall rating:** It is rare for me to assign an overall grade of A+ to a product, but this one is well deserved. An important side benefit is that software developed for the Myarc 9640 should be compatible with this card. For my money the 80 column card from Mechatronic cannot be beaten."

"The 9938 VDP specifications are extensive and too complex to provide full detail here; it ranges from 40 to 80 column modes with 512 colours to 512 by 424 pixels with 16 colours. The total palette of colours ranges between 1024 and 4096, depending on the information source."

Me again... Well, offhand I'd say Mack McCormick likes the unit... The drawing shows it to be about the size of an old TI standalone ("choo-choo") unit, but a tad slimmer. The price given is 235 British pounds - I have no idea what the U.S. cost might be, or if a NTSC version has been yet released. As Mack notes, this unit will bring the 99/4A graphically equivalent to the 9640; whether it is cost-effective to do so would depend on it's price. After all, the cost of a keyboard expansion, Gram Card (or Kracker, if you can find one) and the 80 Column board might add up to the cost of the 9640; and you still couldn't address the extra memory the 9640 can. On the other hand, there seems to be no reason why this couldn't work with the CorComp peripheral system or TI "Choo-choo" units; those without P-E Boxes might want to look into this unit.

MECHATRONIC 128K RAMDISK WITH CENTRONICS PRINTER INTERFACE

Another standalone, having data line passing through it so it may be used with other standalones or the P-E Box. It comprises 32K standard user RAM (configured by the 99/4A O/S into the usual 8k for machine code and 24K for BASIC program area); 96K RAM configured as a RAMDISK (a tad larger than the equivalent SSSD disk); and a Centronics Printer Interface.

The full 128 can be bank switched in 32K banks by your own software (provided that software resides outside the 128K space - for example, in Mini-Memory) and there is a 24 page booklet covering the unit.

The RAMDISK O/S will currently only permit a maximum of 3 files to reside within the 96K space; the O/S may be altered in the future to be directly compatible with TI disk O/S format (127 files).

The Centronics port is not the same as TI's PIO (which is parallel, but doesn't quite conform to Centronics standard) and avoids name conflict by being called PLOT. The unit can be used along with an RS232/PIO card without conflict.

Again, I have no U.S. price or availability. The British price is 105 pounds.

DEPTFORD DOINGS...

I saw a few neat things at the Deptford TI meeting of interest to all... COMMAND DOS was demonstrated; this is loaded in via a Gram Kracker or SuperCart and provides many nifty new commands and allows for batch processing. It will run on the 99/4A and the 9640, and supports 80 columns on the 9640 or 99/4A with and 80 colum board (see above...). I went on about this last time, so I won't go into more detail this time.

Another neat little program shown was TI SINGS. This one is a lot of fun; using allophones, it causes the TI Speech Synthesizer (with the TE-II cartridge in place) to actually sing! Admittedly, the TI's synthesizer certainly sounded funny doing its rendition of the Beatle's "I Saw Her Standing There", and did provoke some laughter, but it ACTUALLY did SING it - all the notes were there! There are other songs provided, such as "Bicycle Built For Two" ("Daisy, Daisy..."); the TI sounds like HAL did in 2001... This program is commercial, but is extremely reasonable, going for about $5.95 from Tenex. A further data disk with many more songs is available for around eleven bucks; you can also program your own songs.

If there is any drawback to the TI-SINGS program, it is that the Sound chip is not provided for; the TI sings acapella... Of course one could program TWO TIs and.... Accompanied or not, this program is a howl; I plan on ordering my copy soon, having uttered the computer hobbyist's war cry: "I Gotta Getta Copyya THAT!!!"

MORE TI-IBM STUFF - THIS TIME FROM GENIAL!

MICROpendium notes various TI to MS-DOS text file conversion programs, including PC PEP (which required both a TI and MS-DOS computer linked together) and the CorComp cartridge (which ONLY works with the CorComp controller).

A new PC Transfer program, written by Mike Dodd, will work with either a Myarc or CorComp controller and sells for only $25.00 from the New England branch of Genial Computerware, Box 183, Grafton MA 01519. (but we can always ask Barry at the meeting...) The program allows formatting compatibility for IBM also. If I understand all this correctly, this give those TI'ers with Myarc or CorComp controllers some measure of compatibility with IBM text files, which will be of interest to those who use MS-DOS machines at work.

TIME TO SHUT MYSELF UP...

Well, I just did a ShowDirectory and found that I've actually written more than I did last time (which I thought was impossible...); enough of my burble! Hope you all had happy holidays, and best wishes for the coming year!... **Hang In There!**

T.I. 99/4A DISK PERIPHERAL OVERVIEW (Continued)

By COLIN HINSON
INT. TI-LINES, 09/86

So far in the previous parts of this article, we have dealt with the floppy disk layout and the methods of accessing the disk itself.

When accessing the disk from a high level (i.e. using file access as opposed to sector access), current information about the disk and file(s) etc are held in VDP RAM so as to avoid having to re-access the relevant sectors on the disk each time a sector is read from a file. In addition to this data space is also required to buffer the data being read from the disk. The memory used for this is at the high end of VDP RAM and is permanently allocated by the power up routine within the DSR ROM of the disk controller, and as a consequence cannot be used by application programs, although its size can be changed by a (GPL) utility routine.

The allocated VDP memory is roughly sub-divided into three categories:

1. Drive Control information
2. File allocation information
3. Data buffering.

Each of these is discussed below.

DRIVE CONTROL INFORMATION

In order to control the Disk hardware, the software needs to know the current status of each drive before it can access it. All this information is readily available (some through checking the actual current status of the drive directly).

The power up routines take care of the FD1771 register initialization.

FILE ALLOCATION INFORMATION

File allocation information is held in the File Control Blocks (FCBs), each open file having an FCB associated with it.

The information contained in the FCB is identical to the File Descriptor Record (FDR) data held on the disk for each file, with the addition of 6 bytes of dynamic information about each file. These six bytes are stored in front of the FDR information (i.e. the FDR starts at FDB byte 6). As the length of the FDR is 256 bytes and there is a data buffer for each file of 256 bytes, the total length of an FDB is therefore 256 + 256 + 6 = 518 bytes.

The format of the 6 byte extension of the FDR is:

Bytes -6 and -5 = Current logical Record offset on Level 2
Bytes -4 and -3 = Physical Record location of the FDR
Byte -2 = Logical Record offset
Byte -1 = Drive ID

The meanings of these entries are as follows:

Drive ID - Contains the Drive number on which the associated file resides. If the highest bit of this byte is set, then the current data block has been modified and it will have to be written back to the drive before the file is closed or a new data block is accessed.

Logical Record offset - Contains the offset of the next logical record in the current physical record. If during a **READ** operation this points to a byte entry of >FF then this indicates an end of record for the current physical record.

Note that this entry is used only for variable length records. For fixed length records, the actual AU and the position within that AU is computed before each I/O operation, and therefore the logical record offset byte is irrelevant.

During **WRITE** operations, this offset points to the first free byte in the physical record. If the next logical record would leave less than one byte in the current record, a byte count of >FF will be written, and the logical record will be located in the next physical record. The first logical record in a physical record can never cause the physical record to overflow as the maximum logical record length is 254 and the physical record length is 256.

Physical location of the FDR - Points to the physical sector on the disk where the FDR resides for the associated file, and is used when it is necessary to re-write the FDR to the disk. It is maintained on read only accesses even though it is not required.

Current Logical Record Offset on Level 2 - Contains the physical record offset of the most recently processed physical record and is independant of READ or WRITE operations. Always contains the logical offset for Level 2 of the datablock which is currently in memory. It should be noted that this system causes fixed length sequential files to be accessed as relative access files on Level 2.

DATA BUFFERING

In order to buffer the data to and from the disk, a 256 byte buffer is maintained for each OPEN file. The buffer is located immediately above the FCB memory area.

One of the VDP RAM buffers is permanently assigned for processing VIBs (Volume Information Blocks - see previous parts of this series). If more than one drive is used in WRITE mode, then the bit maps are moved in and out of this area as required. This buffer is accessed for each access to the disk VIB.

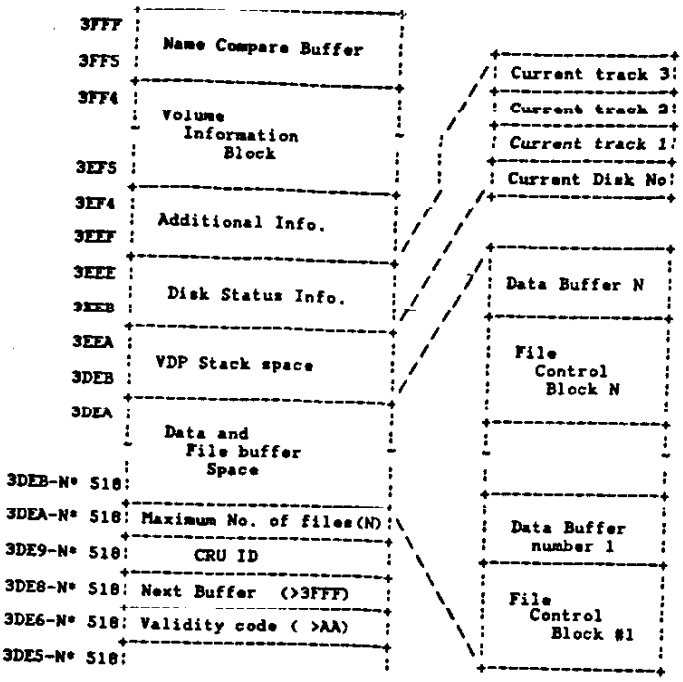
Every Level 3 WRITE operation to a file will eventually be passed on to Level 2 as a physical sector WRITE. To keep the number of disk accesses to a minimum, a flag (the MS bit of the Drive ID) to indicate that the current data buffer has been modified. The data buffer is only written to the disk if the next physical record access involves another physical record. If the file is closed then the last data buffer is written onto the disk if required (this is the reason why the ends of files go missing if you forget to close them!).

VDP MEMORY LAYOUT

The memory layout is outlined in the diagram (such as it is!) below. This block of memory is reserved by the power up routine in the Disk DSR ROM. The size of the area from then on depends upon the number of files which are allowed to be open at the same time, and is initially set to 3. This number can be varied between one and 16 by calling a subprogram (through CALL FILES from BASIC). Each extra file will of course take up 518 bytes.

As for each peripheral, the disk peripheral identifies its reserved area through its CRU address (unique for each peripheral). The area is 'validated' by an >AA byte, followed by the address of the previous top of memory. As the disk peripheral has the highest priority on power up, this entry will always point to the actual top of memory. The disk system does not use this however, and so will work equally well on other CRU locations.

The first entry after the CRU ID contains the number of files for which the area is reserved, and directly determines the length of the reserved area. After this entry come the areas reserved for the FCBs and the associated buffers for each file. In order to simplify(?) the buffer allocation, buffers are not allocated on demand, but as soon as a file is opened. The FCB and buffer are associated with the file for its entire 'open' life.



The VDP Stack area is used to simulate a stack based machine with the TMS 9900, giving the programmer the advantage of being able to use the multilevel stack oriented CALL/RETURN system, rather than the single level BL system used by the 9900 series processors. The stack can of course also be used to PUSH and POP registers and data to and from it.

The disk status information area is used to save the current track numbers of the (3) drives, and the most recently accessed drive number.
The additional information area is no longer used (by the 99/4A).

The VIB buffer is described above.

At the top of memory, an 11-byte buffer is reserved which is used for name comparison. Every high level entry point automatically saves the drive number and the 10 character file name in this entry. If less than 10 characters are available, the buffer is padded with spaces.

Next time: DSR subprograms

XXXXXXXXXXXXXXXXX

DISK PERIPHERAL D.S.R. SUBPROGRAMS

By COLIN HINSON
INTERNATIONAL TI-LINES, 11/86

LEVEL 1 SUBROUTINES

The lowest routines in the disk DSR are called level 1 subroutines. These routines make the higher levels independent of the physical disk medium, e.g. changing the disk software for a double density disk would only involve changing the routines on this level provided the physical sector size remains at 256 bytes.

There are two Sub-programs available on this level:

1. Sector Read/Write
2. Format disk.

The following paragraphs contain a description of the subprograms and their call requirements. All parameters are transferred through the FAC block in CPU RAM. This block is located at a relative position of >4A (which for the 99/4A is >834A).

All the subprograms are called by a 'BLWP @DSRLNK' followed by data statement: 'DATA >A'. (Note that the Editor Assembler manual is WRONG - it gives >10 for the data on the third line of the first paragraph on page 262). Before calling any of the subprograms, location >8356 (name length pointer) must be set up to point at the location in VRAM where the name length and the subprogram number have been written.

e.g. If subprogram >14 is called then a location in VRAM (say >1000) must contain 2 bytes, the first of which is >01 (the name length), and the second of which is the subprogram number >14. Location >8356 in CPU RAM points to the first of these bytes - i.e. contains >1000.

Error codes are returned in >8350

SECTOR READ/WRITE - SUBPROGRAM >10

The transfer block for this subprogram is:

>834A	(Sector number)
>834C	Unit # | READ/WRITE
>834E	VDP Buffer start address
>8350	Sector number

The meaning of each entry is:

Unit Number:

Indicates the disk drive on which the operation is to be performed. For a T.I. controller, this has to be either 1, 2, or 3.

READ/WRITE :

Indicates the direction of data flow:

0 = WRITE
NOT 0 = READ

VDP buffer start address. :

Indicates the start of VDP buffer for data transfer. The number of bytes transferred will always be 256.

Sector number :

Number of the sector to be written or read. Sectors are addressed as logical sectors (0-359 for a single sided single density disk), rather than as a track and sector number, which would require a knowledge of the physical layout of the floppy disk. The sector number has to be given in CPU RAM locations >8350 and >8351, and will be returned in CPU RAM locations >834A and >834B.

DISK FORMATTING - SUBPROGRAM >11

The transfer block for this subprogram is:

>834A	(# of sectors/disk)
>834C	DSR Ver|Unit #| # of tracks
>834E	VDP Buffer start address
>8350	Density | # of sides

The meaning of each entry is:

# of sectors/disk :

Is returned by the routine to provide compatibility between the normal controller and double density or SA200 systems.

DSR Version (This is the MS nibble)

0 indicates the format requires nothing special and can be done on any version of the DSR.

1 indicates the format requires the 2nd version of the DSR for one of two reasons. It may be because a double sided format is requested, or it may be because a # of sectors other than 35 or 40 is requested (but see below!).

Unit Number :

Indicates the disk drive on which the operation is to be performed. For a T.I. controller, this has to be either 1, 2, or 3. This is the LS nibble.

# of tracks :

Indicates the number of tracks to be formatted. In the only versions released, this entry has to be either 35 or 40!!! Upon return, this entry contains the number of sectors per track.

VDP buffer start address :

Indicates the start address of the VDP buffer that can be used by the disk controller to write tracks. The amount of memory used depends on the disk format. For a single density format, the buffer memory used is a nominal 3125 bytes. This can vary with disk motor speed to a maximum of 3300 bytes. To be compatible with double density versions of the controller (such as MYARC), the minimum buffer size must be 8K bytes.

Density - 0 = single

# of sides - Indicates the number of sides to format.

The above subprogram will format the entire disk on the given unit unless the disk in the unit has been hardware write protected. It can use any VDP memory starting at the location given in the transfer block.

LEVEL 2 SUBROUTINES

The Level 2 subroutines use the "file" concept, rather than the "logical sector number". Note that the file concept on this level is limited to an abstract type of file which has NO properties such as "program file" or "data file". A file on this level is merely a collection of data, stored in logical blocks of 256 bytes each.

The logical blocks on this level are accessed by filename and logical block offset. This offset starts with block 0 and ends with block N-1 for a file with a length of N blocks.

MODIFY FILE PROTECTION - SUBPROGRAM >12

The transfer block for this subprogram is:

>834C	UNIT #	Protect code
>834E	Pointer to file name	

The protect bit for the indicated file will be set or reset according to the information given in CPU RAM location >4D:

0 -Reset the file protect bit. The file is no longer protected against modification or deletion.

1 -Set the file protect bit. Disallow SAVE and OPEN for OUTPUT, APPEND, or UPDATE mode.

The pointer to the file name must point to the VDP RAM location of the first character of the file name. The name must be left adjusted in a 10 character field, right filled with spaces. No checks are made to ensure the legality of the file name.

FILE RENAME ROUTINE - SUBPROGRAM >13

The transfer block for this subprogram is:

>834C	UNIT #	Unused
>834E	Pointer to new name	
>8350	Pointer to old name	

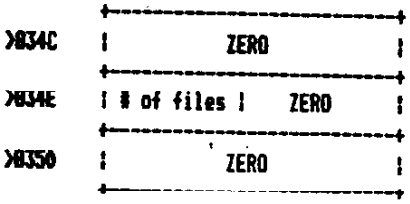
Both pointers to the file names must point to the VDP RAM location of the first character of a file name. Each name must be left adjusted in a 10 character field, right filled with spaces. No checks are made to ensure the legality of the file names.

Since the rename has to be done on the same disk, only one unit number entry is required.

Error codes are returned, as usual, at location >8350. The error codes returned are identical to the standard file management error codes, i.e. only the upper three bits of the error byte are significant.

BUFFER ALLOCATION ROUTINE - SUBPROGRAM >16

The transfer block for this subprogram is:



The "argument" for this subprogram is the number of file buffers to be allocated. Whilst it is not actually necessary to clear >834C, and >8350, experience has shown that strange results sometimes occur if you don't.

The effect of this routine is that an attempt is made to allocate enough VDP RAM space for disk usage to facilitate the simultaneous opening of the given nuumber of files. This number has to be between 1 and 16.

The disk software automatically relocates all buffer areas that have been linked in the following manner (see also previous article on VDP RAM allocation):

Byte 1 - Validation code

Bytes 2 & 3 Top of memory before allocation of this buffer.

Byte 4 - High byte of CRU address for given buffer area. For programs this byte is 0.

The linkage to the first buffer area is made through the current top of memory, given in CPU RAM location >8370.

The top of memory is also automatically updated after successful completion of this subprogram.

A check is made that the current request leaves at least >800 bytes of VDP RAM space for screen and data storage. If this is not the case, or if the total number of buffers is zero or greater than 16, the request is ignored and an error code will be indicated in CPU RAM location >8350.

Successful completion is indicated by a 0 byte in CPU RAM location >8350. A non-zero byte indicates unsuccessful completion.

Next time:

Subprograms >14 and >15 - Direct file access.

XXXXXXXXXXXXXXXXXX

President's Column
By Don Arsenault

First of all I would like to thank all of you for the the show of support in the election at the December meeting.

The past year has been a very eventful one in both the TI world as a whole and within our User Group. We have seen the introduction of the Geneve which promises more power and capability to the TI community. The user Group has purchased a Geneve and it is available for use by our membership at the YDC meetings and any other time by arranging with Rice Hall. We also acquired a diagnostic card which can identify many problems which can occur in the TI console. This card is also available for your use at the YDC meetings and at any other time by contacting our Vice President, Allan Silversteen.

There are many things that I hope to get accomplished this year with your help and support. Some of them are as follows:

A. Make known the facilities and capabilities of our club to all our members and to the many non-member TI owners. We can do this by making ourselves more visible within the PACS group by participating in the PACS computer show which is upcoming in March, and by each one of us talking to other non-member TIers about the benefits of our User Group. There are many TIers who have no idea that we exist. I would like someone who has the artistic talent to design a poster utilizing some of the many graphics available in such programs as TI-Artist, Graphx, Joypaint, The Printer's Apprentice, etc., which we can post in places such as Computer stores and even supermarkets, to advertise our presence.

B. Make the programs in our library more readily available to the membership. This can be done in various ways:

1. Having a complete copy of the library available at the YDC meetings on the 1st Saturday of the month so that any member can make copies of the disks they want.

2. Have copies of the disks available at the LaSalle meetings so that any member can get the programs they want. There would be a small fee (probably about $1.00 per disk) to cover the expenses.

At the January LaSalle meeting we will have available the latest version of FUNNELWEB with documentation for distribution. We will also have a short demonstration of

the program, pointing out some of the new features. We also are scheduled to have a demonstration of MYARC's new Hard disk and floppy controller card.

At the YDC meetings we are still having courses on MultiPlan and Communications as well as the Hardware session where you can get help on practically any hardware project or problem which you may be having.

Thanks again for your support and I hope that together we can make this a banner year for the User Group.

==========

TI-WRITER OVERVIEW PT 1
By JO ANN COPELAND
INT. TI-LINES 07/86

==========

For those of us who were fortunate enough to gain access to the TI-Writer Word Processor, you found having the manual on hand even better. All those confusing keys and statements were simply looked up for explanation. However, if you weren't fortunate enough to get the manual and would like to review the keys and statements YOU ARE IN LUCK! Or, if considered, you are in trouble, as it is the crazy abnormal person again, trying to prove she knows what she is talking about when we all know she doesn't. So, I am here to prove just that. If the readers of this newsletter can stand it, I will try to take a step-by-step review of the Writer/Funlwriter and hope that someone, somewhere, out there can gain knowledge from it. Well, we can always hope, can't we?

For starters, (yes - go ahead and cringe so you can get it over with), the hardware/software required should follow as: TI-Writer Word Processor Cartridge; Program Diskette; TI 99/4A Console; Monitor; Disk Memory Drive; Peripheral Expansion System including Memory Expansion, Disk Controller Card, and RS232 Card; and an RS232 compatible printer. Of course, the infamous stand-alone peripherals may be used. Whew! If you have the TI-Writer Word Processer Module and Disk you will use those, and if you have Funlwriter you will use the Extended Basic module and Funlwriter Disk. After loading, these two work along the same lines, although Funlwriter, in this author's opinion, is much better. I have both, but find Funlwriter easier to use and quicker, plus it saves on feed paper. Okay, check off and see what you've got and let's go from there. Or drop this and go have a cup of tea, and come back later to go on to the next column. The latter are the smarter!

I'll skip the normally good reasons for having the Writer / Funlwriter and assume we all use it consistently.

First, I am going to use CTRL 3 to change my screen color. This blue color gets awfully boring after a while. While I'm doing that, load up your disk and let's go.....

After loading, you'll find another menu...

$WORD PROCESSOR$

PRESS

1. **For Text Editor**
2. **For Text Formatter**
3. **Utility**

Press 1 and you will find yourself involved with the Text Editor. Here you can create, edit or save documents, reports, books, etc. You will also see the lines appear as follows:

```
Edit, Tab, Files, Lines, SearcH, RecoverEdit

   0    (cursor mark)
0001
 $End of file version
```

The first line is the command mode prompt line; second is cursor; third is line number; and fourth is end-of-marker file. The very top line offers you the available categories to choose from, such as going to Files, or Recovering material or setting up Tabs. The cursor should be blinking and is filled-in (the other is hollow in Fixed Mode). Line numbers mark lines, as stated, and can be used with other commands such as copying, moving, etc. End-of-File marker shows the last line of text and is only an on-screen indicator. You will also find, if you use the arrow keys (FCTN S, X, D, E) that if you go across the screen it seems to 'wrap'. You will find this called Word-Wrap, and consists of three 'windows' (views). A line is up to 80 columns (characters) long but only 40 show on screen at one time. So you have three overlapping 'windows', i.e.:

```
!          (-------second window-------)
!(----------first window-----)(-------third window----)
!
!(---------------full 80 column width---------------)
!
```

As you pass your cursor along the next window is automatically dislayed and at the end of the line, instead of hitting a carriage return as on a typewriter the line automatically goes to the next line so you can keep typing and find yourself going down one line at a time. If you want to find the next window from where your cursor is displayed, try pressing FCTN 5 - this is 'horizontal block scrolling'.

==========

HIGH RES GRAPHICS
PART 3

by Ann Dhein

==========

Scratchpad Memory - Most drawing packages have provisions for setting aside part of a picture and later adding it to another picture. This "scratchpad" memory can be handled in two ways: by saving a permanent version of the clipped picture onto a disk which you can reload as needed; or by storing the picture-part in intermediate memory where you can recall it when you need it, even though you have loaded new picture files in and out of the program since the picture part was saved.

When this type of picture-part is saved to disk it should not be confused with a regular picture file. When a picture file is loaded into your program, whatever you had on the screen before is erased and gone, and the new picture takes its place. Picture-parts, however are loaded IN ADDITION to whatever else is already there. These small pictures have become very popular with the drawing community so that they have there own special term - CLIPART.

Each program is unique in its handling of this additional storage. Joy Paint uses internal storage for a Cut and Paste method much like the paint programs for other popular computers do. All screens are saved in the same format. When something is wanted from another picture, save the current picture first, then load in the picture to be borrowed from. "Cut" out the piece you wish to use. Reload the original picture and "Paste" the new part any where on the drawing.

Bitmac uses the "Store" function for internal temporary storage. Current screen graphics can be overlaid with graphics stored on a disk, using what is called "Boolean Input". This allows special graphics effects which are unique to Bitmac.

TI Artist also has unique storage methods. Besides the normal full screen picture files, parts of picture can be saved as "instances" or "slides". Slides are a collection of up to 24 miniature designs that can be independently designed, rotated, and moved around on your drawing. Instances are images that can be added to your drawings or combined togther in whatever manner you wish. They can become a permanent, editable part of your drawing. The nice thing about instances is that they are saved in a DISPLAY VARIABLE 80 format which can easily be transported to Extended Basic programs or TI Writer files as well as being used for clipart.

Graphx has a very powerful "Clipboard" feature. With it you can create and store clipart permanently on a disk and it is also possible to copy a portion of one picture into another, much like Cut and Paste. A portion of a picture, or even several pictures, can be stored, then decided on later as to which ones to keep and which ones to erase.

USE OF COLOR

In the high resolution mode each graphic position available to be used on our electronic drawing board is called a pixel. You may remember being told that the screen is like a grid with 256 pixels across and 192 pixel rows; and that each individual pixel on the screen can be turned off or on separately while you are drawing - all 49,152 of them! Right? Wrong, if you are using color!

Color resolution for the 99/4A is not the same as drawing resolution. We still have the same 192 rows of pixels, but instead of 256 pixels across, we have only 32 graphic positions across each row. Each row of pixels is grouped in eights, starting from the left of the screen, and each set must be the same color - a foreground and a background.

The foreground is the color assigned to the brush or pencil line in each eight-pixel group. The background is the color assigned to those same eight pixels when the pencil is not used. When you first begin using the drawing board, all of the eight-pixel groups have been assigned the same two colors. The color you see before you begin drawing is your background, and, of course, the pencil line is your forground color. You may also see a third color in the form of a border around the perimeter of the screen. This is the screen color. If you don't see it, that means the screen has been assigned the same color as the background.

Now you can see why color resolution is 64 X 192 instead of the drawing resolution of 256 X 192. Any given group of eight horizontal pixels MUST be the same two colors. The groups on either side can carry entirely different colors, but each group is limited to two colors. Knowing this, and arranging your drawings according to the color boundaries is important when working with color.

Most programs make full use of the 99/4A's 15 brilliant colors, allowing control over the forground and background colors, and in many programs over the screen color as well. Sometimes the screen color is called the "backdrop".

All programs using color allow the swapping of one color in a drawing for any other. When the exchange takes place, every incident of that color on the screen is swapped for the new one. Additionally, some programs

like TI Artist and Graphx allow selective repainting of a chosen area.

Some of the programs provide special helps for working with color. TI Artist provides a function that lets a special color cursor move on color boundaries. Graphx does the same, also providing a "Grey and White Checkerboard" function which is handy for planning drawings which will use a lot of different colors. This makes it much easier to plan the various colors in your picture so that they don't bump into each other. When you no longer need the grid simply choose the "Remove Gray Boxes" option.

For **special color effects**, two programs that shine are Draw-A-Bit with its Redraw feature described elsewhere, and Paint 'N Plot makes limited use of color. Only two are used at any one time - foreground and background. These colors can be easily switched so you can see how the various combinations of color look together.

Besides the Graphic Package, which doesn't use color either, Joy Paint is the only major paint program not using color. Here the emphasis is on the manipulating of picture components, and color is used as a background, with the pencil line being your choice of either black or white. Painting refers to filling shapes with the many patterns available, or using the air brush to "spray paint" an area with a chosen pattern.

A **Slide Show** is a method of presenting pictures in a selected order. Bitmac is the only program with this feature built in; Draw A Bit and Draw 'N Plot have disk demos that you can adapt for your own pictures. TI Artist has an excellent companion disk called Display Master that gives you many options in designing your own slide display. Asgard Software puts out a slide show program for Graphx files.

The **Undo** command lets you "take back" the last step of a drawing. If something was moved or erased that shouldn't have been, no harm done, just "UNDO" it. Joy Paint is the 99/4A's only program with this feature but it is quite command in paint programs for other computers.

Like Undo, each program has special features not shared by the others. If you are in the market for a new paint program, one of these might be just the feature you were wishing you could find. For instance, Joy Paint has a drawing area that is actually 92% larger than the screen. To see the rest of the drawing board, the screen is used as a window, and can be moved from side to side or up and down. When the screen dump program is used the whole area, not just what is visible on the screen is printed.

Do you have a second computer that you have wished you could tie in to your 99/4A? The Bitmac software will let you do just this. When the coprocessor function is in effect, the other computer (not necessarily a TI) can manipulate data while the 99/4A is processing elaborate graphics from that data.

Bitmac has a **Cursor Report Feature** which can be turned on or off as desired. It keeps track of the actual pixel location of the cursor. The program also lets you scroll your picture one pixel at a time to the right or left, up or down on the screen. This is handy for getting a drawing onto color boundaries, and also for special effects using the Boolean inputs AND, OR, and XOR.

The TI Artist instance file was already mentioned above as being excellent additional storage for clipart, because these files can be added so easily to any picture you are currently working on. The instance file is invaluable for using as a vehicle to transport your artwork to other mediums. Many support programs have been built around the ability of these instances to be easily used, including Font Writer (Asgard), Art Convert (Trio+) and Character Set and Graphics DEsign III (Texaments).

Besides being used for planning color in drawings, the unique gray box function in Graphx can be used for designing schematics and other precision drawings which require precise measuring.

The Graphx clipboard also lets you experiment with computer animation. If you store the appropriate images on the clipboard you can create short, animated sequences which you can display against a background of your normal Graphx pictures.

Like the Norton Graphics Package, Draw 'N Plot is primarily a programmer's tool. Unlike the Graphics package however, Draw 'N Plot has a very nice, full-featured drawing board. Also, the routines in Draw 'N Plot are in assembly language which considerably speeds up operations. Draw 'N Plot makes an excellent program to design your own Extended Basic programs around; however, memory can be a problem.

Draw A Bit is realy a full-scale programmer's tool too, but the programmer must be somewhat conversant in assembly language as well as Extended Basic to use it with his own programs.

As you work in the Draw A Bit environment, your picture is automatically saved for you in intermediate memory. Any time you wish you may clear the screen and with the push of the right keys, redraw the picture, line

for line. This is a fascinating procedure to watch. Pictures may also be saved in this Draw mode if desired. Also interesting to use is the Connect-Dots option. This is like a line function except that you plot all your dots first; then the lines appear when you are ready for them.

Built right into Paint 'N Print is a font editor that will let you easily change the shape of the resident alphabet. The companion disk additionally allows editing of the texture characters for all sorts of special textural effects.

Paint 'N Print is the only drawing package which allows a screen dump to be in color, providing you have the right printer (the Axiom GP700).

Super Sketch is the only program that includes a touch tablet. This graphics tablet, although deceptively simple looking, is a precision tool that accepts commands through a control arm moves the pointer (your pen) around the tablet, and the computer keeps track of where this pointer is at all times. In this manner, any picture placed on the tablet can be traced onto the screen. The device is so simple that even a child can use it easily.

The Artist Extras package from Inscebot allows the use of the Super Sketch touch tablet with TI Artist. When used this way, the tablet becomes an integral part of the TI Artist program and this is used inplace of a joystick or trackball to allow a design traced with the tablet to appear on the screen.

Next month will offer a closer look at Printers and Screen Dumps, File Management, and additional support packages.

<(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)(*)>

****
PRACTICAL PROGRAMS
fm LA 99ers
****

*** CHANGE SCREEN AND TEXT COLOR IN XB ***

Save program on disk under the the name of "LOAD" Screen and text will change to desired colors. Change line 110 to your desired display colors.

B = Background Color

F = Foreground Color

```
100 CALL CLEAR
110 B=2 :: F=16
120 C=16*(F-1)+(B-1)
130 CALL INIT :: CALL LOAD(9984,C,C,C,C,C,C,C,C,2,0,7,15+
B,4,32,32)
140 CALL LOAD(9999,40,2,0,8,0,2,1,39,0,2,2,0,8,4,32,32,36
,2,0,8,8,4)
150 CALL LOAD(10021,32,32,36,2,0,8,16,4,32,32,36,2,0,8,2,
4,4,32,32,36,4,91)
160 CALL LOAD(-31804,39,8)
170 CALL LOAD(-31952,255,231,255,231)
180 END
```

*** BASIC COLOR SHOW ***

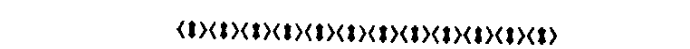
```
100 RANDOMIZE
110 CALL CLEAR
120 FOR A=1 TO 16
130 CALL COLOR(A,INT(16*RND)+1,INT(16*RND+1))
140 NEXT A
150 CALL SCREEN(15*RND+1)
152 FOR I=48 TO 87
153 CALL CHAR(I,"FFAAFF55FFAAFF55")
154 NEXT I
156 FOR X=88 TO 123
157 CALL CHAR(X,"3C1881C3C381183C")
158 NEXT X
160 FOR A=1 TO INT(15*RND+1)
170 X=INT(19*RND+1)
180 Y=INT(19*RND+1)
190 W=INT(500*RND+1)*(RND+1)*(RND+1)
200 FOR B=1 TO INT(10*RND+1)
210 Z=INT(136*RND+24)
220 CALL VCHAR(Y+INT(6*RND),X+INT(14*RND),Z,W)
230 W=INT(W*RND+1)
240 CALL HCHAR(X+INT(6*RND),Y+INT(14*RND),Z,W)
250 NEXT B
260 NEXT A
270 GOTO 120
```

*** PROGRAM PEEKER ***
by G.Mineo

This routine should be saved as a MERGE file. Merge this into an X-BASIC program and type "RUN 30000". You will be able to analize the memory.

S = Forward
Any other Key = Backward

```
30000 DISPLAY AT(24,1):"Start Byte>-1   (First=-1)"
30010 ACCEPT AT(24,12)BEEP SIZE(-4):SBYTE
30020 DISPLAY AT(24,1):"High Byte> -2000"
30030 IMAGE BYTE###### INST>#### ASCII #
30040 FOR I=SBYTE TO HBYTE STEP -1
30050 CALL PEEK(I,INST)
30060 DISPLAY AT(24,1):USING 30030:I,INST,CHR$(INST)
30065 IF INST>128 THEN CALL HCHAR(24,19,42)
30070 CALL KEY(0,K,S):: IF S=0 THEN 30070
30080 IF K=83 THEN I=I+1 :: GOTO 30050
30090 NEXT I
30100 END
```



CHARACTER GRAPHICS WITH TI-WRITER

by Rod Cook

Graphic characters can be defined in TI-WRITER and printed to the printer by using a combination of the transliterate command and the graphics control codes of the printer.

The transliterate commands has the format:

".TL n:char#,char#,.....,char#"

where n is the special character number and char# is the decimal number to be transmitted to the printer. For example, in the following command:

".TL 0:46,84,76"

anytime the special chartacter '0 is encountered in the text, the FORMATTER will transmit the numbers 46, 84 and 76 which in this case happen to be a period followed by a capital T and L. The value of char# can have any value between 0 and 255 although ASCII values only go up to 127.

The transliterate command will define the graphics for the charater to be printed. One transliterate command per character will be required. Each command will have essentially two parts: the control codes to setup the printer into graphics mode and the data. For the purpose of illustration, the control code discussed will be for an Epson MX80. The control code for printer graphics is:

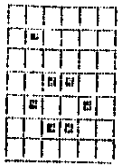
<ESC> "K" N1 N2

where <ESC> is the escape character, number 27 and "K" is number 75. N1 and N2 are numbers that are used to specify how many data numbers follow. This control code puts the printer into a graphics mode that prints 480 dots per 8 inches. An 80 character line is also 8 inches long, therefore 480/80 = 6. There are 8 dots per character and it will take six data numbers to specify the character. The control code portion of the transliterate command will look like this:

".TL 0:27,75,6,0,...data..."

where N1 is 6 and N2 is 0 which tells the printer there will be 6 data numbers to follow.

The data numbers tell the printer which of the 8 pins to fire on the printhead for each of the six verticle rows of dots that make up the character. For example the graphics for the special character '0 are coded as follows:



where in vertical row 1 none of the dots are on so they add up to zero. In vertical row two, the dots at 4,8 and 64 are on so they add up to 76. In vertical row three, 16 and 2 dots are on so they add up to 18 and so on with the remaining three vertical rows.

So the transliterate command to print the special character '0 looks like this:

".TL 0:27,75,6,0,0,76,18,18,12,0"

and anytime the special character for zero (shift 2 in the TI-WRITER special character mode) is encountered in the text by the FORMATTER the transliterated values will be sent to the printer which will result in the defined graphics character being printed.

The 6 by 8 grid that was printed above was printed this way. Four Graphics characters are needed to build the grid.

That are:

The respective definations are:

```
".TL 65:27,75,6,0,255,128,128 "
     128,128,128
".TL 66:27,75,6,0,255,0,0,0,0,0"
".TL 67:27,75,6,0,128,128,128, "
     128,128,128
".TL 68:27,75,6,0,255,128,188, "
     188,188,128
```

so anytime ASCII characters 65, 66, 67 or 68 are encountered in the text, the above graphics characters will be printed. ASCII character 65 is an A, 66 is a B, 67 is a C and 68 is a D. So to print the above grid, the following pattern of characters would be needed:

```
128 AAAAAAB
 64 ADAAAAB
 32 AAAAAAB
 16 AADDAAB
  8 ADAADAB
  4 ADAADAB
  2 AADDAAB
  1 AAAAAAB
    CCCCCC
```

Note the numbers will be printed just as they appear because they have not been redefined. Once the graphics have been printed the A thru D will have to be transliterated back to their regular character if they are to be used in text. The following commands will do this:

```
".TL 65:65
".TL 66:66
".TL 67:67
".TL 68:68
```

There are some limitations as a result of working within the limits of the FORMATTER.

1. It appears the physical length of the transliterate command can not be greater than one line.
2. It also appears that each character is limited to 6 vertical rows of dots. I have not been able to print a character longer than 6 rows within the FORMATTER.
3. The FORMATTER insists on putting white spaces on the top and bottom of the page. To print graphics that are continuous from line to line, as is the grid above, requires a line spacing less than that of six lines per inch. To keep the same white space at the bottom of the page and on subsequent pages will require adjusting the line spacing after the graphics to compensate for smaller line spacing of the graphics.

(%)(%)(%)(%)(%)(%)(%)(%)(%)(%)(%)(%)(%)

%%%%%%%%%%%%%%%%%%%%%%

!%%%(OUCH

by Dennis Hathaway

%%%%%%%%%%%%%%%%%%%%%%

Winter is here and with winter comes low temperatures and extreamely low Relative Humidity (RH). Low RH presents us with the problem of dealing with very high levels of static electricity. So it is timely for a discussion on static. What is static, what can it do to our hardware, and what precautions can we take to reduce problems.

First what is it. Every person, animal, plane, truck, auto, etc. carries with them an electrostatic charge (Q), except when they (or it) are in direct contact with a conductive path to ground. We accumulate this charge as we move around, for example walking on a carpet, touching things (or beings) that carry a charge greater than ours, the rubbing together of our items of clothing (especially clothing of man-made materials), even the friction of air flowing past us. Depending on the materials involved we can become charged either positive or negative with respect to ground. When we touch something with a different level of charge we often feel a sharp tingle, or minute burn (oddly enough, we don't usually feel anything if the potential is less than 2300 volts, because then the total static power involved is small). This sensation is caused by the discharge of the static we were carrying, or to give it it's full name Electro Static Discharge (ESD). More on ESD later.

The charge we carry (Q) is given by Q = CV, where Q is in coulombs, C is in microfarads, and V is in volts. For a human being C is usually around 100 to 250 picofarads, depending mainly on overall bulk, size of feet, and type of footwear. Because womens shoes usually have thin soles, their capacity (C) is often higher than that of a man. Typically Q ranges up to 5 microcoulombs.

A calculation will show that a person whose Q = 3 microcoulombs and whose C = 200 picofarads will be at 15 KV with respect to ground (where 1 KV = 1000 volts).

This voltage can in extreme cases be as high as 40 KV. Beyond this point the charge will bleed off through the air, and may sometimes even form a corona. The drier the air, the less is the bleed off effect and this is why we should be concerned when the RH is low. In upper Wisconsin during the fall of a few years ago, using a grounded electrostatic meter, I measured 23 KV on a man walking towards me over a vinyl tiled floor, while he was wearing crepe soled shoes. At the same location, the plastic cover on a notice board was at 45 KV. Even allowing for meter inaccuracies this still gives an indication of the level of voltages out there at this time of year. We must not let this cause us to think that we generate static only when the RH is low. This is not the case. We generate static ALL YEAR LONG, but when the RH is higher the charge bleeds away more readily. I am sure that all of us have at some time touched the screen of our monitor/TV while working at the keyboard, and have heard the air crackling as we became charged to some 12 to 15 KV. When we next touched the keyboard we partially discharged into the console. TI were fully aware of this problem when they designed the 99/4A, and

have almost certainly taken suitable precautions to prevent damage to the console when we zapped it in this way.

Now what can happen to our equipment if we pass our charge to it in a manner TI did not anticipate? Actually we share our charge with the equipment, untill both we and it are at an equal potential, then current flow ceasses. In practice a typical discharge is found to be somewhere in the order of 20 millijoules (where a joule is 1 watt/second). Now when we discharge into our equiptment it takes place very rapidly, depending on our capacitance C, and our skin resistance (usually around 1000 ohms). The effect is such that we can be feeding some 20 amps into the equiptment for a period of some 100 nanoseconds. If we do this to the connector contacts on a module, or on one of our PEB cards we will cause damage to them. In some cases it will be obvious when a device is destroyed because the system quits. Usually however we end up with a device that is only partially damaged. The device is then known as being amoung the "walking wounded". It carries on working, but is likely to die on us at any time, maybe weeks later when we have completely forgotten that we previously zapped it with 20 millijoules. It is general knowledge that MOS devices used in many TI assemblies are susceptable to this kind of harsh treatment, but it must be realized that bipolar transistors, diodes, and other devices are also affected similarly, but to a somewhat lesser degree.

How do we prevent static damage to our 99/4A? As we are about to sit down at our terminal we should bear in mind that every item around us has some capacity to ground so by touching them briefly we share our charge with them. Do this to a couple of items (preferably large metallic ones) like a chair, or a filing cabinet, etc. Personally I use an exposed screw head on my grounded AC power distribution expansion strip which sits on top of my PE Box. If we are working on a printed circuit (PC) board outside of our PEB, or of our printer then a more elaborate method is needed. Consider, while sitting at the bench/table we may raise our potential by as much as 1000 volts each time we reach out to pick up a tool or device. An aid known as a wrist strap will help here. A conductive band goes on the wrist, and is connected via a 1.5 Megohm resistor to ground. This continuously bleeding away any static as fast as we generate it. The resistor limits current if we accidentally touch something with mains potential on it. These wrist straps are available from most electronic parts vendors for only a few dollars.

While the above discussion is directed to work on the 99/4A, it applies equally to other electronic items - radio, TV, VCR, etc. Static can ruin most electronic equiptment.

It would be well to ensure that ones spouse and children are made aware of this static problem, so they will not nullify our efforts to prevent damage to our equiptment.

We noted earlier that when a static charge is passed from one body to another, a current flow occurs that is initated by a breakdown of the air as we approach the object, and this creates a spark (which we can observe in a darkened room). This current flow is what we now know as ESD. There are other forms of ESD that create sparks of much greater magnitude. For example, lightning is a discharge of the static build up within the clouds. It may be of interest to the reader that bolts of lightning can pass current in the order of 80,000 amps, though the average stroke is around 17,000 amps. This of course is happening when the RH is very hight, often 100%, quite contrary to the situation discussed earlier in this article. Unfortunately (or fortunately, depending on your viewpoint)TI Writer does not readily lend itself to the production of mathematical formulas, so this article will leave it to the reader to confirm calculations at their leasure. It will suffice here to say that due to the inductive reactance of the path taken by the current to ground there are some extremely high voltages produced by a lightning strike. We must realize that while the ohmic resistance of a length of cable may be very low (say 0.01 ohm) its impedance to a 2 nanosecond lightning pulse may be thousands of ohms, and the high voltage appears across this impedance. Power lines, telephone lines etc. may be hit by lightning, so that voltages in the order of hundreds of thousands of volts with respect to ground may appear. The insulation of household wiring breaks down at around 6000 volts, so we can be reasonably sure that we will not encounter anything greater than this on our equiptment. However 6000 volts will kill our 99/4A (or us), so it is wise to abstain from using the system when there is a storm in the area.

When a large industrial motor is turned off, it injects a voltage spike onto the power line. While the power utility uses primary surge arresters to limit this surge, it is often found that a spike of up to 1500 volts between line and neutral appear at our premises. This spike is so brief that it is not usually noticed on our house lights, but our 99/4A power supply will see it, and may pass it on to our system. It is therefore advisable to have a suitable surge arrester in series with the AC feed to our system. This will pass the spike harmlessly to ground.

An additional problem that can arise, is that ESD can cause some strange program quirks that leave us bewildered for hours. When ESD happens, the current flow produces a magnetic field, known as an Electro Magnetic Pulse (EMP), and if this invades our system with sufficent intensity, the resultant voltage induced into

our system as the EMP traverses cables and PC board artwork, may alter the state of one or more bits in RAM, and we all know what this can do to a program.

To summarize, we have seen how man-made static can damage almost as disastrously as Mother Natures lightning, but how with a few relatively simple savegaurds we can avoide trouble.

<8><8><8><8><8><8><8><8><8><8><8><8><8>

NOTICE
MEMBERSHIP
RENEWAL

Hay gang!! Here it is almost mid January 1988 and only a small percent of you have paid your renewal dues for the year. We hope you are planning on staying a part of our TI family. We would like to make this a bigger and better User group in the coming year but we need your help and interest to make it all work, so come on and get reregistered. Also if your dues is not paid by this time next month this will be the last NEWSLETTER we will be able to send you until your membership is reinstated. This is also true with regard to, "THE DATA BUSS" from PACS.

So come on we need you all!!!!!

*** ANSWERS TO LAST MONTHS CROSSWORD PUZZLE ***

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

WORD LIST:
MICROPENDIUM
SPREADSHEET
COMPUSERVE
TEXAMENTS
RYTEDATA
ASSEMBLY
DATABASE
HORIZON
TEXCOMP
TIDBITS
EXBASIC
GENEVE
TRITON
ASGARD
PASCAL
MYARC
PILOT
TENEX
FORTH
RAVE

TI WORD SEARCH

	1	2
	1234567890	1234567890
1	TGPFCFPASR	SLZFJEUWQL
2	NHPASCALTY	PPPV1XTFRI
3	FOPILOTREL	XENETQVJHA
4	XRKHDQYRXM	PSOMBXLSWU
5	ULJTXTAAAL	ASSEMBLYGF
6	IZXYEVHXMC	DTNMFUGURI
7	TOGDELHSER	UVHTTMNAVI
8	ONATXZTPNO	BOPPFVGCRG
9	UTRWUEXRTP	DRAGSANQAY
10	ADKIXPEESE	COMPUSERVE
11	OVUCRXDAAN	HKBRCZEFBE
12	VFQEBEEDLD	J1GRLPMEMJ
13	NMQAVSMSHI	HCRAYMKZFJ
14	PZSETEVHNU	VVWMXVUXRX
15	T1NWFRWEVM	DATABASEJQ
16	CEPHOIIEBS	QXDJCDDCMK
17	GZRQRDMTTI	DBITSCJAVW
18	HDXXTHRAOJ	NAQCAKXGMR
19	UCXFHXTPWN	AAQJLUVTAB
20	XXGYXGGABN	XRRZULCLML