Atlanta 99/4A Computer Users Group

CALL NEWSLETTER

VOLUME IV NUMBER 1 JANUARY 1986 Atlanta, Georgia

PRESIDENTS CORNER

WHAT HAPPENED TO THE JANUARY MEETING?

First, an apology to any who were inconvenienced by the regular monthly meeting for January being cancelled. The library was closed that Sunday due to the Martin Luther King Jr. holiday celebration. Library personnel informed us on Thursday, January 16th about the closing.

Our first priority was to inform as many people as possible. Notices were posted on the two club bulletin boards. Many of the regulars were informed by word of mouth. I called most of the newer members so they wouldn't waste a trip and begin to wonder what had happened to us. Even so there were a couple of people I know who made the trip to the Atlanta Public Library on Sunday the 19th only to find the library closed. To those people, I can only offer apologies.

With the cancellation of the regular meeting, it was decided to have an Executive meeting on the day instead. That is another term for a meeting of the officers, chairmen and committee members, and those interested in doing the work that makes the club run. We met (read that as sardined) at Ed Banovatz's apartment. About twenty showed up. The two major concerns were getting people to work on the Newsletter and the Club Library. Marshall Gordon and Ed Banovatz headed up those two groups respectively. Other issues discussed were considering another location for our monthly assemblage (Ga. Tech was suggested) and improving the meetings. It was thought that following a printed agenda and presenting more back to basics programs would help in this.

The response to this impromptu meeting was most satisfying. We have made passionate pleas many times for assistance. I hope the results are as impressive as the turnout.

Gary Matthews

EDITORS NOTES

In the July '85 newsletter, on page 6 we published changes to TK-Writer's LOAD program. I have received several letters stating that the changes did not work. Since my copy with the changes works fine I could not understand the problem.

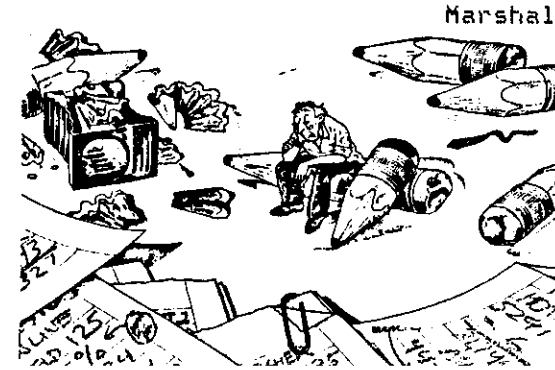
Someone was finally kind enough to send me a copy of his program, with the changes and I finally understand the problem. Apparently two extra lines were added to the program that were not in mine. Since the changes load a short assembly language program from Ex-Basic that looks for an internal program location. The extra lines change the PEEK location that is being sought and any attempt to load the Editor or Formattor program is doomed to failure.

The program below is correct and works on my system. Please use it as a guide for your own version.

```
>100 CALL CLEAR :: CALL INIT
 :: CALL PEEK(-2043,A,B):: IF
 A()84 OR B()75 THEN 108
>102 CALL LOAD(16360,85,84,73
 ,76,73,84,250,212,70,79,82,7
 7,65,84,250,132,69,68,73,84,
 79,82,250,22)
> 104 CALL LOAD(8196,63,232):
 : GOTO 110
>108 CALL LOAD("DSK1.TK-WRITE
 R")
>110 DISPLAY AT(6,2):"PRESS ;
 ": :" 1 - TO LOAD EDITOR": :
 " 2 - TO LOAD FORMAT TER": :
 " 3 - TO LOAD UTILITY"
>120 CALL KEY(0,K,S):: IF S=0
 THEN 120 ELSE IF K<49 OR K>5
 1 THEN 120 ELSE K=K-48
>130 ON K GOTO 140,150,160
>140 CALL LINK("EDITOR")
>150 CALL LINK("FORMAT")
>160 CALL LINK("UTILIT")
>170 END
```

If you still have problems send me a copy of your program listing and I will try to help.

Marshall



CALL NEWSLETER

CALL NEWSLETTER is the voice of the Atlanta 99/4A Computer Users Group. P.O.Box 19841, Atlanta, GA. 30325.
It is published at least 10 times a year. The A9CUG is not affiliated with any commercial company or organization.
CALL NEWSLETTER is published by and for the members of the A9CUG to enhance their knowledge of home computers. CALL NEWSLETTER is composed of articles written and/or donated by members of our group and from articles appearing in other home computer users' groups around the world. Opinions expressed by the authors do not necessarily represent those of the Officers or members of the A9CUG.
Permission is hereby granted to any users group recieving our newsletter to reproduce any article appearing in this newsletter, unless the article is otherwise noted, provided credit is given to the author and CALL NEWSLETTER. The A9CUG freely exchanges newsletters with other groups around the country. If another club would like to receiver our newsletter but does not have one of their own to exchange; we will gladly sent it to them. We do ask that they send $5 a year or 50 cents a newsletter to help cover costs.
Membership is open to family and individuals who own or are interested in using and programming home computers. Membership includes copies of this newsletter as they are published, access to the meetings of the main group and sub groups, and the groups Public Domain Library. Annual dues are $15.00.

**** C L U B O F F I C E R S ****

Gary Matthews 233-3096	President
Jim Hubbard (1)345-5905	Vice President
George Sears 396-4112	Treasurer
Melvin Carter	Secretary
Marshal Gordon	Newsletter Chairman
B.L. Miller	Newsletter Staff
Ed Banovatz 993-5399	Library Chairman
Charles Dupree 366-1914	BBS #1 Sysop
Jim Fairchild 991-6250	BBS #2 Sysop

F O R S A L E

SELLING - Console,Periph.Exp.box with DD Controller, 32 K card, Myarc RS232, Star Gemini 10X Printer, Magnavox Amber Monitor, Cassette Recorder & cables, 17 cartridges, MBX Expanision, 4 languages, TI-Writer, and lots more. Newnan,Ga Robert Stephens (404)253-2623 8:45pm M-F

SELLING - Exp box,(all TI cards)32K,RS232, Disk Cntrl,TI Disk drive,Muliplan,TI-Writ, Ed/Assembler. All for $300 Call Al Patrick (404)441-2970 Norcross, Ga.
Also selling Console,Speech synth,Ext/Bas, Mini-Mem.,TI Prog. Recorder,24 modules, numerous tape programs.All $250 Al Patrick

MINI - MEMORY BATTERY

by Richard J. Baley
68 A Church St.
Gonic, N.H. 03867
NH99ER USER GROUP

Apparently some people have checked with TI and found that it would cost up to $35.00 to replace the battery in their Mini Memory. However for those brave souls who are willing to replace the battery themselves, it can be done for $1.79. To find if your battery needs to be replaced, measure the battery voltage, it should be 3 volts, if it's much less than that, replace it.

The battery you need is a Radio Shack CR2032 (CAT#23-162). These cells have a shelf life of between 5 and 10 years and should last almost that long in the circuit. The case is the postive terminal just like the original but unlike the original, the CR2032 doesn't have leads and these must be carefully soldered on.

===> WARNING!!! Lithium batteries can be destroyed by heating them and certain types can EXPLODE!!!!!!!!!!!!!

If you don't think you're competent to make this modification, don't try, you might destroy your Mini Memory, or worse.

Scrape the center of the case where you are going to solder a solid #20 (or there abouts) wire. A lead from a 1 or 2 watt resistor is ideal. Melt a small glob of solder onto the end of the wire and quickly solder it to the battery case. This is best done with a 100 watt soldering gun. Make sure the gun is hot before you try to solder the wire on. Soldering should take 1 second.

Have a helper with a wet towel ready to press on the battery as soon as you remove the soldering gun. The insulation between terminals may be thermal plastic and could deform allowing the battery to short if you aren't quick. Cut the soldered lead close to the resistor body and flip the battery over and solder a lead on the other side, making sure that it doesn't touch the positive terminal. Make sure that this lead points 180 degrees away from the other lead so the battery will mount the same way as the original battery. Bend the leads so they will fit into the slots for the original battery. Before you remove the original, note that the positive lead is connected toward the outside of the board. Quickly solder the replacement in the same way. Check the voltage across the battery. If it reads 3 volts, you're all set.

IMPRESSIONS FROM THE CHICAGO TI-FAIRE

THE SHOALS UG

On November 2nd, at Triton College in Chicago was held the now famous TI-Faire. This was the third time the Faire has been held and I understand that each time it has drawn a larger crowd. I was fortunate enough to have been in Chicago on company business just days before the Faire. Due to the kindness of my bosses I was allowed to extend my visit an extra day.

If you have been reading the Chicago newsletter that we get, you will know that the Faire is a big vendor show. all of the major vendors were invited to attend. Most did just that. The only major players I noticed that failed to attend were Cor-Comp, maybe the traveling prohibited them from attending. Anyway, here is a list of those that did attend. Asgard Software, Bytemaster Computer Services, CG drives,Competition Computer Products, Corporate Disk Company, Data Systems, Databionics, DataBiotics, Great Lakes Software, Hunter Electronics, J D Limited, J KH Software, Micro Format, Miller Graphics, Myarc, New Horizons User Group, Techni-Graphics, Thomson Software, TI Forum (Compuserve), Tomputer, and the Will county Users Group.

At this point I have to be honest with you. My major reason for wanting to attend the Faire was not to see what the Vendors were selling. We TIer's are an unusual bunch of folks. We seem to refuse to dry up and all blow away. Here we are entering the the third year since Black Friday, with no big name software or hardware company supporting us, and most of us still happy with our machines and overwhelmed with all of the great new software and hardware to use. The great software and hardware are produced by individuals or very small struggling companies. As such, we tend to know a lot of names of the creators and inovators. If you visit CompuServe upon occasion, or read MICROpendium, or even some of the newsletters from places we trade with, you learn a lot about what these people are trying to do to support the TI community. As such, they seem to become friends without having ever spoken to them or seen them face to face. This face to face visiting was what I had in mind, and I am very happy to say that I had a fantastic time. If you came to the Forum meeting this month then you got to finally see the faces that go along with the names we all seem to know so well.

I arrived at Triton College 15 minutes before the gates opened and paid my $1 to get in. There were over 750 people there when the gates finally did open at 10:30 am. The Faire was held in a room that appeared to be not much larger than a basketball court. Very small, and very crowded. I milled around a while wondering how I was going to find anyone in the sea of faces. Craig Miller was scheduled to speak at 10:30. Since that floor was already so crowded I went upstairs to the fireside room to see what Craig had to say. I was lucky to get up there when I did because that room was filled too. It held roughly 200+ seats and soon standing room only. Craig Miller was one of the few faces that I knew before the Faire. We were lucky enough to see his face on the LA 99ers video tape. One of the other faces on the tape was that of Terrie Masters, President of the LA group. Through the crowd, I saw her face so I knew that she had made it to the Faire. Craig's talk was on the amazing Gram Kracker. It is a device that looks like a long fat widget that plugs into your module port. With it you can down load any rom or grom onto disk. This means that you can now download ANY module to disk. The Gram Kracker also has up to 80K worth of memory that you can use to load modules into. Craig said that it was capable of holding Extended Basic, TI-Writer, and Editor/ Assembler with related files, and still have room for a little more. I brought back a 4 page brochure that describes it in much more detail. The price will be $174 which is expensive but not really more than the price of a normal good piece of hardware.

After Craig's talk, I braved the floor again. I had hoped that it would have cleared out some, but instead it was even more crowded. After a search I finally tracked down Terrie Masters. Terrie is one of the people that I have come to know though our newsletter exchange. I have even had the good fortune to have talked with her on the phone a time or two. Terrie is a very strong vocal TI supporter. She is also a really nice person. She was kind enough to lead me around and introduce me to some of the other TI biggies- Mack McCormic, Howie Rosenburg, and Barry Traver. I chatted with these and other folks and missed lunch and a couple of other Fireside talks. I was also able to track down John Benke (a great tunnels of Doom game writer, and also one of the hard working Chicago Faire workers), Donald Thomson (wrote the great Disk + Aid program), Jim Horn and Johathan Zittrain (CompuServe sysops), Bill Moseid (author of DataBioTics Miniwriter series), and lots of others. I even got to talk with Craig Miller and his wife for a while.

At 3:30 pm the Faire's BIG event was to take place. Myarc was supposed to announce the NEW computer. At 3:00 I returned to the Fireside room and found that no chairs were left. It was already standing room only. By 3:30 there was no

CONT ON PAGE 4

IMPRESSIONS FROM THE CHICAGO TI-FAIRE

CONT FROM PAGE 3

room left at all. Lou Phillips of Myarc began his talk with a short discussion on the company of which he is the president. He then went into a discussion on Myarc's new Extended BASIC Level IV. This module allows you to use the Myarc 128/512K card to write Extended Basic programs. Finally the 32K barrier has been broken. The module also offers Text mode and bit Mapped graphics and is fully compatible with old Extended Basic programs, running most up to 3 time faster. This was great news, however the natives were restless for the new computer to be unveiled. That time did finally arrive amidst hopes and disappointments. The disappointment was that the machine was not ready yet. In fact it is not expected to be ready until the January consumer electronic show. This crushed a lot of hopes because the machine is really vapor-ware until you can actually buy one. However, Lou Phillips overcame at least my disappointment by saying "Myarc is committed to producing the new machine, they have always lived up to their commitments in the past, they wanted to let us know that a new machine IS in the works, and what it will do." These comments made me feel a lot better about the machine. It will come with 256K expandable to many more times that. It is to be about 95% compatible with existing software. It will allow you to use your existing P-Box and all cards except the 32K card. It will use either the TI, CorComp, or Myarc Disk Controller. It will come with the new Extended Basic built in. The compute is expected to sell for about $499. Overall I was very impressed, enough so that I took no notes and have forgotten most of what he said. I am sure that we will get a lot more details before the machine is released in January.

Well, how to sum up this rambling. First, let me say that I was very impressed with what the Chicago group pulled off. They lost count of attendance at 3000+ mark. Second, it was great to go to a computer show that was aimed totally at what I was interested. And finally, I was most impressed with the people that I met. It's not hard at all to see how our dead machine thrives so well with all of the super folks out there that are dedicated to helping it live. Its been said that computers are nothing but cold emotionless inhuman boxes. This is probably true. But I discovered that the TI-99/4A at least, is one small emotionless inhuman box that seems to draw out an unbelievable amount of human friendship and caring. Long may it continue to do so!

John E. Taylor- CIS* 74766,241

DOWNLOADED BY
B.L. MILLER

NLQ FOR THE GEMINI 10X

Johnson Space Center

A new product from Germany has arrived on the American Market and is proving to be a big hit with Gemini 10X owners. It's a plug in chip that allows the 10X to produce Letter Quality Print that rivals the SG-10. The chip has been available in Europe for over a year, so you can be assured that all the bugs have been worked out. I have one on my 10X and couldn't be happier with it's performance.

The NLQ mode can be involked by changing Dip-switch settings or by simple printer commands in your program. I had some samples at the last meeting and everyone who saw them thought they were super. If you missed it, here is a sample of what the chip can do.

The letters "w" and "p" are fabulous. Print is very near the true typewriter. You would be hard pressed to tell the diference. Letters are round, not square. A plus for readability.

The letters are formed during two passes across the paper. Of course, this reduces the print speed to about half. The second pass completes the distenders and emphasizes the print. The print quality is remarkable.

Just about anyone can install it. It takes about 20 minutes. The chip replaces two integrated circuits found on the board right behind the carriage.

The NLQ type face resides where the ITALIC face used to be. In fact, the codes that involked ITALIC print now involk NLQ print. SO GOODBYE ITALIC PRINT. Ihave tried the chip with TI-WRITER and have experienced no problems.

Now you are asking yourself, how much is this chip? The answer is $57.50 each or a group discount is available if we buy several at one time. It becomes much cheaper than $230.00 for a new SG-10.

The NLQ chip is sold by:

E.S.P. CORPORATION
7900 NORTH TAMIAMI TRAIL SARASOTA, FL. 34243 PHONE 813-355-6797

About two years ago, Star Micronics changed one of the chips in the 10X. This change makes it necessary for you to open your printer to determine which chip is needed in yours. Look over the board in your printer. If you find a chip labled D78016176 then you need chip number G10M. If you find a chip labled D78006 then you need NLQ chip number 610. You will have one or the other in your printer. It may sound confusing but, a call to the company will result in instant help. Once your printer has the chip in place, you will be very pleased with the enhancement.

COME ON SG-10, TRY THIS LITTLE 10X ON FOR SIZE.

DISK DRIVES BY JIM NESS

From LEHIGH 99'ER

It's funny (at least to me), but there are lots of people who seem to know lots of stuff about their computers, and all those tiny chips, and how the bits and bytes are handled. And there seems to be next to nobody that knows anything about disk drives, and how they work. Sensing this huge gap in man's knowledge, I decided to figure out what makes them tick.

The great thing about disk drives is that they can find files buried randomly within a huge field of data, and they do it pretty fast. Actually, they can do it so fast because it's not at all random.

The mechanical concept is not all that complicated. A small motor spins at 300 rpm (at least in this country, with its 60 hz power supply), and there is a tiny stepping motor attached to a read-write head. A stepping motor is a common item in indexing applications, where you want a motor to move a precise distance and stop on a dime. The read/write head is just a smaller version of what you have on a cassette recorder.

The stepping motor "steps" the head from track to track on a diskette. The tracks are concentric circles, not a long spiral as you would have on an album.

All of this is ultimately controlled by the disk software with your computer. Usually this is located in ROM within the machine. In most machines, the ROM is only sophisticated enough to load in the official Disk Operating System (DOS) which is located on the disk in the drive when the machine is turned on. The DOS contains all the file handling software, copying software, etc, and because it is on disk, it can be easily modified and/or updated as time goes by.

Our friends at TI decided to put the whole thing in ROM, which has a few bad side-effects. First, it makes it hard to update and improve the software, which is located in the Disk Controller Card. Second, although the machine is a 64k machine, just like all the others, TI has set aside so much memory for special purposes, that there is only 32k left to play with. They set aside 8k for cartridges, 4k for disk drive, 4k for RS232/PIO cards, 4k for the Operating System, (can't complain about that one), and 8k for various interfaces (speech, sound, VDP). Ok those are all good applications to have, but if you don't use them, you still can't use that memory for other things.

Anyway, all of the controlling software for the TI/99/4A is located in the ROM card, as I said. This software tells the step motor when to step to the next track, when to return to the beginning, etc.

There is no standard for how a computer keeps track of data. In the case of TI, there is a directory of existing files, and a map of where they are located, at the beginning of each disk. These files are not necessarily all in complete groups. If you delete a 12 sector file from a disk, there is a 12 sector gap recorded in the map. Then if you add a 20 sector file, the software will put the first 12 sectors in the gap, and put the rest in the first available spot. When you ask for a file that is broken up this way, you can hear the disk head scooting along to read each individual segment.

Because the disk drives themselves are pretty standard, there are a few things that don't change. For instance, there are 48 tracks per inch in most 5 1/4" systems (There is a new 96 TPI system around, not TI compatible). And most systems only use 35 or 40 of the available 48 tracks. There are either 9 or 18 sectors per track (single or double density). Each sector holds 256 bytes of data. And the standard design allows 250,000 bits per second to be written.

Wow, you say, 250k! That is about 25k bytes per second, right? How come I can not load a 25k pgm in one second then? Two reasons. First, as I said, the transfer of data is actually controlled by the ROM software in the TI99/4A. And to be as good as it is ,it had to be a little bit slow. Not REAL slow (anyone ever use a C64 disk drive?), but not as fast as it could be. The second reason also has to do with software, but it is a universal problem associated with single density storage.

The major difference between single and double density storage is the way in which the data is coded. In order for the software to keep track of where the read head is located on a particular track, there are clock or synch bits laid down with the data bits. in the old fashioned single density format, a synchbit was laid down ahead of each "0" bit, so there were never two "0" bits in a row. That kept the software from getting lost if there were a lot of "0" bits in series. Putting all those synch bits on the disk took up a tremendous amount of space that should be used for data.

So, some genius came up with a way of encoding the clock bits in with the data bits, so that no unnecessary space was lost. Wala, double density storage was born! And double density, as used with the CorComp software, is said to increase transfer speed by at least 80%, mostly because the number of bits to transfer is cut way down.

So much for the exciting story of double density versus single density. How about double sided versus single sided? Well, obviously, it requires two read/write heads in the drive. Did you know that when reading a disk, the software reads, first, a track from side one, then the opposing track from side two, and continues back and forth?

The disk head needs something to keep the disk stationary against it. In a single

MUSICAL NOTES

In the December issue of the Manasota Users Group were the following programs. I've added a little extra from a June '84 article, by Jim Hubbard from our newsletter to make the music more interesting to those of you who wish to do some music programing.

HERE ARE TWO MORE INTERESTING EXAMPLES OF PROGRAMMING MUSIC ON YOUR TI.....

The first program sets values for all of the frequencies you are going to use in the song before the call sound lines. The X,Y,Z values allow changes in duration or timing.

(Ed's note: the second song is the same as the first except for the CALL SOUND statement, this organ type sound is from Jim's article. For a pleasant Tremolo sound you might try:

```
)XXX FOR J=1 TO 8
)XXX CALL SOUND(-50,N,1)
)XXX CALL SOUND(-50,N*1.3,1)
)XXX NEXT J
```

If anyone is interested in have this article by Jim reproduced in a future issue, please let me or any club officer know.)

The last song is from the TOPICS LA 99ERS and was programed by Brett Fijan and Chick De Marti. It combines spaceship graphics with an appropriate sound generator which you can regulate with a speed input. It is fun to play around with it and see if you understand the programming well enough to make your own modifications of color and sound.

Perhaps this discussion of TI's sound capabilities will serve to remind some users of the fun you can have just messing around with your TI - not to mention the fine music teaching possibilities. If that is an interest you have, you should look up programs written by Regena in her Programmer's Reference Guide.

```
100 X=200
110 Y=400
120 Z=600
130 REM DEFINE MUSICAL NOTES

140 A1=110
150 B1=123
160 C1=131
170 CS1=139
180 D1=147
190 DS1=156
200 E1=165
210 F1=175
220 FS1=185
230 G1=196
240 GS1=208
250 A2=220
260 AS2=233
270 B2=247
280 C2=262
290 CS2=276
300 D2=294
310 DS2=311
320 E2=330
330 F2=349
340 FS2=370
350 G2=392
360 GS2=415
370 A3=440
380 AS3=466
390 B3=494
400 C3=523
410 D3=587
420 E3=659
430 F3=698
440 A4=880
450 CALL SOUND(Y,B2,0)
460 CALL SOUND(Y,C2,0)
470 CALL SOUND(Y*2,D2,0)
480 CALL SOUND(Y*2,B3,0)
490 CALL SOUND(Y*2,G2,0)
500 CALL SOUND(Y,A3,0)
510 CALL SOUND(Y,G2,0)
520 CALL SOUND(Y*2,G2,0)
530 CALL SOUND(Y*2,FS2,0)
540 CALL SOUND(Y*2,FS2,0)
```

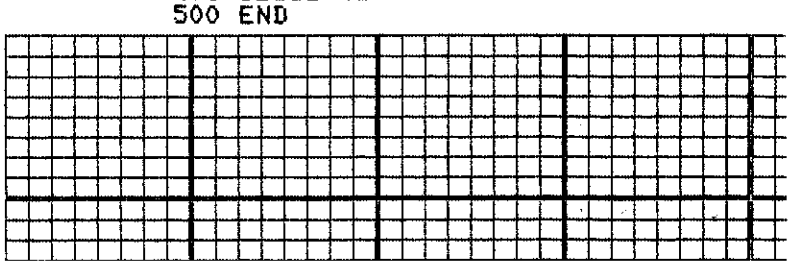
```
100 X=200
110 Y=400
120 Z=600
130 REM DEFINE MUSICAL NOTES

140 A1=110
150 B1=123
160 C1=131
170 CS1=139
180 D1=147
190 DS1=156
200 E1=165
210 F1=175
220 FS1=185
230 G1=196
240 GS1=208
250 A2=220
260 AS2=233
270 B2=247
280 C2=262
290 CS2=276
300 D2=294
310 DS2=311
320 E2=330
330 F2=349
340 FS2=370
350 G2=392
360 GS2=415
370 A3=440
380 AS3=466
390 B3=494
400 C3=523
410 D3=587
420 E3=659
430 F3=698
440 A4=880
450 CALL SOUND(Y,B2,0,B2*2,0
,B2*7.5,30,-4,V)
460 CALL SOUND(Y,C2,0,C2*2,0
,C2*7.5,30,-4,V)
470 CALL SOUND(Y*2,D2,0,D2*2
,0,D2*7.5,30,-4,V)
480 CALL SOUND(Y*2,B3,0,B3*2
,0,B3*7.5,30,-4,V)
490 CALL SOUND(Y*2,G2,0,G2*2
,0,G2*7.5,30,-4,V)
500 CALL SOUND(Y,A3,0,A3*2,0
,A3*7.5,30,-4,V)
510 CALL SOUND(Y,G2,0,G2*2,0
,G2*7.5,30,-4,V)
520 CALL SOUND(Y*2,G2,0,G2*2
,0,G2*7.5,30,-4,V)
530 CALL SOUND(Y*2,FS2,0,FS2
*2,0,FS2*7.5,30,-4,V)
540 CALL SOUND(Y*2,FS2,0,FS2
*2,0,FS2*7.5,30,-4,V)
```

```
100 FOR I=96 TO 128 STEP 8
110 CALL CHAR(I,"FFFFFFFFFFF
FFFFF")
120 NEXT I
130 CALL CLEAR
140 CALL SCREEN(2)
150 RESTORE
160 DATA 10,2,96,6,10,8,104,
6,10,14,112,6,10,20,120,6,10
,26,128,6
170 DATA 11,2,96,6,11,8,104,
6,11,14,112,6,11,20,120,6,11
,26,128,6
180 DATA 12,2,96,6,12,8,104,
6,12,14,112,6,12,20,120,6,12
,26,128,6
190 DATA 13,2,96,6,13,8,104,
6,13,14,112,6,13,20,120,6,13
,26,128,6
200 FOR I=1 TO 20
210 READ A,B,C,D
220 CALL HCHAR(A,B,C,D)
230 NEXT I
240 CALL SCREEN(5)
250 FOR I=9 TO 13
260 CALL COLOR(I,2,1)
270 NEXT I
280 INPUT "SPEED? ":S
290 CALL HCHAR(23,1,32,32)
300 CALL SCREEN(2)
310 FOR C=5 TO 8
320 CALL COLOR(C,16,5)
330 DISPLAY AT(5,2):"FLYING
SAUCER MUSIC PLEASE"
340 NEXT C
350 CALL COLOR(9,9,1)
360 CALL SOUND(S,330,0,330.5
,0)
```

```
370 CALL COLOR(9,2,1)
380 CALL COLOR(10,12,1)
390 CALL SOUND(S,370,0,370.5
,0)
400 CALL COLOR(10,2,1)
410 CALL COLOR(12,16,1)
420 CALL SOUND(S,294,0,294.5
,0)
430 CALL COLOR(11,2,1)
440 CALL COLOR(12,16,1)
450 CALL SOUND(S,147,0,147.5
,0)
460 CALL COLOR(12,2,1)
470 CALL COLOR(13,6,1)
480 CALL SOUND(S,220,0,220.5
,0)
490 CALL COLOR(13,2,1)
500 S=S-25
510 IF S<1 THEN 240
520 GOTO 350
```

```
100 REM +-------------------+
110 REM + Graphsheet Maker +
120 REM +  By John Behnke  +
130 REM +                  +
140 REM + Epson or Gemini  +
150 REM + Printer Required +
160 REM + Basic or X-Basic +
170 REM +-------------------+
180 CALL CLEAR
190 INPUT "NUMBER OF SHEETS?
 ":A
200 REM CALL SCREEN(2)
210 @$=CHR$(27)
220 FOR I=1 TO 228
230 A$=A$&CHR$(128)
240 NEXT I
250 B$=SEG$(A$,1,7)
260 C$=CHR$(255)&SEG$(A$,1,6
)
270 FOR I=1 TO 4
280 FOR J=1 TO 8
290 E$=E$&C$
300 NEXT J
310 E$=E$&CHR$(255)
320 NEXT I
330 F$=@$&"K"&CHR$(484)&CHR$
(0)&E$
340 G$=@$&"K"&CHR$(228)&CHR$
(0)&A$
350 OPEN #1:"PIO.CR"
360 FOR B=1 TO A
370 FOR C=1 TO 11
380 PRINT #1:@$&CHR$(64)&@$&
"3"&CHR$(16)
390 FOR D=1 TO 8
400 PRINT #1:F$;F$;CHR$(10)
410 NEXT D
420 PRINT #1:G$;G$;@$&"3"&CH
R$(2)
430 NEXT C
440 PRINT #1:@$&"3"&CHR$(17)
450 FOR I=1 TO 9
460 PRINT #1:CHR$(13)&CHR$(1
0)
470 NEXT I
480 NEXT B
490 CLOSE #1
500 END
```



PRINT USING

USING "PRINT USING" WITH YOUR PRINTER
by Richard Roberts
MSP 99 NEWSLETTER

Reprinted from "Topics", the monthly newsletter of the LA 99ers. Richard Roberts is President of the Dallas TI99/4A Users Group.

One of the more obscure statements available with TI Extended BASIC is one called PRINT USING. Even more obscure is the fact that this statement can be used to format variables and constants that will be dumped to your printer. On page 150 of the Extended BASIC manual, several examples of how PRINT USING can be used to format data for screen display are shown, but nary a word of how to do the same with open files. It can be done, and is much more powerful than you may realize.

Any discussion of PRINT USING will require an understanding of the IMAGE statement, so if you are not familiar with it, you'd better brush up on it first. The PRINT USING statement uses IMAGE in one of two ways, either with a string expression, or a line number reference. I prefer the latter, as it alows for more flexibility, but since these different methods are explained in the manual, I will limit this to a few simple examples that are not shown in the manual.

```
100 TCOST=19.55
110 IMAGE ##.##
120 OPEN #1:"PIO"
130 PRINT #1,USING 110:TCOST
```

Running this sample program will effectively show how the PRINT USING statement will work with an open file. Of course, there are many other variations of IMAGE that can be used, so experiment with them and watch how it performs when line 130 dumps it to the printer. Shown below are a few more examples for use with an open file.

```
110 IMAGE "##.## ##.##"
130 PRINT #1,USING 110:COST1,COST2
```

This IMAGE statement will allow you to print two (or more) variables at a predetermined spot on the same line. The length of the string expression in the IMAGE statement can be as long as you wish, up to the limit of an Extended BASIC line.

```
110 IMAGE "########## ##.##"
130 PRINT #1,USING 110:"TOTAL COST",TCOST
```

This version shows how you can format the printed line for string data as well as numerical data. A string variable could be used in place of the string constant, as below.

```
105 A$="TOTAL COST"
110 IMAGE "########## ##.##"
130 PRINT #1,USING 110:A$,TCOST
```

It is also possible to place the IMAGE statement inside in PRINT USING statement, as shown below.

CONT ON PAGE 8

DISK DRIVES: CONT FROM PAGE 5

sided drive, there is a small arm holding the back side of the disk against the head. in a double sided drive, that arm would be in the way of the back side read/write head, so the solution was to use two heads,directly across from one another, to hold the disk in place. In order to keep them across from one another, they alternate reading or writing as I said above. Very interesting, right? So if you wreck one side of a dbl sided disk, you can kiss the whole thing goodbye.

Downloaded by B.L. Miller

PRINT USING: CONT FROM PAGE 7

First, Delete line 110.

```
130 PRINT #1,USING "##.##":TCOST
or
130    PRINT    #1,USING    "##############
##.##":A$,TCOST
```

A few other points to remember include the fact that IMAGE and PRINT USING can be used to round off calculated variables. A single string expression such as "#######.##" will round off and align decimal numbers as small or as large as needed and place a number at any designated location.

This function could save many hours of algorithm development for accomplishing the same thing. So, in the long run, the PRINT USING statement is one that any programmer should be very familiar with, and use as much as possible.

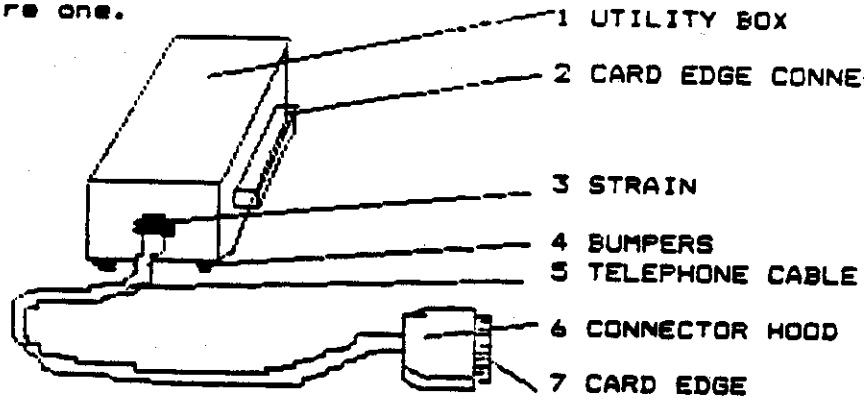
CABLE BOX

by Jim Edwards (SFV 99ers)

One feature of the T.I.99 that has never been hard for me to criticize was the physical size and design of the peripheral cable and connector. It always seemed to take up an undeserved portion of desk space. With only a goal in mind and virtually no "hardware saave", I set out to alleviate the problem. It seemed a simple task to build a compact connector that would plug in without disturbing the original components. Actually, the most difficult aspect of the project was rounding up the parts.

That proved to be an education. Card edges and their matching connectors have several configurations. For example 22/44 means that it has 22 conductors on both sides. Spacings vary as well: .10, .125, .156, etc. This refers to the distance between the centers of the conductors. This project requires 44 conductors (22 on a side) with .10 centers. Finding a card edge connector was difficult enough, but finding the male counterpart was impossible. A section was literally cut out of an abandoned board.

I found most of the parts at Pacific Radio while the card was found in a card board box at All Electronics. Obviously, the exact parts may vary but be certain of the number of conductors and spacing. Once everything is rounded up, simply solder the wires together making sure to match one end to the other. Optionally, an interupt switch can be added for those screen dump programs that require one.



#	PART	MANUFACTURER	PT.#	COST
1	Utility Box	CALRAD	90-785	$2.10
2	Edge Card Connector	GC Electronics	41-875	$4.74
3	Strain			.25
4	1/4" Bumpers	RUSSELL IND.	REC-207SH	$1.79
5	50 Conductor Telephone Cable			
6	Connector Hood	GC Electronics	41-1003	$2.48
7	Edge Card Scavanged From PC Board.			$1.50
				$12.84

ADDING A LOAD INTERRUPT SWITCH TO THE SPEECH SYNTHESIZER

Richard J. Bailey
68A Church Street
Gonic, N.H. 03867
NH99'ERS USERS GROUP

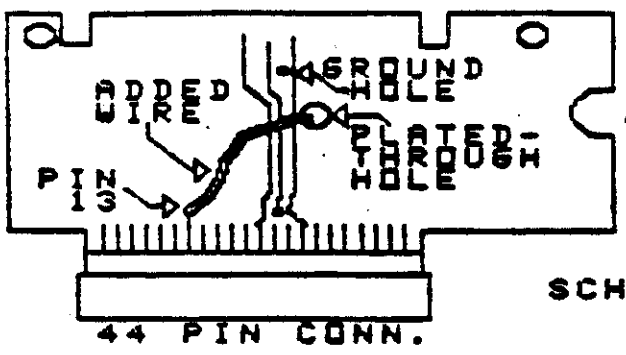
A number of people have asked me about the load interrupt switch I had added to my speech synthesizer to allow dumping screens from the various cartridges using the excellent screendump program that was written by Danny Michaels. So here are instructions to allow you to modify your own synthesizer to accomplish this.

Keep in mind that you have to know enough about electronics to add the parts needed for the modification without messing up your synthesizer. I have made the modification to my own synthesizer so I know that it works, but if you mess up, then you're out a synthesizer. You could add the same parts inside the console and have a small switch sticking out the back if you want the modification self-contained or don't have a speech synthesizer.

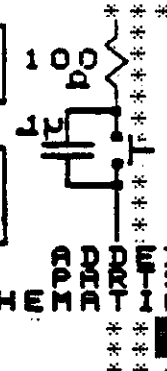
The only part really needed is a minature pushbutton switch with normally open contacts, but if you add a 100-500 ohm resistor in series with the switch and a .01-.1 MFD capacitor across the switch, there will be less chance for contact bounce (If you really want bounce-free contact enclosure use cross-coupled gates as an R-S flip-floop). The added parts syschematic and location diagram of the speech synthesizer board is shown above. These were drawn with GRAPHX.

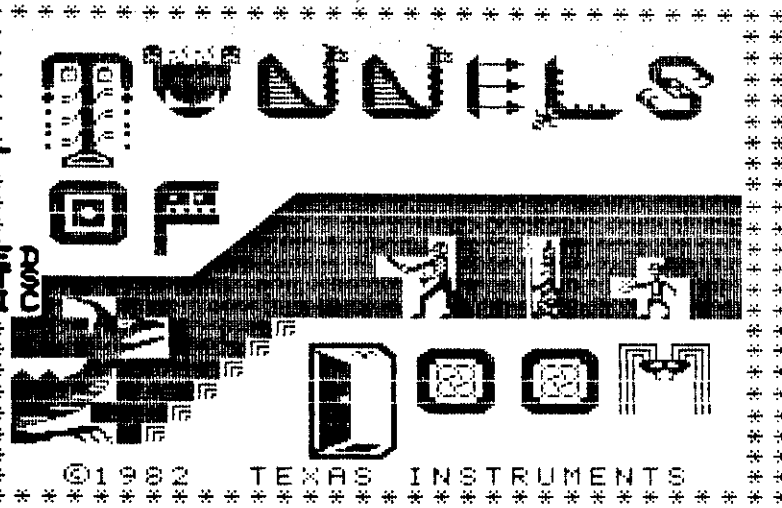
To modify your unit, do the following:
1) Buy the parts. The switch must not stick >1/4 inch beyond threads.
2) Dismantle synthesizer. Note how shield slides together.
3) Clear large plated-through hole of soder.
4) Solder 2 1/2 inch piece of wire to pin 13 of 44 pin connector.
ALL OTHER PARTS GO ON TOP SIDE OF CIRCUIT BOARD
5) Solder one end of 100 ohm resistor in ground hole.
6) Solder 1 1/2 inch piece of wire to other end of resistor.
7) Solder wire to swich and .1 MFD capacitor.
8) Drill hole in middle top of shield for switch.
9) Mount switch making sure everything fits.
10) Reassemble unit making sure nothing shorts.

You can now follow the instructions for the screendump program to check the operation of the switch. You may find other interesting uses of the switch, if you do, please pass them on to the newsletter.



BOTTOM VIEW OF SPEECH SYNTHESIZER BOARD





ANOTHER USE OF THE LOAD INTERRUPT SWITCH,
by Gary Brown in the J*U*G*S Newsletter

Assemble this source code:

```
AORG >FFFC
DATA >A000,B0BE END
```

Load SBUG
Load the interrupt program you have assembled above.
Press Function =(Quit)
Do not turn console or Pbox off.
Insert a cartridge and start it running.
Press the interrupt switch and you will have the SBUG title screen.

Follow the instructions and go into addresses 6000 thru 8000 find the program in these addresses. Then you can disassemble these addresses.

Remember no one assumes any liability for your computer should you try any of these modifications.

```
100 CALL CLEAR :: DISPLAY AT
(5,7):"HYPNOSIS PATTERN":: :
:"this will blow your mind"
:: FOR T=1 TO 300 :: NEXT T
:: CALL VCHAR(1,1,32,768)
110 CALL SCREEN(2):: CALL CO
LOR(1,2,2):: CALL HCHAR(10,1
,33,160):: CALL HCHAR(12,15,
32,2):: CALL COLOR(1,2,16)
120 CALL CHAR(32,"F0F0F0F00F
0F0F0F0F0F0F0F0F0F0F0"):::
 CALL CHAR(33,"0F0F0F0FF0F0F
0F0F0F0F0F0F00F0F0F0F"):: GO
TO 120
```

TIPS FROM THE TIGERCUB

#29

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

Over 130 original programs in Basic and Extended Basic, available on casette or disk, only $3.00 each plus $1.50 per order for PPM. Entertainment, education, programmer's utilities.
Descriptive catalog $1.00, deductable from your first order.
Tips from The Tigercub, a full disk containing the complete contents of this newsletter Nos. 1 through 14, 50 original programs and files, just $15 postpaid.
Tips from the Tigercub Vol. 2, another diskfull, complete contents of Nos. 15 through 24, over 60 files and programs, also just $15 postpaid. Or, both for $27 postpaid.
Nuts & Bolts (No. 1), a full disk of 100 Extended Basic utility subprograms in merge format, ready to merge into your own programs. Plus the Tigercub Menuloader, a tutorial on using subprograms, and 5 pages of documentation with an example of the use of each subprogram. All for just $19.95 postpaid.
Nuts & Bolts No. 2, another full disk of 100 utility subprograms in merge format, all new and fully compatible with the last, and with 10 pages of documentation and examples. Also $19.95 postpaid, or both Nuts Bolts disks for $37 postpaid.
Tigercub Full Disk Collections, just $12 postpaid! Each of these contains either 5 or 6 of my regular $3 catalog programs, and the remaining disk space has been filled with some of the best public domain programs of the same category. I am NOT selling public domain programs - my own programs on these disks are greatly discounted from their usual price, and the public domain is a FREE bonus!

TIGERCUB'S BEST
PROGRAMMING TUTOR
PROGRAMMER'S UTILITIES
BRAIN GAMES
BRAIN TEASERS
BRAIN BUSTERS!
MANEUVERING GAMES
ACTION GAMES
REFLEX AND CONCENTRATION
TWO-PLAYER GAMES
KID'S GAMES
MORE GAMES
WORD GAMES
ELEMENTARY MATH
MIDDLE/HIGH SCHOOL MATH
VOCABULARY AND READING
MUSICAL EDUCATION
KALEIDOSCOPES AND DISPLAYS

For descriptions of these send a dollar for my catalog!
The offer made last month is still good until 1 January - a 10% rebate directly to the user group if one of their members mentions the user group when ordering from me. So far, I've had only 10 responses - and I suspect that 8 or 9 of those didn't even know about the offer!

I goofed again. In the I/O ERROR routine in Tips #28, the ON ERROR STOP will do no good in the place where I put it. It should be placed after the file is opened in line 100 so that it will become the current error trap if the file is opened correctly.

And the CALL KEY example in Tips #28 will look better if R=14. A couple of very knowledgeable programmers have written to tell me that I was wrong, and the manual is right, about CALL KEY status -1. They say that -1 simply means that the same key is being pressed as was pressed during the last keyscan, and that it could have been released and repressed in the interim. This may be, but try this routine and see if you can release and repress a key without getting a status code 0 (no key pressed) and status code 1 (different key pressed) before another status code -1.

```
100 CALL KEY(0,K,S):: PRINT
K,S :: GOTO 100
```

George Steffen has responded to the challenge in the last Tips, by publishing in the LA 99ers Topics a remarkably compact routine to translate the internal format string representation of numeric data back into numbers. The following lines will update the Menu Loader accordingly.

```
100 !by A. Kludge/M. Gordon/
T. Boisseau/J. Peterson/G. S
teffen/etc.Version 80, 11/85
140 @,@@,A,A$,B,C,D$,E,F,FLA
G,I,J,K,KD,KK,M,M$,N$,NN,P,P
$,PG$(),PP,PP$,Q$,S,ST,T$(),
TT,VT,V(,),W$,X,X$,Y,K2,S2
810 F=1 :: E=ASC(SEG$(M$,1,1
)):: M=ASC(SEG$(M$,2,1)):: I
F E=0 AND M=0 THEN GOTO 817
ELSE IF E>128 AND M>128 THEN
 F=-1 :: E=255-E :: M=256-M
815 FOR I=1 TO 6 :: M=M+(ASC
(SEG$(M$,I+2,1)))/100^I :: N
EXT I :: M=M*F*100^(E-64)
817 PRINT #PP:M
870 FOR P=1 TO NN-1 :: PRINT
 #2:PG$(P);TAB(15);V(P,3);TA
B(20);T$(ABS(V(P,1)));TAB(25
);V(P,2);TAB(31);CHR$(89*ABS
(V(P,1)<0)):: NEXT P :: CLOS
E #2
```

The change in the last line is my own, because it was pointed out to me that the catalog output to the printer did not indicate protected files.
That last line is a good example of the power of relational expressions to accomplish compact programming. The variable V(P,1) picks up its value from the variable A which is read from the disk directory in line 350. This is a number from 1 to 5, indicating the type of file, and if the file is write-protected the number is negative. A true expression has a relational value of -1. If the file is protected, V(P,1)<0 is true, and its value is -1, converted by ABS to +1 and multiplied by 89 to give ASCII 89, converted by CHR$ to "Y". If not protected, V(P,1) is a positive number, V(P,1)<0 is false and has a relational value of 0; 89 times 0 is still 0, and CHR$(0) prints nothing.

George also mentioned in a letter that my remarks on the UPDATE mode applied only to VARIABLE files; that RESTORE without a number,to return the record pointer to the beginning of a file, works only with VARIABLE files; that RESTORE with a number works only with RELATIVE files; and that therefore the only way to RESTORE a SEQUENTIAL FIXED file is to close it and reopen it.
On trying this out, I find that you can write to a FIXED SEQUENTIAL file and still be able to read the following records - but you can't simply "read a record, change it in some way, and then write the altered record back out on the file", as the Reference Guide indicates, because you will change the record FOLLOWING the one you read! It is possible to UPDATE a FIXED SEQUENTIAL file without reading it all into an array and writing it back out, but you must read sequentially to the record you want, close the file, reopen the file, read back to the record just before the one you want to update, then write in the updated record.

I have received several other suggestions regarding the Menu Loader, too many to describe here. You can all modify it to your own tastes and needs. Remember to turn off the pre-scan and ON ERROR while you're working on it, then add any new variable names or CALLs to the pre-scan. And remember, that last line MUST be the LAST line of the program! You can resequence it higher, and change the GOTO accordingly, but don't put anything after it!
I did change my version to slash the zero, since this will carry over into a program that is loaded. If you do this, be sure to add a CALL CHAR to the list in line 150!

```
190 CALL CLEAR :: FOR S=1 TO
 14 :: CALL COLOR(S,7,16):: 
NEXT S :: CALL COLOR(0,2,16)
:: CALL CHAR(48,"003A444C546
44400")
```

When you just want to load a program, waiting for it to be read from the disk directory can be a drag. And, you may have trouble recognizing the filename. So, here is the Tigercub Quickloader which I have placed on all my Collection Disks.
First you will need Catwriter, another program that writes a program. This

one will read the disk directory, ignore everything other than programs, ask you for a complete program name for each filename, and write all that into a MERGE format program called CATMERGE.

```
100 !CATWRITER by Jim Peters
on
110 OPEN #1:"DSK1.",INPUT ,R
ELATIVE,INTERNAL :: INPUT #1
:N$,A,J,K :: OPEN #2:"DSK1.C
ATMERGE",VARIABLE 163 :: LN=
1000 :: FN=1100
120 X=X+1 :: INPUT #1:P$,A,J
,B :: IF LEN(P$)=0 THEN 160
 :: IF ABS(A)=5 OR ABS(A)=4 A
ND B=254 THEN 130 ELSE X=X-1
 :: GOTO 120
130 DISPLAY AT(12,1)ERASE AL
L:P$;"    PROGRAM NAME?" ::
ACCEPT AT(14,1)SIZE(25):F$
140 PRINT #2:CHR$(INT(FN/256
))&CHR$(FN-256*INT(FN/256))&
CHR$(147)&CHR$(200)&CHR$(LEN
(F$))&F$&CHR$(0):: FN=FN+1
150 M$=M$&CHR$(200)&CHR$(LEN
(P$))&P$&CHR$(179):: IF X<11
 THEN 120
160 IF M$="" THEN 180
170 PRINT #2:CHR$(INT(LN/256
))&CHR$(LN-256*INT(LN/256))&
CHR$(147)&SEG$(M$,1,LEN(M$)-
1)&CHR$(0):: LN=LN+1 :: M$=
"" :: X=0 :: IF LEN(P$)<>0 TH
EN 120
180 PRINT #2:CHR$(INT(LN/256
))&CHR$(LN-256*INT(LN/256))&
CHR$(147)&CHR$(200)&CHR$(3)&
"END"&CHR$(0)
190 PRINT #2:CHR$(255)&CHR$(
255):: CLOSE #1 :: CLOSE #2
```

Next, key in the Quickloader. Do not change the line numbers, do not RESequence, because CATMERGE will be merged into the middle of it and that last line must be the last. Then, enter MERGE DSK1.CATMERGE and then SAVE DSK1.LOAD .

```
100 CALL CLEAR :: DIM M$(48)
:: CALL CHAR(94,"3C4299A1A19
9423C"):: CALL SCREEN(2):: F
OR SET=1 TO 14 :: CALL COLOR
(SET,15,1):: NEXT SET :: DIS
PLAY AT(1,4):"TIGERCUB QUICK
LOADER"
110 X=X+1 :: READ M$(X):: IF
 M$(X)<>"END" THEN 110
115 CALL PEEK(8198,A):: IF A
<>170 THEN CALL INIT
120 R=3 :: FOR J=1 TO X-1 ::
 READ X$ :: DISPLAY AT(R,1):
STR$(J);TAB(4);X$ :: R=R+1 :
 : IF R<23 THEN 150
130 DISPLAY AT(24,1):"CHOICE
? OR 0 TO CONTINUE 0" :: ACC
EPT AT(24,26)VALIDATE(DIGIT)
SIZE(-2):N
140 IF N<>0 THEN 155 :: R=3
150 NEXT J :: DISPLAY AT(24,
1):"CHOICE?" :: ACCEPT AT(24
,9)VALIDATE(DIGIT):N
160 IF SEG$(M$(N),LEN(M$(N))
,1)="*" THEN DISPLAY AT(12,1
)ERASE ALL:"Return to BASIC"
 : :"Type OLD DSK1."&M$(N):: 
STOP
170 CALL CHARSET :: CALL CLE
AR :: CALL SCREEN(8):: CALL
PEEK(-31952,A,B):: CALL PEEK
(A*256+B-65534,A,B):: C=A*25
6+B-65534 :: A$="DSK1."&M$(N
):: CALL LOAD(C,LEN(A$))
180 FOR J=1 TO LEN(A$):: CAL
L LOAD(C+J,ASC(SEG$(A$,J,1))
):: NEXT J :: CALL LOAD(C+J,
0):: GOTO 30000
30000 RUN "DSK1.1234567890"
```

If you don't want to give your Basic-only programs a filename ending in an asterisk, you can leave out that warning routine, or you can modify it to warn of E/A or MiniMemory programs. If Catwriter has picked up any unloadable program-format files, etc., just delete them from the DATA lines.

The first issue of the GENIAL TRAVELER has arrived, and it is SUPERB! This is a magazine-on-a-disk, a SS/SD flippy loaded with 700 sectors of some of the finest articles and programs you'll ever see! And the programs are ready to run, you don't have to key anything in. The subscription price, until the end of 1985 at least, is $30 for 6 issues, which computes out to $5 per disk - many of you are paying your own user group that much for a one-sided disk of public domain!
If the subscribers will only have the guts to refuse to let their friends copy this for free, this venture will surely survive and contribute greatly to the advancement of the TI. The address is -
GENIAL COMPUTERWARE, 835 Green Valley Drive, Philadelphia PA 19128.

Gene Burchfield asked if I had a program to print banners vertically. I had never heard of such a thing, so I wrote one.

```
100 DISPLAY AT(12,1)ERASE AL
L:"TIGERCUB STREAMER PRINTER
" !by Jim Peterson
110 DATA 0000,0001,0010,0011
,0100,0101,0110,0111,1000,10
01,1010,1011,1100,1101,1110,
1111
120 RESTORE 110 :: DIM B$(16
):: FOR J=1 TO 16 :: READ B$
(J):: NEXT J :: P$(0)=" " ::
 P$(1)=CHR$(238)
130 INPUT "TEXT TO BE PRINTE
D? ":T$ :: PRINT :: INPUT "P
RINTER DESIGNATION? ":PD$ ::
 OPEN #1:PD$
140 PRINT :: INPUT "SIZE? (1
-10) ":Z :: IF Z<1 OR Z>10 T
HEN 140
150 FOR J=1 TO LEN(T$):: A=A
SC(SEG$(T$,J,1)):: IF A=32 T
HEN GOTO 200
160 CALL CHARPAT(A,H$):: FOR
 N=1 TO 15 STEP 2 :: K$=SEG$
(H$,N,2):: FOR L=1 TO 2 :: L
$=SEG$(K$,L,1):: B=POS("0123
456789ABCDEF",L$,1)
170 M$=B$(B):: FOR M=1 TO 4
:: N=VAL(SEG$(M$,M,1)):: N$=
N$&RPT$(P$(N),Z):: NEXT M
180 NEXT L :: FOR Q=1 TO Z/2
+.5 :: PRINT #1:TAB((81-Z*8)
/2+.5);N$ :: NEXT Q :: N$=""
 :: NEXT W :: FOR R=1 TO Z/2
+.5 :: PRINT #1:"" :: NEXT R
190 NEXT J :: STOP
200 FOR T=1 TO Z*4 :: PRINT
#1:"" :: NEXT T :: GOTO 190
210 CALL KEY(0,K,S):: IF S=0
 THEN 210 ELSE RETURN
```

If your printer doesn't have the special characters of the Gemini, substitute 88 instead of 238 in line 120, to print X's, or whatever else you want. If you do have the special characters, try some others, such as 239, for this and other graphics printing programs. This routine will print a handy reference chart of them.

```
100 IMAGE ### #   ### #   ##
# #   ### #   ### #   ### #
110 P$=RPT$(CHR$(251)&CHR$(2
53),21):: X=0
120 OPEN #1:"PIO" :: PRINT #
1:CHR$(27);"E"
130 PRINT #1:P$;" ASCII COD
ES FOR GEMINI SPECIAL CHARAC
TERS":P$
140 FOR J=160 TO 175 :: K=J-
X
150 PRINT #1,USING 100:K,CHR
$(J),K+16,CHR$(J+16),K+32,CH
R$(J+32),K+48,CHR$(J+48),K+6
4,CHR$(J+64),K+80,CHR$(J+80)
:: NEXT J
160 IF FLAG=1 THEN STOP ELSE
 FLAG=1 :: PRINT #1:"":"":P$
:"TI-WRITER CODES FOR GEMINI
 SPECIAL CHARACTERS":P$ :: X
=128 :: GOTO 140
```

Another one that just looks pretty -

```
100 !KALEIDOSPRITES by Jim P
eterson
110 CALL CLEAR :: FOR CH=100
 TO 128 STEP 4 :: FOR L=1 TO
 4 :: RANDOMIZE :: X$=SEG$("
0018243C425A667E8199A5BDC3DB
E7FF",INT(16*RND+1)*2-1,2)
120 B$=B$&X$ :: C$=X$&C$ ::
NEXT L :: CALL CHAR(CH,RPT$(
B$&C$,4)):: B$,C$="" :: NEXT
 CH :: Z=2 :: CALL SCREEN(5)
130 CALL MAGNIFY(Z):: K=1 ::
 FOR J=1 TO 7 :: S=96+4*J ::
 R=16*J :: C=100*RND+20
140 IF J>5 AND Z=4 THEN T=5
:: GOTO 160
150 T=INT(15*RND+2):: IF T=5
 THEN 150
160 CALL SPRITE(#K,S,T,R,C,#
K+1,S,T,177-R,C,#K+2,S,T,K,2
41-C,#K+3,S,T,177-R,241-C)::
 K=K+4 :: NEXT J
170 Z=INT(2*RND+1)*2 :: GOTO
 130
```

```
100 !DISK MATCHER by Jim Pet
erson
110 DISPLAY AT(8,9)ERASE ALL
:"DISK MATCHER": : : :" To c
ompare a backup disk":"with
a master and list any":"file
s found on one but not"
120 DISPLAY AT(15,1):"on the
 other.": : : :"       Press
 any key"
130 CALL KEY(0,K,S):: IF S=0
 THEN 130
140 DISPLAY AT(12,1)ERASE AL
L:"INSERT MASTER - PRESS ENT
ER" :: CALL KEY(0,K,S):: IF
S=0 THEN 140
150 OPEN #1:"DSK1.",INPUT ,R
ELATIVE,INTERNAL :: INPUT #1
:D1$,A,J,K :: DIM F1$(127)
160 X=X+1 :: INPUT #1:F1$(X)
,A,J,B :: IF LEN(F1$(X))<>0
THEN 160 ELSE CLOSE #1
170 DISPLAY AT(12,1)ERASE AL
L:"INSERT BACKUP DISK": :"PR
ESS ENTER" :: CALL KEY(0,K,S
):: IF S=0 THEN 170
180 OPEN #1:"DSK1.",INPUT ,R
ELATIVE,INTERNAL :: INPUT #1
:D2$,A,J,K :: DIM F2$(127)
190 Y=Y+1 :: INPUT #1:F2$(Y)
,A,J,B :: IF LEN(F2$(Y))<>0
THEN 190 ELSE CLOSE #1
200 DIM F(127):: FOR J=1 TO
X :: FOR L=1 TO Y :: IF F2$(
L)=F1$(J)THEN F(L)=1 :: GOTO
 220
210 NEXT L :: PRINT F1$(J);"
 NOT ON BACKUP"
220 NEXT J
230 FOR M=1 TO Y :: IF F(M)=
0 THEN PRINT F2$(M);" NOT ON
 MASTER"
240 NEXT M :: END
```

A very useful tip from Jim Swedlow, in the Orange County ROM newsletter - INPUT respects any trailing print separator on a preceding PRINT command. Try it -

```
100 PRINT TAB(20);:: INPUT B
$
```

MEMORY FULL IN LINE 480
Jim Peterson

TIPS FROM THE TIGERCUB

#3#

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

Over 13# original programs in Basic and Extended Basic, available on cassette or disk, only $3.## each plus $1.5# per order for PPM. Entertainment, education, programmer's utilities. Descriptive catalog $1.##, deductable from your first order.

Tips from The Tigercub, a full disk containing the complete contents of this newsletter Nos. 1 through 14, 5# original programs and files, just $15 postpaid.

Tips from the Tigercub Vol. 2, another diskfull, complete contents of Nos. 15 through 24, over 6# files and programs, also just $15 postpaid. Or, both for $27 postpaid.

Nuts & Bolts (No. 1), a full disk of 1## Extended Basic utility subprograms in merge format, ready to merge into your own programs. Plus the Tigercub Menuloader, a tutorial on using subprograms, and 5 pages of documentation with an example of the use of each subprogram. All for just $19.95 postpaid.

Nuts & Bolts No. 2, another full disk of 1## utility subprograms in merge format, all new and fully compatible with the last, and with 1# pages of documentation and examples. Also $19.95 postpaid, or both Nuts Bolts disks for $37 postpaid.

Tigercub Full Disk Collections, just $12 postpaid! Each of these contains either 5 or 6 of my regular $3 catalog programs, and the remaining disk space has been filled with some of the best public domain programs of the same category. I am NOT selling public domain programs - my own programs on these disks are greatly discounted from their usual price, and the public domain is a FREE bonus!

TIGERCUB'S BEST
PROGRAMMING TUTOR
PROGRAMMER'S UTILITIES
BRAIN GAMES
BRAIN TEASERS
BRAIN BUSTERS!
MANEUVERING GAMES
ACTION GAMES
REFLEX AND CONCENTRATION
TWO-PLAYER GAMES
KID'S GAMES
MORE GAMES
WORD GAMES
ELEMENTARY MATH
MIDDLE/HIGH SCHOOL MATH
VOCABULARY AND READING
MUSICAL EDUCATION
KALEIDOSCOPES AND DISPLAYS

For descriptions of these send a dollar for my catalog!

I goofed again! If you tried the Quickloader in Tips #29 with a disk containing more than 2# programs, you may have already noticed that line 14# should go to 16#, not 155.

Here's another Tigercub Challenge - can you run this and get these results?

```
>LIST
18# PRINT PI
11# PRINT MAX
12# PRINT PI
13# PRINT MAX
>RUN
 #
 #
 3.141592654

* SYNTAX ERROR IN 13#
```

Some of you sharp-eyed newsletter editors may have noticed that this text is being hyphenated to avoid some of those gaping blanks that occur when only a few long words will fit on a right-justified line. The only way that I have found to accomplish this is to set the TI-Writer right tab for the actual column width to be printed and then, whenever a word is hyphenated, backspace and replace the blanks on that line with carets, adding enough extra carets to justify the line - like this -

```
whenever^a^word^^is^^hyphen-
```

It helps to go into fixed mode with CTRL 9 when you are inserting extra carets.

When using this method, it is also necessary to set the paragraph indentation with IN # on the command line; if indentations are desired, they can be filled with caret signs, like this:

```
^^When using this method,
```

I am told that my old 3D Sprite Routine made it to the Golden Quickies section of CompuServe, so here is an updated version. I have found that sprites can be controlled much more easily (although not moved as rapidly) with CALL LOCATE, rather than turning them loose with CALL MOTION and then trying to catch up with them!

```
19# CALL CLEAR :: CALL SCREEN(5):: FOR SET=2 TO 8 :: CALL COLOR(SET,8,5):: NEXT SET :: DISPLAY AT(3,12):"3-D SPRITE DEMO"
11# DISPLAY AT(22,1):"BY TIGERCUB" :: CALL CHAR(4#,"FF818181818181FF818181818181FFFF818181818181FF818181818181FF")
12# CALL CHAR(36,RPT$("F",64)):: CALL MAGNIFY(4):: FOR X=2 TO 22 STEP 2 :: CALL SPRITE(#X,36,X/2+1-(X>7)-(X>13),32+X*6,4#+X*6):: NEXT X
13# S=1 :: CALL SPRITE(#S,4#,16,46,7):: FOR C=6 TO 42 STEP 2 :: CALL LOCATE(#S,46,C) :: NEXT C :: FC=44 :: FR=46 :: Y=#
14# FOR C=FC TO FC+44 STEP 2 :: CALL LOCATE(#S,FR,C):: NEXT C :: FC=FC+44 :: CALL SPRITE(#S+2,4#,16,FR,FC):: CALL DELSPRITE(#S):: TC=FC-32
15# FOR C=FC TO TC STEP -2 :: CALL LOCATE(#S+2,FR,C):: NEXT C :: TR=FR+34 :: FOR R=FR TO TR STEP 2 :: CALL LOCATE(#S+2,R,TC):: NEXT R
16# CALL SPRITE(#S,4#,16,TR,TC):: CALL DELSPRITE(#S+2):: FR=TR :: TR=FR-72 :: FOR R=FR TO TR STEP -2 :: CALL LOCATE(#S,R,TC):: NEXT R
17# CALL SPRITE(#S+2,4#,16,TR,TC):: CALL DELSPRITE(#S):: FR=TR :: TR=FR+5# :: FOR R=FR TO TR STEP 2 :: CALL LOCATE(#S+2,R,TC):: NEXT R
18# Y=Y+1 :: IF Y=11 THEN CALL DELSPRITE(#S+2):: GOTO 13# ELSE S=S+2 :: FC=TC :: FR=TR :: GOTO 14#
```

Ian Swales in Belgium can write some of the most intricate routines, and pull them into the tightest knot. I had searched everywhere for a sorting routine for 2-dimensional arrays, and invented some ridiculous ones, before Ian sent me this jewel.

```
1## !DEMO of two-dimensional sorting routine
11# !Set up array to be sorted
12# CALL CLEAR :: DIM A$(2#,4):: RANDOMIZE :: DEF X$=CHR$(26*RND+65)
13# FOR J=1 TO 2# :: A$(J,1)=X$&X$&X$ :: A$(J,2)=STR$(INT(1##*RND+1)):: A$(J,3)=X$&STR$(INT(1#*RND)):: A$(J,4)=INT(1#*RND))&X$ :: NEXT J
14# INPUT "SORT BY?(1-4)":K
15# J=2# !2-dimensional array sorting routine by Ian Swales
```

```
160 DIM Q(28):: FOR X=1 TO 2
8 :: Q(X)=X :: NEXT X
170 M=8
180 FOR X=1 TO J-1 :: IF A$(
Q(X),K)<=A$(Q(X+1),K)THEN 21
0
190 M=-1
200 T=Q(X):: Q(X)=Q(X+1):: Q
(X+1)=T
210 NEXT X
220 IF M THEN 170
230 FOR X=1 TO 28 :: FOR L=1
 TO 4 :: PRINT A$(Q(X),L);"
";:: NEXT L :: PRINT :: NEXT
 X :: GOTO 140
```

Did you ever need a routine that would accept either a string or a numeric value? Try this -

```
100 N=8 :: ON ERROR 110 :: A
CCEPT M$ :: N=VAL(M$):: GOTO
 120
110 ON ERROR STOP :: RETURN
120
120 ON (N=8)+2 GOTO 130,140
130 PRINT M$ :: GOTO 100
140 PRINT N :: GOTO 100
```

A useful tip from Stephen Shaw in England - if you have a long program which wil run only in Basic, and which will load from disk with CALL FILES(1) but runs out of memory when you try to run it; and if you have the MiniMemory module -
Insert MiniMemory module, select Basic, enter CALL FILES(1), Enter NEW, enter OLD DSK1.(filename). When loaded, enter SAVE EXPMEM2. When SAVEd, enter CALL LOAD(-31888,63,255), enter NEW, enter OLD EXPMEM2, and enter RUN. That is still a lot faster than loading a long program from tape!

Another reason for never using the default mode of so-called UPDATE when opening a file (without specifying INPUT or OUTPUT) is that you will get an I/O ERROR #1 if the file is write-protected.

Has anyone found a way to go from Extended Basic to Basic without losing the program in memory, or at least fouling it up?
CALL LOAD(-32116,4) has been published in many newsletters as a way to do this, but has anyone actually made it work?

If you are printing out of TI-Writer Editor, finish your letter with CTRL U, SHIFT L, CTRL U and when it is printed the paper will automatically feed to the top of the next sheet.

To make a note to yourself while programming, just type 1! and whatever you want to make note of, then LIST "PIO":1, and then type 1 and enter to delete the line.

TI-Writer puts an extra space after every period that is followed by a space. If you don't want this extra space after abbreviations such as "Mr." or St.", use a caret sign ^ instead of a space after the period, Mr.^Jones. But TI-Writer puts only one space after ? or ! so if you want two, put a caret after the symbol !^

One of the very best tips for this month comes from Paul A. Meadows, in the September 85 newsletter of T.I.N.S. (Nova Scotia, Canada) -
How to print up to 132 characters in a line (condensed print, of course) out of TI-Writer! Just prepare your file as usual but in line 8881 put formatter commands such as .LM 10;RM 132; IN +5;FI;AD . The Fill and Adjust are necessary, the Indent is up to you, as are the left and right margins - but notice that right margin set way over at 132?
Now, instead of saving the file with SF, type PF and then C DSK1.(filename) to print to the disk. This not only strips out the control C characters, it also erases the TI-Writer tab line that was applied to the last line of the file.
So now, with your printer opened and initialized for condensed print, go into the TI-Writer formatter mode and print your file!

I have made the following changes to my working copy of the Tigercub Menuloader. This sets up my Gemini printer to skip over the perforations and print full page width in elite print with a wide left margin for ring-binder punching. Other printers may need changes in these codes.

```
620 DISPLAY AT(12,1)ERASE AL
L:"PRINTER? PIO" :: ACCEPT A
T(12,18)SIZE(-18):P$ :: GOSU
B 895 :: PP=3
840 DISPLAY AT(24,1):"PRINTE
R NAME? PIO" :: ACCEPT AT(24
,15)SIZE(-14):PP$ :: GOSUB 8
95 :: PRINT #2:SEG$(D$,1,4)&
" - Disknane= "&N$
895 OPEN #3:P$,VARIABLE 132
:: PRINT #3:CHR$(27);"8";CHR
$(2);CHR$(27);"N";CHR$(10);C
HR$(27);"N";CHR$(6):: RETURN
```

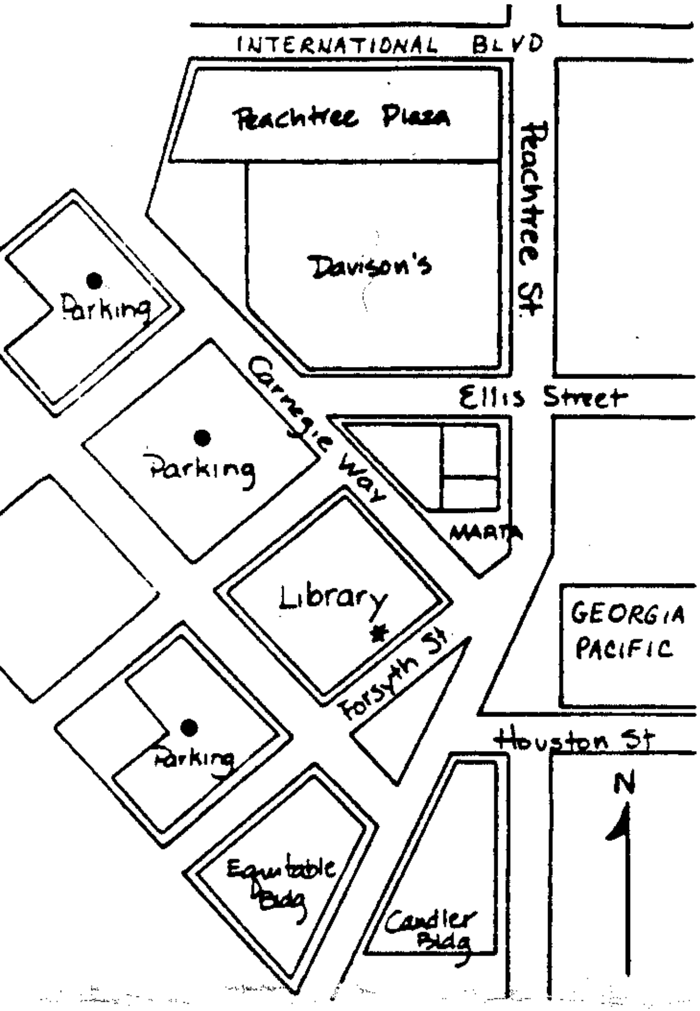
I always keep a backup of everything, on the flipped side of another disk, and I often want to verify that the backup has everything that is on the master, and vice versa.

```
100 DISPLAY AT(3,6)ERASE ALL
:"TIGERCUB DOUBLECAT": :" To
 compare the contents of": :
"a disk with a backup." !by
Jim Peterson
110 DISPLAY AT(12,1):"INSERT
 MASTER DISK": :"PRESS ENTER
"
120 CALL KEY(8,K,S):: IF S=8
 THEN 120
130 DATA DF,DV,IF,IV,P
140 RESTORE :: FOR I=1 TO 5
:: READ T$(I):: NEXT I
150 DIM F$(127):: OPEN #1:"D
SK1.",INPUT ,RELATIVE,INTERN
AL :: INPUT #1:A$,J,J,K :: F
$(8)=A$&" "&STR$(K)
160 X=X+1 :: INPUT #1:F$(X),
I,J,K :: IF F$(X)="" THEN 17
0 :: F$(X)=F$(X)&" "&T$(ABS(
I)):: GOTO 160
170 X=X-1 :: CLOSE #1 :: DIS
PLAY AT(12,1)ERASE ALL:"REMO
VE MASTER DISK": :"INSERT BA
CKUP DISK": :"PRESS ENTER"
180 CALL KEY(8,K,S):: IF S=8
 THEN 180
190 OPEN #1:"DSK1.",INPUT ,R
ELATIVE,INTERNAL :: INPUT #1
:A$,J,J,K :: DISPLAY AT(1,1)
ERASE ALL:F$(8);:: DISPLAY A
T(1,15):A$&" "&STR$(K);
200 Y=Y+1 :: R=R+1 :: GOSUB
290 :: INPUT #1:A$,I,J,K ::
IF A$="" THEN 260 :: K$=A$&"
 "&T$(ABS(I))
210 IF K$=F$(Y)THEN DISPLAY
AT(R+1,1):F$(Y);:: DISPLAY A
T(R+1,15):K$;:: GOTO 250
220 IF K$<F$(Y)THEN DISPLAY
AT(R+1,15):K$;:: Y=Y-1 :: GO
TO 250
230 DISPLAY AT(R+1,1):F$(Y);
:: R=R+1 :: GOSUB 290 :: Y=Y
+1
240 IF K$=F$(Y)THEN 210 ELSE
 IF K$<F$(Y)THEN 220 ELSE IF
 Y<X THEN 230 ELSE DISPLAY A
T(R,15):K$;
250 GOTO 200
260 IF Y>X THEN 280
270 R=R+1 :: GOSUB 290 :: FO
R J=Y TO X :: DISPLAY AT(R,1
):F$(J):: R=R+1 :: GOSUB 290
 :: NEXT J
280 DISPLAY AT(24,1):"     P
RESS ANY KEY" :: CALL KEY(8,
K,S):: IF S=8 THEN 280 ELSE
CLOSE #1 :: END
290 IF R<23 THEN RETURN
300 DISPLAY AT(24,1):"PRESS
ANY KEY" :: DISPLAY AT(24,1)
:" " :: CALL KEY(8,K,S):: IF
 S=8 THEN 300
310 CALL CLEAR :: R=1 :: RET
URN
```

And that is just about

MEMORY FULL!

Jim Peterson



Atlanta
99/4A
Computer
Users
Group

NEXT MEETING:

SUNDAY, FEB 16

ATLANTA PUBLIC LIBRARY

3:00 P.M.

FOR MORE DETAILS, CALL 231-0992

(A DONATION WILL BE REQUESTED - MEMBERS - 50¢, NON-MEMBERS - $1.00)

ATLANTA 99/4A COMPUTER USERS GROUP
POST OFFICE BOX 19841
ATLANTA, GEORGIA 30325

BULK RATE
U.S.POSTAGE
PAID
ATLANTA, GA
PERMIT No. 492

UG EX IN
Miami County Area 99/4A H.C. U.G.
P. O. Box 1194
Peru, IN 46970

* DUES ARE DUE THIS MONTH
** DUES WERE DUE LAST MONTH
*** THIS IS YOUR LAST NEWSLETTER