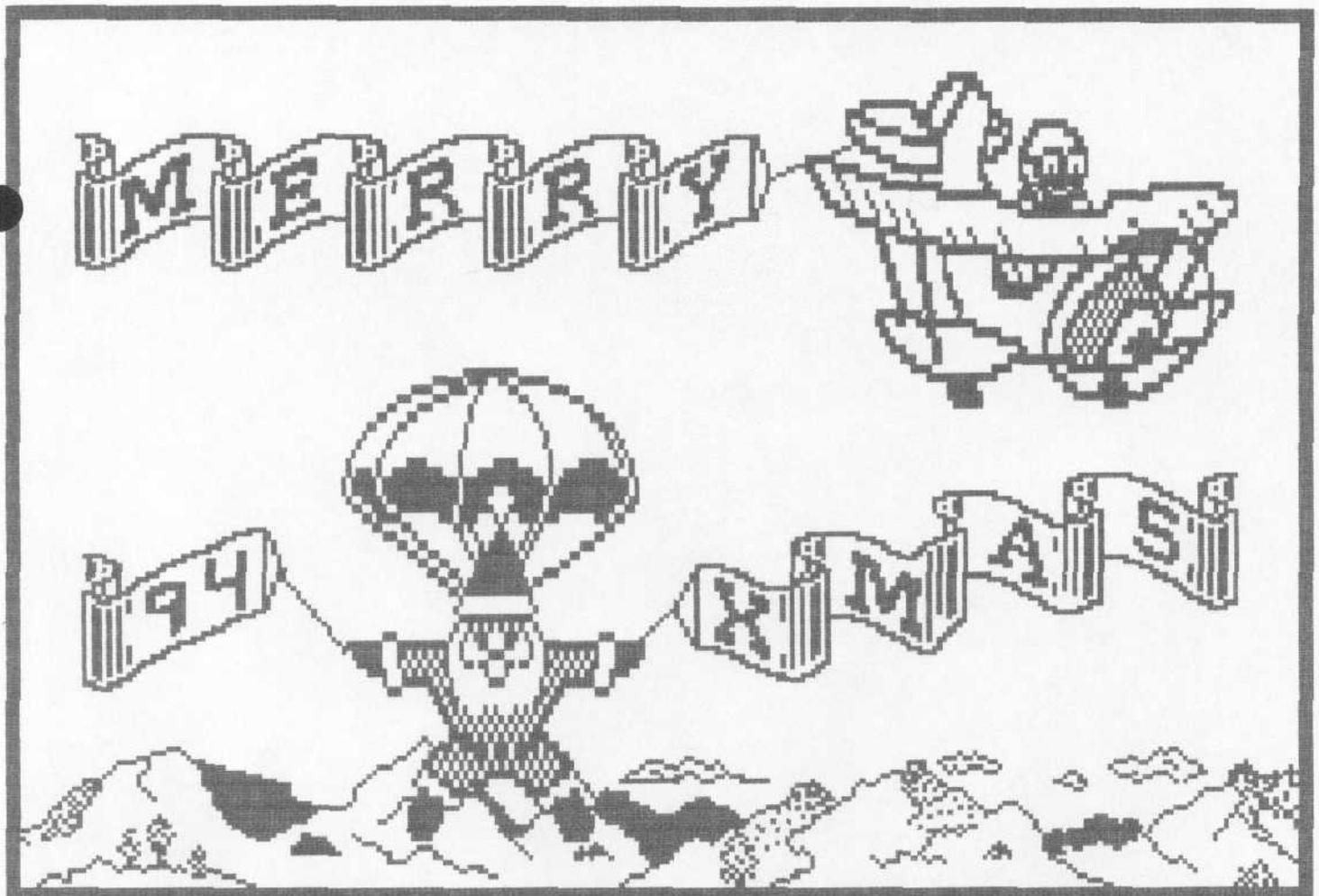




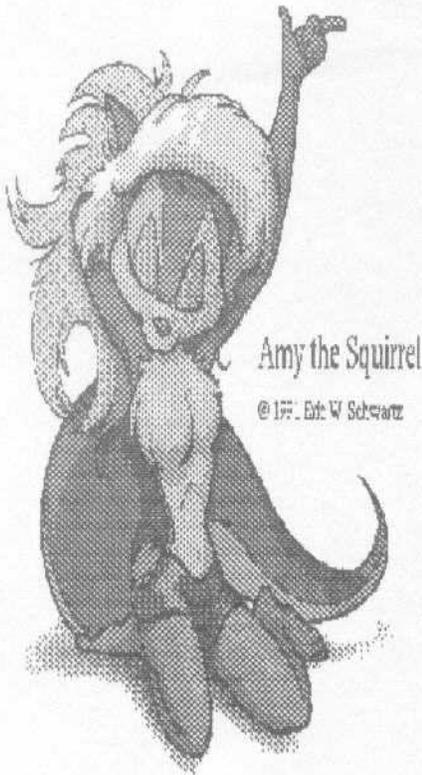
The Ottawa TI-99/4A
User's Group

NEWSLETTER

Volume 13, Number 10 ... December 1994



OTIUG: 222 Guigues Ave, Apt 603
Ottawa, Ontario CANADA K1N 5J2



Amy the Squirrel

© 1997. Ed: W Schwartz

COMING EVENTS

December meeting: 6 Dec 94

January meeting: 3 Jan 95 (to be confirmed)

Meetings take place at Lucie Dorais' apartment, at 7:30pm. Everyone is welcome for a friendly, informal meeting to discuss the TI.

Please call in advance, Lucie's number is on the list on the last page.

Any and all articles are welcome and solicited for publication in this newsletter. Submit your articles to Lucie Dorais c/o editor, or directly to me by internet email at mbrent@proton.com

DISCLAIMER

The OTIUG is a non-profit organization based in Ottawa, Ontario, Canada, providing support to users of Texas Instruments and compatible home computers. Opinions expressed in this newsletter are those of the individual authors, and not necessarily those of the OTIUG itself. Delivery to Gilligan's Island not available at this time.

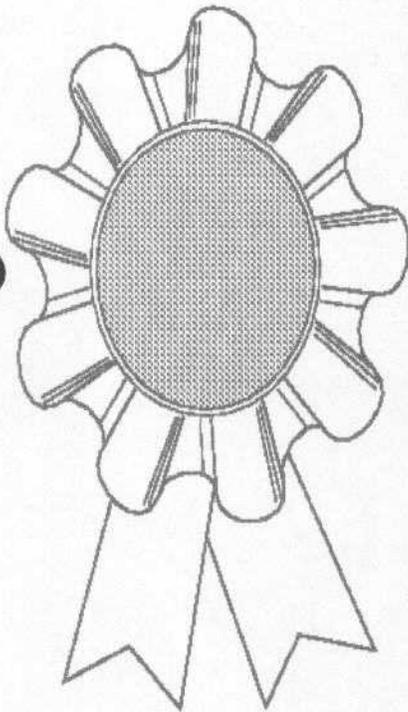


The President's Two Cents Worth

Christmas season... this year, the decorations, music etc even started before Halloween in some stores... good thing I cannot go out too much.

Not that I am not in the Xmas spirit: I started last month, almost as a game, to translate my TI program GIFTS to Visual Basic; I am still working on it... I added a few features, not much: just one for your "friend" to place the gifts on screen, instead of the computer, but mostly graphics, very detailed; like for the cover here, I used a lot of little TIPS pictures (transported to PC and converted); but now I also have to colour them... And of course I want to "publish" GIFTS before December, so it has time to make its way around some BBSs before it is outdated.

Talking about graphics, the choice of the subject for this month cover imposed itself on me after I decided I should mention the work of someone very dear to me, but who does not like to be singled out. Suffice it to say that he, once again, spent hours a few weeks ago to put Texlink back on track, and that for only the four or five regular users (and the occasional ones: just at the time Texlink had problems, Jack McAllister tried to log on and called me from Kentucky to tell me Texlink was sick... He finally managed to log successfully and leave a message. Thanks Jack!).



The name of our own Santa Claus is somewhere near the back of the newsletter...

In the November issue, Mike mentioned that I had asked him to take out the membership form... True... Since no one has volunteered to take over after me, and since Mike himself is leaving for BC in April, I figure this will be the end of the newsletter, which is where most of the UG subscription money goes (a small part goes to mama Bell for Texlink). And since no one, except our last faithfuls, Jeff Brown and Mike Ward-Brent, has protested the Club's demise, I don't feel overwhelmingly guilty.

Time to quit, deadline already passed, and Mike left me a message half an hour ago... this month, he is deadly serious about the schedule... actually, because of the holidays, the January newsletter will have to be done... two weeks from now! I will try to find something exciting to write about, and something else than frogs frolicking on a TI for the cover. In the meantime, have all a very Merry Christmas!

Tales from the Editor Zone

Hello again, and welcome to another episode of the OTIUG Newsletter.



Heh, from reading the pres notes, you'd think I was an evil slave driver, hovering over Lucie and Jeff with a bullwhip in one hand, and an iron mace in the other. I don't think I'm "that" bad... I'm really a kind, generous, loving individual... NO! Jeff, you may NOT take a break! Get back to your seat! You have an article to write!

errr... <Grin>

But, yes, I'm trying to get the deadlines down pat this month, cause next month (December is still next month right now!) I'll be in British Columbia from the 12th-28th, and I won't be able to do the newsletter during that time. So, either the newsletter will be very early, or very late.

Well, things look pretty good this month, with Jeff and Lucie supplying good articles. I threw in a short little text from my collection, on the care of diskettes.. it's good for a quick laugh. I'm running VERY short on pictures that are even mildly relevant... time to go hunting, I guess. <grin>

Anyway, Merry Christmas to all, and the best of the season to you. May Santa bring you everything you're hoping for!

PS: if your address has changed, please be sure to let me know! I can't effect the changes if I don't know about them!!

PS2: I'll keep inventing the name of Jeff's article till he tells me what it should be! <grin>

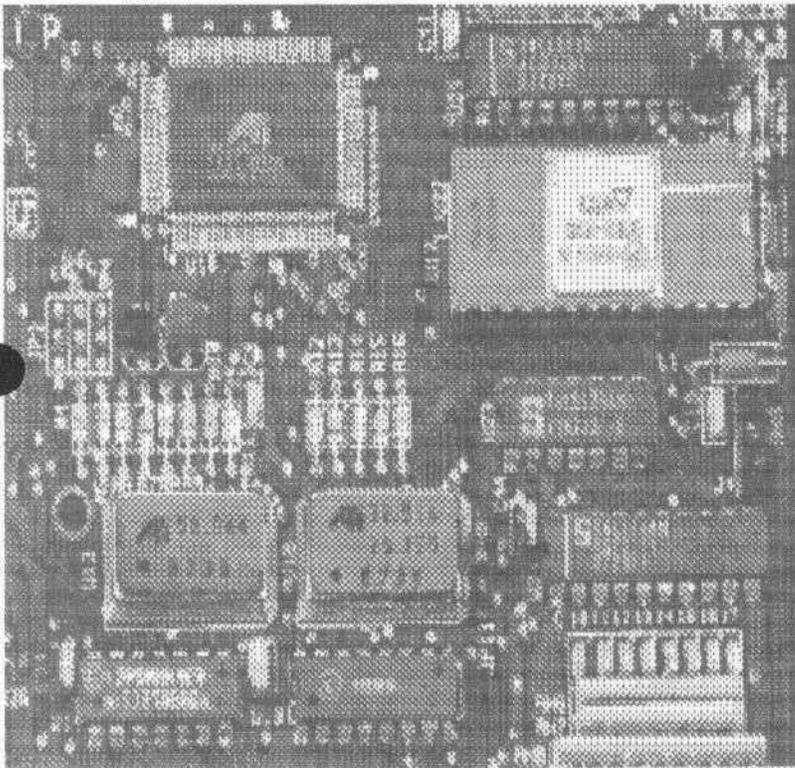
Merry Christmas

Jeff's World II - the Party Continues!

Heh!!! I haven't forgotten you yet! (God forbid I ever do...) Well, I've been having fun lately teaching myself a new language (yes, another one...) Turbo Pascal. I recently got access to my High School's computers (even if I'm not taking those courses, that teacher's a nice guy!). They have QBasic (bleah!) and Turbo Pascal, so far as programming languages go. Pascal is actually quite easy when you get used to it, similar with respect to C and Basic.

Anyhow, returning to the TI world... Reading back in the articles showed me something I didn't like, I broke a promise to you guys. You recall I promised to do up a word processor program in XB using the

windowing routines. I had begun some work on it, and proved how useless the program would be. I dropped it, thinking I would come back to it later. Obviously it slipped my mind, since I got caught up in more interesting things (my term program, now officially called "Term 80". I'm having trouble keeping up with my mail! Quite some demand out there! I could make \$\$\$ from it, but I'm not that kind of guy, the term is useful to me, and I might as well share it!).



Anyhow, I'm sorry to say, I never returned to it. The preliminary program was deleted in a system crash way back in June/July. Who needs another word processor anyhow! Again, my apologies to you. If you did come up with some kind of program, by all means!! I want to see it! I found the windowing utils particularly useful in a paint program I did up in XB... including scrolls and stuff... quite nice,

called NeoChrome... a tad slow though, but people around here seem to like it a lot! (I had trouble keeping 'em off it so I could work! If you're interested in it, call me. I was planning an all assembly version too, and now I know the techniques well enough to proceed... remind me!)

Ok, last month I said I'd show you a program which prints out TI Artist fonts in XB (then convert it to assembly.) I have a habit of doing thing in XB first, as a model (albeit a slow, sluggish one), to base my final assembly program on. For those of you who may figure I'm going too fast, I am! I find it difficult to slow down... don't worry though, I'll do better with the explanations!!!

So, the purpose of the program is to load and be able to print out any text file using TI-Artist Plus fonts. Ok. Most fonts are rather large, so we'll need some buffering space. No assembly needed so we can use the lower 8k bank for storage of the Font's patterns, otherwise expect a number of Memory Full messages!!!!

Reading font file:

According to the TI Artist manual, a font file has the following format:

Record #1 : Ascii character begin defined (ex. C)

#2 : Width, Hight, Skip width. The Width and Hight are in characters, so 2,4,12 means 2 chars wide, 4 chars high and skip 12 pixels. The Skip width is the distance in pixels from the beginning of the character, to the beginning of the next.

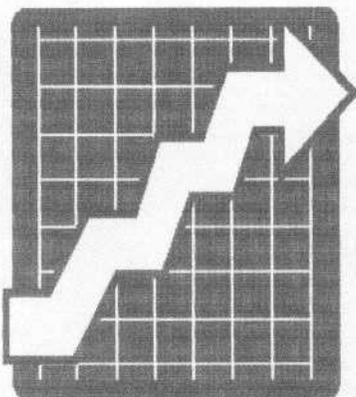
#3+: Character definitions, given left to right, top to bottom, in decimal. (ex. 1,4,65,34,21,2,3,4). This goes on until Hight*Width chars are defined. (the whole font character.)

#? : Sequence repeats for additional chars, until EOF.

Reading this in XB is relatively simple. Storing in low-memory is done with CALL LOAD (needs a CALL INIT call beforehand). Reading back these values is accomplished with CALL PEEK. Simple enough?

Printing a text file using font:

Now, THIS is NO easy job!!! It also explains why the program will run slowly (though, in assembly it does it at amazing speed!). Don't expect a light-speed dump with this... XB has to read a font character and update the print strings which are then printed when a Carriage Return or end of line (either on the printer, or in the file.)



I will see what I can do about explaining the print routine... It fetches the next character from a disk file, then looks up a table to see where in low-memory it finds that character (if it's non-existent, it skips it.) Through a series of rather complex instructions, it converts it to a standard 8 pin e-son Normal Density Graphics string.

Program in general:

Ok, included in the program is a simple menu system and cataloging routine (to save time). A reminder that you have to reload the font file if you leave the program. (Assembly version keeps font in RAM as long as XB is booted, or until memory is re-initialized with a CALL INIT or reboot.)

Well... here it is!!! (Ed note: don't type the lines that start with "->", they are comments for you!)

1 | Font Print program, XB version.

2 | By Jeffrey Brown, OTIUG.

99 ON WARNING NEXT

-> kill warnings

100 DIM CHW\$(255),CHADD(255),CHL\$(7,2)

-> CHW\$ holds the char hight/width/skip.

-> CHADD has addresses in low memory

-> CHL\$ holds current print string.

110 STADD=8200 :: ENDADD=16383 :: CURADD=STADD :: CALL CLEAR :: CALL INIT

-> STADD=lowmem start (skip ~~utilTab~~ UtilTab)

```

-> ENDADD=end of lowmem
-> CURADD=current address
-> CALL INIT to allow CALL LOAD operations.
120 CALL SCREEN(4):: DISPLAY AT(1,9):"Font Print"
130 DISPLAY AT(4,9):"Information": "Current font: "&FN$:"Free Space ";ENDADD-CURADD+1
    -> current buffer info.
140 | MAIN MENU easy to understand...
141 CALL HCHAR(10,1,32,480)
150 DISPLAY AT(10,9):"Main Menu": "1) Load font file": "2) Print document": "3) Quit"
160 CALL KEY(0,K,S):: IF S<1 OR K<49 OR K>53 THEN 160 ELSE IF K=49 THEN GOSUB 1000 ELSE
IF K=50 THEN GOSUB 2000 ELSE 170
161 GOTO 141
170 DISPLAY AT(24,1):"Quit: Sure? Y/N N" :: ACCEPT AT(24,18)VALIDATE("YN")SIZE(-1)BEEP:S$:S$ ::
IF S$="" THEN 170 ELSE IF S$="N" THEN 141
180 END
999 | LOAD FONT FILE MENU again simplistic.
1000 CALL HCHAR(10,1,32,480):: DISPLAY AT(10,9):"Load Font": "1) Load file": "2) Catalog disk": "3)
Quit to main"
1001 DISPLAY AT(6,1):"Current font: "&FN$:"Free Space ";ENDADD-CURADD+1
1010 CALL KEY(0,K,S):: IF S<1 OR K<49 OR K>53 THEN 1010 ELSE IF K=49 THEN 1100 ELSE IF
K=50 THEN 1300 ELSE RETURN
1099 | LOAD FILE
1100 LF$=FN$ | LF$ is the last filename, FN$ is current.
1102 DISPLAY AT(16,1):"File name: "&LF$ :: ACCEPT AT(16,12)SIZE(-16)BEEP:LF$ ::DISPLAY
AT(18,9):"Working..."
    -> open file and all...
1110 ON ERROR 1111 :: OPEN #1:LF$,INPUT ,DISPLAY ,VARIABLE 80 :: GOTO 1120
    -> on error to permit error trapping!
1111 DISPLAY AT(16,1):"File Error!!!"
1112 CALL KEY(0,K,S):: IF S<1 THEN 1112 ELSE RETURN 1000
1120 R=20 :: C=1 :: CURADD=STADD :: MAXH,MAXW=1
    -> ok, current Row/Col (for character display), Maximum hight/width to allow
    -> initialization in print routine.... allowance for largest chars.
1125 | check EOF and get next char.
1130 IF EOF(1)=0 THEN LINPUT #1:A$ ELSE CLOSE #1 :: ON ERROR STOP :: FN$=LF$ ::GOTO
1000
1132 IF A$<>"" THEN CHADD(ASC(A$))=CURADD ELSE A$=" " :: GOTO 1132
1140 DISPLAY AT(R,C):A$ :: C=C+1 :: IF C>28 THEN R=R+1 :: C=1
    -> show it!
1150 INPUT #1:W,H,S :: CHW$(ASC(A$))=CHR$(W)&CHR$(H)&CHR$(S):: MAXH=MAX(MAXH,H)::
MAXW=MAX(MAXW,W)
    -> init CHW$ to width/hight/skip of char. Adjust MAXH/MAXW. Notice the use
    -> of MAX here... there will be more.
1160 FOR A=1 TO W :: FOR B=1 TO H :: INPUT #1:C1,C2,C3,C4,C5,C6,C7,C8 :: CALL
LOAD(CURADD,C1,C2,C3,C4,C5,C6,C7,C8):: CURADD=CURADD+8 :: NEXT B :: NEXT A
    -> this line fetches the character data records and stores them using CALL
    -> LOAD as explained.
1170 GOTO 1130

```

1299 | CATALOG DISK ok, a disk cataloger.

1300 GOSUB 1302 :: GOTO 1000

1301 GOSUB 1302 :: GOTO 2000

-> ok, cheap way to do this... subroutine!

1302 DISPLAY AT(16,1):"Disk name: "&SEG\$(FN\$,1,5):: ACCEPT AT(16,12)SIZE(-16)BEEP:DR\$

1310 ON ERROR 1311 :: OPEN #1:DR\$,INPUT ,INTERNAL,RELATIVE :: INPUT #1:A\$,Z,Z,Z: DISPLAY

AT(18,1):"Name: "&A\$:: GOTO 1320

1311 DISPLAY AT(16,1):"File Error!!!"

1312 CALL KEY(0,K,S):: IF S<1 THEN 1312 ELSE RETURN 1410

1320 R=19 :: C=1

1330 INPUT #1:A\$,Z,Z,Z :: IF A\$<>"" THEN DISPLAY AT(R,C):A\$ ELSE 1400

1340 R=R+1 :: IF R=25 THEN R=19 :: IF C=14 THEN 1390 ELSE C=14 :: GOTO 1330

1350 GOTO 1330

1390 CALL KEY(0,K,S):: IF S<1 THEN 1390 ELSE CALL HCHAR(19,1,32,192):: GOTO 1320

1400 CALL KEY(0,K,S):: IF S<1 THEN 1400 ELSE CLOSE #1 :: ON ERROR STOP

1410 RETURN

1999 | PRINT FILE MENU EASY ENOUGH!

2000 CALL HCHAR(10,1,32,480):: DISPLAY AT(10,9):"Print File": "1) Print file": "2) Catalog disk": "3)

Quit to main"

2010 CALL KEY(0,K,S):: IF S<1 OR K<49 OR K>53 THEN 2010 ELSE IF K=49 THEN 2100 ELSE IF K=50 THEN 1301 ELSE RETURN

2099 | PRINT FILE

2100 LF\$=FN\$:: PC\$=CHR\$(27)&"K" :: LW=500 :: RCW,CW=1 | NORMAL DENSITY GRAPHICS

-> ok, PC\$ is the print code, adjust to suit your printer if need be.

-> LW is max line width.

-> CW is current width (position in print string). RCW is ?? forget...

2102 DISPLAY AT(16,1):"File name: "&LF\$:: ACCEPT AT(16,12)SIZE(-16)BEEP:LF\$:: DISPLAY AT(18,9):"Working..."

-> open file...

2110 ON ERROR 2111 :: OPEN #1:LF\$,INPUT ,DISPLAY ,VARIABLE 80 :: GOTO 2120

2111 CALL ERR(Z1,Z2,Z2,Z2) | get error code.

2112 DISPLAY AT(17,1):"File error!!! Code #";Z1

2119 CALL KEY(0,K,S):: IF S<1 THEN 2119 ELSE RETURN 2000

2120 OPEN #2:"PIO.CRLF",OUTPUT,DISPLAY ,VARIABLE 254 :: PRINT

#2:CHR\$(27)&"2"&CHR\$(8)&CHR\$(27)&"A"

-> open printer... needs CRLF to work. Sends a printer init string too...

2125 FOR A=0 TO MAXH-1 :: CHL\$(A,0),CHL\$(A,1),CHL\$(A,2)=RPT\$(CHR\$(0),255):: NEXT A | clear print line buffer array...

2126 | get next char... send CRLF sequence if end of line.

2130 IF EOF(1)=0 THEN LINUT #1:A\$:: P=1 ELSE GOSUB 2301 :: CLOSE #1 :: PRINT

#2:CHR\$(13)&CHR\$(11):: CLOSE #2 :: ON ERROR STOP :: GOTO 2000

2140 DISPLAY AT(20,1):A\$ | show line.

2150 C\$=SEG\$(A\$,P,1):: P=P+1 :: IF C\$="" THEN 2130 ELSE C=ASC(C\$)

-> get next char...

2160 IF C=13 THEN 2300 ELSE IF CHADD(C)<>0 THEN W=ASC(CHW\$(C))::

H=ASC(SEG\$(CHW\$(C),2,1)):: S=ASC(SEG\$(CHW\$(C),3,1))ELSE 2150

-> get address of character.

2165 IF CW+S>LW THEN 2300

```

-> if the skipwidth plus the current width exceeds the maximum line width,
-> print line.
2166 | main print routine. It takes each character definition and rotates it,
-> then places it into the print string. Try to decipher this!!! A faster
-> way would be to rotate upon loading... but I'm lazy... do it if you like.
2170 LCW=CW :: ADD=CHADD(C):: FOR A=0 TO H-1 :: FOR B=0 TO W-1 :: CALL PEEK(ADD,
C1,C2,C3,C4,C5,C6,C7,C8):: FOR C=7 TO 0 STEP -1
2180 V=2^C :: T$=CHR$((C1 AND V)/V*128+(C2 AND V)/V*64+(C3 AND V)/V*32+(C4 AND
V)/V*16+(C5 AND V)/V*8+(C6 AND V)/V*4+(C7 AND V)/V*2+(C8 AND V)/V)
2182 IF (CW-1)/255<>INT((CW-1)/255) THEN TT$=SEG$(CHL$(A,INT(CW/255)),
1,CW-INT(CW/255)*255-1)ELSE TT$=""
2183 TT$=TT$&T$ :: IF (CW+1)/255<>INT((CW+1)/255)THEN TT$=TT$&SEG$(CHL$(A,INT(CW
/255)),CW-INT(CW/255)*255+1,255)
2184 CHL$(A,INT(CW/255))=TT$
2190 CW=CW+1 :: NEXT C :: ADD=ADD+8 :: NEXT B :: CW=LCW :: NEXT A :: CW=LCW+S
2210 GOTO 2150
2299 | PRINT OUT THIS LINE
2300 GOSUB 2301 :: GOTO 2150
2301 FOR A=0 TO MAXH-1 :: PRINT #2:PC$&CHR$(CW-INT(CW/256)*256)&CHR$(INT(CW/256))
2305 PRINT #2:SEG$(CHL$(A,0),1,MAX(MIN(CW,255),0));SEG$(CHL$(A,1),1,MAX(MIN(CW-255,255),
0));SEG$(CHL$(A,2),1,MAX(MIN(CW-500,255),0))&CHR$(13)&CHR$(11):: NEXT A
2320 CW=RCW :: FOR A=0 TO MAXH-1 :: CHL$(A,0),CHL$(A,1),CHL$(A,2)=RPT$(CHR$(0),255)::
NEXT A
2330 RETURN

```

Explanation of 2305... lots of MAXs and MINs!!! They are limit checkers. They could be encoded as IF-THEN statements, but this is faster. It works like this:

$\text{MAX}(\text{Low-limit}, \text{MIN}(x, \text{High-limit}))$

If x is greater than the high limit then $\text{MIN}(x, \text{high-limit})$ returns the high limit, then $\text{MAX}(\text{low-limit}, \text{High-limit})$ (because that's what the MIN evaluated to) returns the high limit. So if $X > H\text{-lim}$, return $H\text{-lim}$. If X is less than the low limit, $\text{MAX}(\text{low-limit}, \text{MIN}(x, \dots))$ returns low-limit. Too hard?

This shrinks down to:

```

if x>Hlim then return(Hlim);
else if x<Llim then return(Llim);
    else return(x);

```

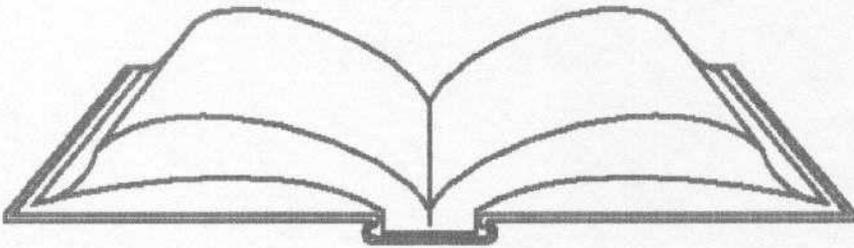
Well, you might see that kind of code pop up in Lucie's column someday!! Well, type in the XBasic program, try it out, and I'll begin the conversion to Assembly next month. It may take a few months to properly show you how.

P.S: Does anyone know where I can pick up a few 8k*8 static SRAMs for my 192k Horizon RAMdisk cheaply? Active is still pretty expensive.

Till next month! Happy Tling!!! (TI telecommuting!)

NINE YEARS AGO...

by Lucie Dorais



Well, a change of title, but only this month, as I could not find the December 1984 issue of the Newsletter. And since I doubt that I will still be around a year from now, here is what a Christmas number looked like in 1985.

To start with, the editor had gone to great extent to put a graphic on the cover, only instance before April 1986. It is a schematized Christmas tree; since the text looks very much like TI characters, I suspect Graphx was used.

The Pres Two Cents worth start again (refer to last month; the Pres is not the same, now Berry Minuk) with a note that the Dec. meeting (what, they did the newsletter AFTER the meetings then?) "was a bit disorganized. We will try to have more order at the Jan. meeting". Great demos at that meeting: a new printer from Centronics and, better remembered, a new version of the MAXIMEM personally presented by its creator, Guy Gournay, who came "out of the blue" from in Montreal; the new version allowed you to dump modules to cassettes as well as to disk; I wonder how many people did that.

Bob Boone had had a suggestion box at a previous meeting; one suggestion, which was still debated not too long ago: would it save money to bring the newsletter to the meeting for people who attend (yes, but the question was the same then as now: the Newsletter served as a reminder of the meeting). Again, the "disorganization" of the last meeting is mentioned, so I guess it was something to witness (I was there, but don't remember).

"Mr., Ms or Mrs M Taylor wants guidance on uploading to and downloading from BBSs."; Ah, Mike Taylor, a quiet man of few words... probably very new to the Group then. Peter Arpin was already SysOp by that time, and praised for his excellent work; apparently, there were not enough files to download on the BBS; rather, we were then so hungry that we always wanted more (we excluded me: I had got my drives by then, but was not modemed yet).

At that time, the UG was collecting money to acquire a whopping 10 megs hard drive: another page mentions how the "hard disk drive" drive is coming, there was \$955 already. but that was not enough... We needed \$1200. We finally got the HD, and it is still working for Texlink.

Bob's final note: "The most amazing thing is that WE'RE STILL GROWING!"; but some members were already leaving for "THEORETICALLY bigger and better systems" (not mentioned). DM1000 had seen the light and was a huge success. The Ottawa UG was even officially represented at the Chicago Faire in

November, as well as the Sherbrooke, Qu bec, group SHER-TI (still in existence; the name is a French pun on "Dear TI...") and the Nova Scotia group, TINS (which has only recently ceased to exist I think).

Dennis Denny, a very nice man as I remember him, wrote about his printing problems with the CorComp RS232 card: there was an easy hardware fix (that was for the PEB card only, as I never had any problems with the CorComp Mini-Expansion box card). Another blurb, by Bob Lanoy, about the new Centronics printer, with special offer to Tiers if bought in quantity; any sold at all?

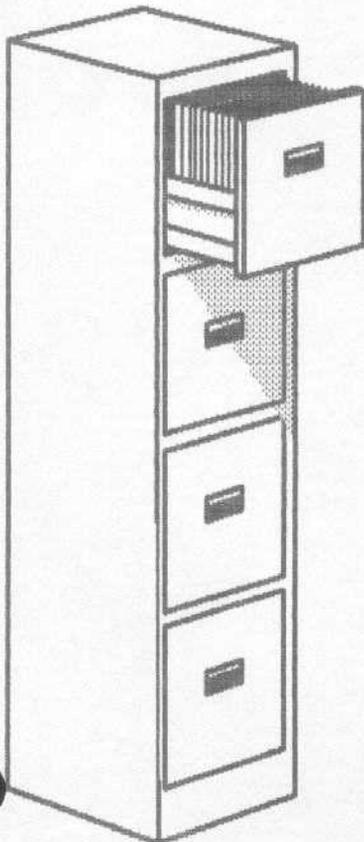
Big ad from CompuCentre in Carlingwood Mall, designed with Graphx, all text though. The game modules were then down to \$10.88 to \$19.88 (somewhere else, a second hand console could be had for \$50 only, second hand XB module for \$60; a complete disk system, one SSSD drive, was \$1200).

"NUAC NEWS by Bob Boone"; now, how many remember NUAC? That was the Association of Canadian Users' Groups; it never really took flight, but Bob and Jane Laflamme worked a great deal on the project.

The big question: "DISK HEADS - To clean or not to clean?" The answer: not more than once or twice every four or five years...

Finally, a long list of CALL LOADs by Terry Atkinson (of Halifax) with, for once, a lengthy explanation of what they do. And last but not least, Art Byers reviews the first issue of The GENIAL TRAVELER, on of the first "Diskazines", published by Barry A. Traver in Philadelphia; Barry is still very much around on GENIE, and his handle is... GENIAL.AL. I guess that is how he wants the world to know him <big Grin! when I write to him, I address him as "Genial Barry...">.

(IM)PROPER CARE OF FLOPPIES



1. Never leave diskettes in the disk drive, as data can leak out of the disk and corrode the inner mechanics of the drive. Diskettes should be rolled up and stored in pencil holders.
2. Diskettes should be cleaned and waxed once a week. Microscopic metal particles can be removed by waving a powerful magnet over the surface of the disk. Any stubborn metallic shavings can be removed with scouring powder and soap. When waxing the diskettes, make sure the surface is even. This will allow the diskette to spin faster, resulting in better access time.
3. Do not fold diskettes unless they do not fit into the drive. "Big" diskettes may be folded and used in "little" disk drives.
4. Never insert a diskette into the drive upside down. The data can fall off the surface of the disk and jam the intricate mechanics of the drive.
5. Diskettes cannot be backed up by running them through the xerox machine. If your data is going to need to be backed up, simply insert two

diskettes into the drive. Whenever you update a file both disks will be updated.

6. Do not insert or remove your diskettes from the disk drive while the red light is flashing. Doing so could result in smeared or possibly unreadable text. Occasionally the red light remains flashing in what is known as a "hung" or "hooked" state. If your system is "hooking" you will probably need to insert a few coins before being allowed access to the slot.

7. If your diskette is full and you need more storage space, remove the disk from the drive and shake vigorously for 2 minutes. This will pack the data enough (Data Compression) to allow for more storage. Be sure to cover all the openings with scotch tape to prevent loss data.

8. Data access time can be greatly improved by cutting more holes in the diskette jacket. This will provide more simultaneous access points to the disk.

9. Diskettes may be used as coasters for beverage glasses, provided that they are properly waxed beforehand. Be sure to wipe the diskettes dry before using.

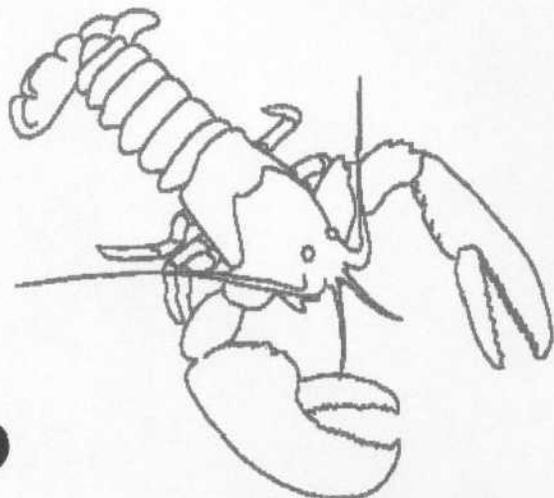
10. Never use scissors and glue to manually edit documents. The data is stored much too small for the naked eye, and you may end up with data from some other document stuck in the middle of your document. Razor blades and scotch tape may be used, however, provided the user is equipped with an electron microscope.

11. Periodically spray diskettes with insecticide to prevent system bugs from spreading.

VIRUS FUN

More computer viruses have been detected in or around the Seattle area. Please be alert for them. The flavour is truly American this time (no viruses breed in Canada?), and some have been eradicated since (Mario Cuomo, Ollie North...)

(Note: this is the second and last installment of the collection; the first part was published in the September newsletter.)



AT&T Virus: Every three minutes it tells you what great service you are getting.

MCI Virus: Every three minutes it reminds you that you're paying too much for the AT&T Virus.

Ross Perot Virus: Activates every component in your system, just before the whole damn thing quits.

Mario Cuomo Virus: It would be a great virus, but it refuses to run.

Ted Turner Virus: Colorizes your monochrome monitor.

Dan Quayle Virus: There is something wrong with your computer, and we just can't figure out what.

New World Order Virus: Probably harmless, but it makes a lot of people really mad just thinking about it.



Adam And Eve Virus: Takes a couple of bytes out of your Apple.

Congressional Virus: The computer locks up, screen splits erratically with a message appearing on each half blaming the other side for the problem.

PBS Virus: Your programs stop every few minutes to ask for money.

Ollie North Virus: Causes your printer to become a paper shredder.

Nike Virus: Just does it.

Sears Virus: Your data won't appear unless you buy new cables, power supply and a set of shocks.

Jimmy Hoffa Virus: Your programs can never be found

again.

Congressional Virus #2: Runs every program on the hard drive simultaneously, but doesn't allow the user to accomplish anything.

Kevorkian Virus: Helps your computer shut down as an act of mercy.

Health Care Virus: Tests your system for a day, finds nothing wrong, and sends you a bill for \$4,500.

George Bush Virus: It starts by boldly stating, "Read my docs...No more new files!" on the screen. It proceeds to fill up the free space on your hard drive with new files, then blames it on the Congressional Virus.

Cleveland Indians Virus: Makes your accelerated IIGS machine perform like a TRS-80.

LAPD Virus: It claims it feels threatened by the other files on your PC and erases them in "self defense."

Chicago Cubs Virus: Your PC makes frequent mistakes and comes in last in the reviews, but you still love it.

Oral Roberts Virus: Claims that if you don't send it a million dollars, its programmer will take it back.

Ed McMahon Virus: Notifies you that you just MAY have won 5 Megabytes of data if you send in your entry form right away.

(c) Copyright 1994 T/TalkNET Online Publishing and GENie. From the GENie Lamp electronic magazine, June 1994, from a message posted in the Apple II RoundTable on March 31 by D. Kerwood.

