

LA 99ers

COMPUTER GROUP

Newsletter

VOL. 6 NO. 7 JULY 1987

T O P I C S

!!!!99'FEST-WEST'88 TEXPO!!!!

John Martin and SNUG announce the dates and location for 1988 FEST-WEST. Feb. 27 and 28 Las Vegas Nevada at the Palace Station Hotel, Sahara Boulevard.

Cost information and possible air/hotel packages are being formulated now and information will be released as received.

Firm commitments of participation have been received by Regena, Genial Traveler, Bytemaster, LA 99ers, with many more expected. Mark your calendars, request your vacations, make your plans (are you listening Janie LaFlamme?). This should be a GREAT one.

I (Terrie) did it again, we used the wrong disk for Marketplace last month and once again the wrong price for at least XB:BUG was listed. Sorry about that. Should be right this month.

Last months meeting very much illustrated the power of the 99/4A. The very potent argument for XBasic set forth by Barry Traver, coupled with the fantastic strength and intricacies of Fractals, as demonstrated by Doug Moore.

Thanks to Ron Albright for his herculean effort in squelching (against formidable odds) a rather malicious rumour. Funny how the same jealous mean streaked people try variations of the same malice. Incredible.

The Clubs raffle pile has been "enriched" by the Programming Aids 1, Checkbook Manager, Beginners Basic, Draw Poker, I'm Hiding, Terry Turtle Adventures, Airline, Elementary TI, Adams Hint Book, Computer Playground and Games TIS Play. The preceding have been donated by Jerry Price of Tex-Comp in exchange for mention that he had done so.

Visitors from afar, in addition to the FEST-WEST'87 attendees, we have been lucky in meeting other area 99/4A owners. Barry Traver and family, and now Howie and Sylvia Rosenberg, and recently a gent from down under Bill Perreau. How nice. If LA is on your agenda, please contact one of us.

A comment to Steve Mehr's comments on Norm Weiss Basic class. The manual you receive along with this class is not to be believed. Norm has compiled an incredible amount of material to complement his course. I can personally testify to its girth, and value. This class should be a consideration, it will open up many areas to those of us who are a bit tenderhearted. Imagine knowing not only the whys but also the hows. There are quite a few Norm graduates among both the LA & SFV group.

A note of sympathy to Tom Freeman and family, his father-in-law passed on July 4th. The delay in this newsletter is Tom's trip to Durham.

Joy Paint 99 is picking up some well deserved steam around these parts. We have a review of it by Jerry Steinberg who is just too shy to demo it for us. Perhaps we can prevail upon Jan Williams to do that. Both of them sing the praises of this fine program now offered in our Marketplace. Buy it and try it. We ordered it for FEST-WEST but were unable to do it justice without proper demo, let's pick up the slack.

Can we become a BBS oriented club??? From none it now looks like we will come online with THREE locations. John Williams in Laguna Niguel, Danny Nelson in South Bay area and Steve Chalcraft in Norwalk. Hope we get enough Indians to support these three Chiefs. What say gang, can we develop modem fever? We know from PC Pursuiters across the USA, there is now a large number of out of state callers wandering around for a flat fee of \$25.00 per month. Many of these use the local PC Pursuit number

to call same area toll numbers without local telco charges. Believe Walt Howe in Massachusetts reduced his local toll by couple of hundred dollars. Not bad, might even get me to roam. Hopefully the topics will stay close to computer subjects, hear tell there is a bit of other stuff floating about.

We have recently received the Surgeon General's report on Aids forwarded to us by Jack Shattuck of Delaware. This is now a part of our Library and is available from Fred Moore.

A disk of programs by Galen Read was also sent to us, this by Innovative Programming, if interested, "go see Fred, go see Fred, go see Fred".

PRINTER DRIVERS

=====

by George Steffen, LA 99ers

Recently, Fred Moore, our group librarian, asked me to make a few changes to a program which would print a disk catalog on label(s) so it could be stuck right on the disk. I did it so that it worked on my printer, but then found that Fred's TI Impact Printer was slightly different from my Epson MX-80 with Dots-Perfect. So I made a few minor corrections which could be merged to adapt it for the TI. Then Fred obtained the use of a COEX-80 printer belonging to the group and wanted to use that so that he would not have to keep changing labels and paper in his TI. Again, I made some more changes and put them in the merge format. I now had three programs for the price of one.

Then, Fred obtained an Axiom system which included the Seikosha GP-550 Printer. When I attempted to correct the program for this printer, I ran into two problems. First, the Seikosha uses ASCII for numbers to be transmitted in printer commands (18="018") whereas the Epson and COEX use the character for the number (18=CHR\$(18)). Also, the printing pins on the Seikosha are spaced at 1/60th inch rather than the more common 1/72nd inch. This resulted in a slight text overlap when the labels were printed at 8 lines per inch.

Since some of the printer commands in the program used unprintable characters, I had set up the commands at the beginning of the program. The next step was to put all the information needed for the commands in DATA statements and then construct the commands from them. Since the number of characters in the command is not fixed, the first number in each statement is the number of characters in the command. I found this number valuable later when I wanted to TAB a line which included a command. Because of the different line spacing, I included a number showing lines per inch which the program uses to calculate the number of lines which will fit on one label and the instruction on how to position the labels in the printer.

Note by Tom - Having spent much of the last two weeks out of state and away from my computer (the withdrawal pains are only now beginning to go away) I haven't found much time for writing. It's all I can do to format this newsletter! I dashed off my article the night before I left - hope it isn't too murky.

One additional message to all of you who appreciate this club's attempts to aid the TI "world." We haven't yet publicized it much, but Mike Dodd's compilation of articles for the Graa Kracker, appropriately entitled Kracker Facts, is currently available from us for \$5.00. It is a must for all GraaKracker owners.

The final program is shown below with the lines which need be merged for the various printers. There are only seven lines in this program, including one REM to show what printer it is for and one DATA for the instruction to be displayed. The other DATA statements could be included in one line, but were separated for clarity. Each line is the data for one command series to be transmitted to the printer.

Line 200 gives details on the printer for which the program module was designed. It will work, of course, on any other printer which uses the same commands. However, notice that there are two Epson programs. TINYPAT/E will work on those printers equipped with GRAFTRAX Plus, but it will not work the other way around. Be sure everything is the same before using a program on a printer for which it was not designed. Six and 2/3 lines per inch (10 lines per label was the closest legible line spacing on the Seikosha.

Line 210 is the command to be sent to the printer at the beginning of the program. It resets the Top of Form (if available), sets line per inch and form length. The last number, which is not included in the character count, is the lines per inch.

Line 220 is the reset command to restore the printer to normal when the program is over.

Line 230 is the Form Feed. This is a standard included in ASCII, but was included here in case I find a printer which does use something else.

Line 240 cancels compressed print and sets emphasized if available. Emphasized was used for appearance because the compressed print normally looks darker than the normal.

Line 250 sets the compressed print and cancels emphasized if it has been set.

Line 260 is the instruction for getting the labels aligned. Room for the longest instruction was left on screen so there are some blank lines if there is a shorter instruction.

Line 370-380 and Line 590 set up the printer commands and get the other necessary parameters.

The purpose of this article is not specifically the disk catalog program, but the idea of designing programs so they are readily adaptable to various printers. This means putting each command series to be transmitted to the printer in a DATA statement and including any other information that the program might need about the printer. In this way, the only thing needing change when you change printers is the appropriate DATA statements.

```

100 REM TINYCAT - SEE LINE 2
00 FOR ENVIRONMENT
110 REM BY GEORGE F. STEFFEN
, LOS ANGELES 99er USER GROU
P
120 REM VERSION 3.0, JUNE 20,
1987
130 REM WILL PRINT DISK CATA
LOG ON 4 IN WIDE LABELS
140 REM FOLLOWING LINE SETS
SYSTEM VARIABLES: PRINTER NA
ME, DEFAULT DISK NUMBER AND
LABEL HEIGHT
150 DATA PID,DSK1,1.5
160 REM FOLLOWING DATA LINES
GIVE NUMBER OF CHARACTERS I
N PRINTER COMMAND, CHARACTER
NUMBERS, AND SOMETIMES AN A
DDITIONAL BIT OF DATA
170 REM LINES ARE: OPEN PRIN
TER AND SET LPI AND FL AND G
IVE LINES PER INCH, RESTORE
PRINTER TO NORMAL, FORM FEED
AND ITS LENGTH
180 REM SET NORMAL PRINT AND
BOLD ON, SET COMPRESSED PRIM
T AND BOLD OFF, INSTRUCTIONS
FOR GETTING TOP OF FORM ALI
GNED
190 REM LINES 200-260 SHOULD
BE CHANGED ACCORDING TO PRI
NTER YOU ARE USING
200 REM FOR EPSON MX80 PRINT
ER WITH GRAFTRAX PLUS
210 DATA 7,27,64,27,48,27,67
,12,8
220 DATA 2,27,64
230 DATA 1,12
240 DATA 3,18,27,69
250 DATA 3,15,27,70
260 DATA MAKE SURE TOP OF LA
BEL IS INCORRECT POSITION AN
D HIT ANYKEY WHEN READY.
270 REM FILE TYPES AND INSTR
UCTIONS
280 DATA DF,DV,IF,IV,PR,DO Y
OU WANT TEST PRINTING TO REGI
STER POSITION N
290 IMAGE ##### ##
##

```

```

300 IMAGE ##### AVAIL
### USED ### ###
310 IMAGE NAME ##### U
AVAILABLE ##### U
SED ##### NUMBER 8000X
320 IMAGE ##### FILES.
330 REM START PROGRAM
340 DEF N$(X)=SEG$(" "&STR
$(X)&"",LEN(STR$(X))-4*
(X=0),5)
350 DIM R$(127)
360 READ PN$,DN$,LH :: DISPL
AY AT(1,1)ERASE ALL:"THIS PR
OGRAM WILL PRODUCE A DISK CA
TALOG ON A 4 IN WIDE LABEL."
: "PRINTER NAME?":PN$
370 GOSUB 570 :: OP$=A$ :: R
EAD LPI :: GOSUB 570 :: RP$=
A$ :: GOSUB 570 :: SFF$=A$ :
: SFF=I :: GOSUB 570 :: SNP$
=A$ :: GOSUB 570 :: SCP$=A$
380 SCP=I :: FOR I=0 TO 5 ::
READ T$(I):: NEXT I :: ACCE
PT AT(6,1)SIZE(-28)BEEP:PN$
:: READ A$ :: DISPLAY AT(8,1
):A$ :: OPEN #1:PN$&".LF"
390 ACCEPT AT(9,28)SIZE(-1)V
ALIDATE("YN")BEEP:A$ :: IF A
$="Y" THEN PRINT #1:TAB(I);"
-----" :: I=I+5 :: GOTO 390
400 CLOSE #1 :: DISPLAY AT(8
,1):T$(0)
410 CALL KEY(0,K,S):: IF S<1
THEN GOTO 410 ELSE OPEN #1:
PN$,DISPLAY ,VARIABLE 70,OUT
PUT :: PRINT #1:OP$;
420 DISPLAY AT(14,1):"DISK T
O BE CATALOGUED?":NULL ENTR
Y WILL TERMINATE":DN$ :: LH=
INT(LH*LPI)! LINES PER LABEL
430 REM END PRELIMINARIES -
FOLLOWING LOOPS FOR EACH DIS
K
440 ACCEPT AT(16,4)SIZE(-1)V
ALIDATE(DIGIT)BEEP:DN$ :: IF
DN$="" THEN GOTO 550 ELSE D
N$="DSK"&DN$&". "
450 OPEN #2:DN$,RELATIVE,INT
ERNAL,INPUT :: L=0 :: INPUT
#2:A$,A,B,C :: DISPLAY AT(18

```

```

,1):USING 310:A$,C,B-C
460 ACCEPT AT(20,8)SIZE(-5)V
ALIDATE(DIGIT,UALPHA)BEEP:DS
KNO$ :: PRINT #1:SNP$:: PRI
NT #1,USING 300:A$,C,B-C,DSK
NO$
470 DISPLAY AT(22,1):USING 3
20:"COUNTING" :: FOR I=1 TO
127 :: INPUT #2:A$,A,B,C ::
IF LEN(A$)=0 THEN 490
480 R$(I)=SEG$(A$&"
",1,10)&N$(B)&T$(ABS(A))&N$(
C):: NEXT I
490 K=1 :: I=I-K :: DISPLAY
AT(22,1):USING 320:I :: LNS=
LL-2 :: PRINT #1:SCP$:: CC=
SCP ! PREPARE TO PRINT FILES
500 L=LL*3 :: IF I<L-B OR I>
L-6 THEN PRINT #1: :: LNS=LN
S-I :: CC=0
510 L=INT((I-K)/3)+1 :: ST=M
IN(LNS,L):: FOR K=K TO ST+K-
1 :: COL=1 :: FOR L=K TO ST+
2+K STEP ST ! SET UP LINES T
O PRINT ON THIS LABEL
520 PRINT #1:TAB(COL+CC);R$(
L):: COL=COL+23 :: NEXT L :
: CC=0 :: NEXT K ! GET AND P
RINT RECORDS
530 LNS=LL-1 :: PRINT #1:TAB
(1);SFF$:: CC=SFF :: K=L-ST
+1 :: IF K<I THEN 510 ! GO T
O NEXT LABEL
540 CLOSE #2 :: GOTO 440 ! F
INISHED THIS DISK
550 PRINT #1:RP$ :: CLOSE #1
:: STOP :: RESTORE PRINTER
TO NORMAL
560 REM SUB ROUTINES
570 A$="" :: READ I :: FOR J
=1 TO I :: READ K :: A$=A$&C
HR$(K):: NEXT J :: RETURN !
CREATE PRINTER COMMANDS
580 END
=====
200 REM FOR COEX-80 PRINTER
210 DATA 5,27,49,27,67,12,8
220 DATA 5,27,58,27,67,66
230 DATA 1,12

```

```

240 DATA 1,15
250 DATA 1,18
260 DATA "TURN PRINTER OFF,
MAKE SURE TOP OF LABEL IS IN
CORRECT POSITION AND THEN
TURN PRINTER ON. HIT A
NY KEY WHEN READY."
=====
200 REM FOR EPSON MX-80 (INC
LUDING TI IMPACT) PRINTER
210 DATA 7,27,48,18,20,27,67
,12,8
220 DATA 5,27,58,27,67,66
230 DATA 1,12
240 DATA 3,18,27,71
250 DATA 3,15,27,72
260 DATA "TURN PRINTER OFF,
MAKE SURE TOP OF LABEL IS IN
CORRECT POSITION AND THEN
TURN PRINTER ON. HIT A
NY KEY WHEN READY."
=====
200 REM FOR EPSON MX80 PRINT
ER WITH GRAFTRAX PLUS
210 DATA 7,27,64,27,48,27,67
,12,8
220 DATA 2,27,64
230 DATA 1,12
240 DATA 3,18,27,69
250 DATA 3,15,27,70
260 DATA MAKE SURE TOP OF LA
BEL IS INCORRECT POSITION AN
D HIT ANYKEY WHEN READY.
=====
200 REM FOR SEIKOSHA 6P-550A
PRINTER AT 6 2/3 LINES PER
INCH
210 DATA 13,27,78,27,84,49,5
6,27,15,27,90,48,48,57,6.667
220 9,27,78,27,6,27,90,48,54
,54
230 DATA 1,12
240 DATA 4,27,78,27,23
250 DATA 4,27,36,27,67
260 DATA "MAKE SURE TOP OF L
ABEL IS INCORRECT POSITION A
ND HIT ANYKEY WHEN READY."

```

DSRLNK's That Don't Work

=====

by Tom Freeman

Have you ever run across a program that didn't seem to work properly with a RAMdisk? In working, sometimes briefly, with MANY different programs, I have found that to happen under the following circumstances: the RAMdisk must be located at CRU base >1200 or greater, and the program must use sector access at some point. If you have only a Myarc RAMdisk the first condition will not obtain, since it is set at >1000, but Horizon RAMdisks have switchable CRU base settings, so if you have both you might have set the HRD at >1200, or it might be there for other reasons. The second condition obtains when the program involves disk cataloging (usually - the DSR routines are mighty slow), sector editing, or other circumstances such as XMODEM transfers in a terminal emulator.

The problem seems to occur because the disk controller card reports an error when the disk drive number, which is used in the sector access routine, is higher than the maximum number of drives that the card will handle. This is not a problem when the card to be accessed is at CRU base >1000, because the DSRLNK routine never looks at the DSR of the FDC card - it finds the RAMdisk first. It is a problem however if for instance you have a HRD emulating DSK5 at CRU base >1200. Since most programs are designed to trap errors, you cannot get past the FDC. By the way this is not a problem with the Myarc 80 track EPROM - it does not report an error.

So what do you do? Following a suggestion contained in the Docs for DM-1000 that came with my HRD, I have created a messy, but workable solution. Basically the idea is that if DSRLNK starts searching at CRU base >1200 first, goes up to >2000, which is the first non valid value, then circles back to >1000 and finally looks at >1100 (the address of the FDC) last, the FDC never has a chance to report an error before the other devices are searched. If you have the source code for a program, then it can be rewritten and reassembled, so no problem! The problem is if you have only a "program" (memory image file) or a DIS/FIX 80 file (hereafter referred to as DF80). A patch will then have to be made. So let's get to the method.

The first problem will be that you have to know where the program is resident in memory, so that the searches outlined below can be made. 1) If you have a "transparent" editor such as SBUG6, which loads into RAM at >6000 (but you need a GramCracker or Superspace for that) this is easy. Just clear out all RAM, load the program (and run it a bit), then quit by turning the computer off, (in case the program clears itself out when you quit the normal way) load SBUG6 and search where there is code instead of 00's. 2) If the file is a DF80 and is UNcompressed you can examine it in a regular editor such as E/A or TI-Writer and scan the first five bytes of each line to see where they load. 3) If the file is compressed DF80, then you will need to look at the actual file on disk with a sector editor, and look at the first 3 bytes of each sector (6 characters if you look in hex). Remember that if the first byte is '9'

then you will be looking at the actual address next, whereas if it is 'A' then the address is relocatable and must be added to >24F4 or >A000 (XB or EA). 4) Finally if the file is "program" then you can find where each section loads by looking at the third word of the first sector in each section.

All of the previous 4 methods can be avoided if you use DISKASSEMBLER on the original file on disk, as it will tell you all of the addresses used. However using DKA, or methods 2-4 above, you will still have to run the program to make sure certain areas haven't been wiped out. Now HOW do you search. If the program is located in high memory (>A000 - >FFFF) then you can use DEBUG assembled for XB, or ADR6'd in low memory, or better yet the GramCracker editor, if you have it. If the program is relocatable DF80 code then you can load SBUG after it (or DEBUG). And of course SBUG6 can always be used as well. You can also use DISK+AID or DISKU directly on the files on disk (the searches are quite fast if the files are themselves on a RAMdisk) but this can be difficult if the files are DF80 files rather than "program," since only two bytes are located contiguously.

The first goal is to find what address DSRLNK is at, the second to find an address where available text that is expendable is located. This is why the program needs to be run for a while, to make sure that the text is still there! Title screens are usually good, since you don't really need them, and if you can return to them you know they are still there. That was the problem in DISK+AID; the area where the text for the title screen is located is apparently used by other parts of the program (probably a buffer). Note that you cannot necessarily use an area outside of the program because it might be used as a buffer.

STEP 1 - Finding DSRLNK

Here is a relevant short portion of a fairly standard DSRLNK, similar to that used by many programs. [NOTE: if the program uses MG's DSRLNK-6PLLNK routine then my method will not work, as this routine uses console ROM which cannot be modified!] The indicates any number of lines of code which we don't need. The machine code in hex is also given as this is what will be used. The labels are my own - you need to find which addresses correspond to them!

```

.....
020C 0F00 A LI R12,>0F00
C30C B MOV R12,R12
1301 JEQ $+4
1E00 SBZ 8
022C 0100 AI R12,>0100
04E0 03D0 CLR @>03D0
028C 2000 C CI R12,>2000
13xx JEQ E
C80C 03D0 D MOV R12,@>03D0
.....
02E0 xxxx E LWPI xxxx

```

The code to search for is 020C 0F00. I have never seen this particular combination in any other part of any program - no guarantee though, so do a complete search. If you have been able to search in memory, then you have the address. If you have searched sectors of a "program" file then the address is computed by taking the load address found in the third word of the first sector and adding to it >100 for each sector beyond the first, subtracting >0006 (the first six bytes of the first sector are not loaded) and then adding which byte in the sector the code is located at.

In a DF80 compressed file search for (in hex) 42020C420F00. If this is unsuccessful try either half of the sequence and then examine what comes before or after on a different "line" (record). In an uncompressed file search for (in ASCII) 0020C0F00. You can then determine the address by looking at the beginning of the record (tag 9 will give an absolute address, tag A gives a relative address which must be added to >24F4 in XB or >A000 in EA) and then adding 2 for each two bytes which follow a tag B or C until you get to the code you found (tag + 2 bytes = 5 characters in an uncompressed file, = 6 characters in a compressed file viewed in ASCII format).

Now that you have the address A, the addresses for B, C, and D are computed by adding >04, >12, and >18 respectively, to A. Address E is computed by taking the xx in the JEQ after C (13xx), multiplying by 2 and adding to D (all in hex, of course).

Step 2 - The Big Patch

This patch requires 48 (>30) bytes of continuous space that can be replaced. In some cases this amount of space can't be found. Then you will have to make the patch in sections, being very careful to adjust the addresses correspondingly. I found this to be true in DISK+AID, as I had to use parts of the menu screen that I didn't care about e.g. View Grom Memory, View VDP memory etc. I will also indicate how to do this.

Let us say you find an appropriate area beginning at address Q. Here is the code you need: (1st column is address relative to Q, 2nd column is the machine code with some values being determined by this patch or the addresses found in Step 1 - these are indicated by parentheses, the 3rd column is labels, the 4th opcodes, and the 5th operands).

```
0000 020C X LI R12,Y
0002 (Y)
0004 C80C MOV R12,0C+2
0006 (C+2)
0008 020C LI R12,>1100
000A 1100
000C 0460 B 0B
000E (B)
0010 020C Y CI R12,>2000
0012 2000
0014 1302 JEQ Z
0016 0460 W B 0D
0018 (D)
001A 020C Z LI R12,V
```

```
001C (V)
001E C80C MOV R12,0C+2
0020 (C+2)
0022 020C LI R12,>1000
0024 1000
0026 020C V CI R12,>1200
0028 1200
002A 16F5 JNE W
002C 0460 B 0E
002E (E)
```

If the code has to be discontinuous, then the above addresses can be adjusted. If for example there were only >20 bytes available at Q, but another >20 bytes at address R, then the code above beginning at W would start at R. In this case the only change would be for the JEQ Z just before W. The 1302 machine code would be 13xx where xx would be (Z-(Y+6))/2. Similar changes can be made even if the code has to be divided into three parts.

Step 3 - The Little Patch

Finally 2 changes have to be made to DSRLNK itself. At A, which presently reads 020C 0F00, replace the code with 0460 (X) which translates to B 0X. At C, which reads 020C 2000, patch in 0460 (Y), translating to B 0Y. X and Y of course are the addresses from Step 2.

Now DSRLNK will do the following: when it gets to A, it will branch directly to X in the patch. The patch will immediately change the code at C back to B 0Y if it had been modified on a previous access to DSRLNK. Then R12 is loaded with >1100 and we branch back to DSRLNK at the next step (B), where the current CRU base is "closed" and >100 is added - thus we start at >1200. The code then moves down to C which is now a branch back to the patch. The current base is compared to >2000 (which is not allowed). If not there yet we branch back again to DSRLNK at D and continue. If no valid DSRR0M is found DSRLNK eventually gets back to B and then at C goes back to our patch. Eventually >2000 is reached and then the patch goes on to Z. Aha!! Now the DSRLNK code is changed again, so that the branch at C is now to V instead of Y. The current base is changed to >1000 which is where the original DSRLNK would have started. But the comparison for the "end" is made to >1200 instead, which was the first that we checked. If not there yet we again go back to the newly modified DSRLNK (hence the need to change it back again at the beginning of each access). Finally >1100 is the last one checked and if nothing is found the program goes on to E which was where the original DSRLNK went if nothing was found. Thus by making >1100, which is the floppy disk controller, last, errors produced by it are avoided!

Now all you have to do is type in the changes at the appropriate places with a sector editor (or a regular editor, if an uncompressed DF80 file). This is cumbersome in a DF80 compressed file but straightforward and much easier in a program file.

Well there it is - you'll get (pretty?) garbage where you replaced text. But it will work! If there isn't any available text to replace - sorry! Better contact the programmer!

QUADCOL/AL, AN UNDOCUMENTED ADDITION TO THE UTILITIES DISK

Recently Mike Dodd wrote assembly language routines which make QUADCOL run much faster. The main portion of the XBasic code remains the same, as do the instructions. The main exception is that you no longer have to delete the first three <LF>'s from the formatted file, as a line has been added that does three LINPUTs and ignores the input string since it is useless. This of course can be added to the previous version as well! Thus if you have also taken the precaution of adding .PL1 to the end of your original file, you may be able to go directly to QUADCOL/AL without even looking at the formatted file.

In copies of the disk sent out after 6/10/87, I have also added the following small routine, at the request of several users (and it can also be inserted into the original program):

```

161 DISPLAY AT(17,1):"TITLES          70 ELSE IF TT$="" THEN 169
?"??? TO END,~ TO SKIP Y" ::      163 ACCEPT AT(19,1)SIZE(-28)
ACCEPT AT(17,28)SIZE(-1)VAL        BEEP:TT1$ :: IF TT1$<>" THE
IDATE("YN")BEEP:AN$ :: IF AN      N TT$=TT$&TT1$
$="N" THEN 170 ELSE PRINT #2      164 PRINT #2:CHR$(14)&" "&
:CHR$(15)                          TT$&CR$&LF$ :: GOTO 162
162 ACCEPT AT(18,1)SIZE(-28)        169 PRINT #2:CR$&LF$ :: GOTO
BEEP:TT$ :: IF TT$="" THEN 1      162
    
```

This small routine allows you to put a title at the beginning of the file. It will be printed in double wide and compressed mode. If you wish true double wide then take out the CHR\$(15) in line 161, and if you wish to leave it in compressed mode take out the CHR\$(14) in line 164. The three spaces in line 164 represent a left margin of 6 which is what I usually use. You could replace the string with RPT\$(" ",LEFT/2) if you use different margins. The routine allows you to concatenate 2 lines together on one title line (ACCEPT AT only allows a 28 character input) but remember that all trailing spaces at the end of the first line are cut off by the computer. Inputting a "~" will feed a blank line, and a null input will get you to the beginning of the actual file input.

My thanks to Mike Dodd for sending me his revision.

CORRECTIONS for Utility Programs Booklet

The following corrections should be made in the documentation booklet for the Utility Programs Disk. All but the first were due to my failure to pay attention to the T!Writer Formatter's habit of erasing * and ^. The first I cannot explain. In any case, since the programs were all on disk in proper form, you shouldn't have had any problems. Sorry anyway!

P.	PROGRAM	LINE	STATEMENT	CORRECT CODE	INCORRECT CODE
2		420	2	... A\$(X);CR\$... A\$(CR\$
20	ASL/CL	460	3	ADDR=ADDR+RLFLAG*9460	ADDR=ADDR+RLFLAG60
20	ASL/CL	650	5	A=A+A2*16^(L-X)	A=A+A2 (L-X)
21	CL/ASL	320	2	... CHR\$(65+Q*17)	... CHR\$(65+Q)
21	CL/ASL	400	1	N=N-Q*9460	N=N-Q60
21	CL/ASL	440	4	N=V1*256+V2	N=V16+V2
21	CL/ASL	460	end	... N+Q*9460	... N+Q60
21	CL/ASL	580	2	P=16^(4-X)	P=16 (4-X)
22	ORIGINS	1010	4	A=A+A2*16^(4-X)	A=A+A2 (4-X)
22	HIDDEN	32762	2	Z5=Z1*256 ...	Z5=Z16 ...
22	HIDDEN	32763	4	Z3=Z1*256+Z2-65536	Z3=Z16+Z2-65536
24	CHECKSUM	190	1	Z=ASC(A\$)*256 ...	Z=ASC(A\$)6 ...

TIPS 'N Thoughts

==== == =====

by Tom Fairbairn, from MSP 99 Newsletter

A great many of us find need, from time to time, to generate documents that exceed 80 columns. This could be a combination of output files, for example, from MULTIPLAN or some other spreadsheet program that you are merging into compressed font and on wide paper (could be up to 255 columns), special reports, or possibly a compressed font style printout on conventional width (8.5 inch) paper (which could be up to 136 columns).

Now, TI-Writer has a screen width of 80 columns. In all cases, the manual that comes with TI-Writer deals with using the program to work with 80 or less print columns. No examples nor intimations are given that would indicate TI-Writer can do any more than this.

If you remember, I mentioned a bit back that there are three different sets of margins we must be concerned with when using TI-Writer: the screen margins, set by the margin/tab ruler; the formatter margins that regulate the printed margins and indenting; and the printer margins that regulate the final positioning of the result on paper.

As it turns out, the limit of characters per line permitted by the formatters is 256 (in the range of 0 to 255). So far as many printers are concerned, you may use the number of characters that will fit on a line within the margins set on the printer. Therefore, if you do not force CRs within a line to be printed, you may use however many characters the printer can handle.

The key is the settings of the formatter and printer margins. The Formatter takes the Editor file and formats each line to the length you specify with the .LM and .RM commands. If you specify margins, for example, of 10 at the left and 127 at the right, the Formatter will set up lines of 117 characters (EN: sic--actually 118) with the first character at 10 positions from the left margin that is set in the printer. So long as this long a line fits within the printer's limits, it will be entirely printed on one line by the printer.

In any event, if you are going to make use of the capability, you have to be certain that you include the necessary commands to your printer within the file. Generally, control of the printer may be accomplished with the necessary margin, font selection, and character size commands to your printer. You must also set the Formatter left and right margins appropriately.

So far, I have waltzed all around a problem that is of major importance during entry of your data into the Editor file. There is absolutely no way that the Editor can be made to work with more than 80 characters per line for the screen display. Note this isn't a limitation on how many characters per line will print, just on how many you will see on each line as you are entering data.

The easiest way to get around this is to calculate the number of lines it will take to form an even division of the number of characters you will print. If we are considering 118 characters per line, then two lines of 59 characters each can be set up in the Editor display: this means that every other line represents the end of a printed line. For 256 characters per line, use 4 lines of 64 characters each, where every fourth line represents the end of the printed line. If possible, try to use formatter and printer margin settings that are an integral multiple of two, three, or four lines of the screen per print line.

Breaking the printed line in this fashion allows you to see when you should hyphenate text entry or break off a string of data to wrap it down to the next line. It does not make for a real easy process to see what a formatted printout will look like, though. Very few word processors of which I am aware can do much better in this regard, in any event. Even the big quarter-million dollar system I use at work can't handle more than 132 columns, period, and then only if the system is specified accordingly by the sysop.

Where the ability to use 256 column line widths is useful is in the presenting of spreadsheets. Some spreadsheet print formatters as they are provided in the TI version limit the spreadsheet printout to 80 column chunks. Even if you use a wide printer carriage and compressed font, the printout is still in 80 column chunks that you have to cut and paste to make the full sheet.

By using TI-Writer and its Editor, you can create wide sheets from the spreadsheet print files, providing they can be written to a disk in DIS/VAR 80 format. You have to use the editor to read selected lines from the spreadsheet print file and format them into full length print lines under TI-Writer, then use the TI-Writer Formatter to produce the final printout. The process involves "printing" the spreadsheet output to the disk instead of the printer so that TI-Writer can read the spreadsheet document. You would probably do best to also take the hardcopy of the original sheet so that you can properly format the resulting sheet under TI-Writer.

Using a larger-than-80-column print width also permits you to use either compressed or elite type sizes in order to cram more characters per line on a conventional width page and thus use less pages. You should keep in mind, though, that the compressed font may not reproduce too well under some processes, so you want to be careful where and how you use it.

Centering will also work with the wider-than-normal line lengths. Do the centering in the normal manner, but remember that, in the case of expanded width characters, you will have to recalculate the characters-per-line approximately to the size of the characters you are using

and the actual width of the printout. A formula for this was discussed in another column that specialized in the use of the centering commands under TI-Writer.

To merge a spreadsheet printout into a wide document, we need to know how they structure their output files.

When you create a spreadsheet under any such program, you are building a document that can contain columns of information. The columns can be of any selected width, and also may be as many lines as are necessary to contain the formatted information.

As the spreadsheet program you are using formats a spreadsheet for printout, it will break the printout at any point where the full column will no longer fit within the 80-character limits of your normally 8.5-inch page. It will print the first few columns continuously until all rows are displayed; it then shifts to the next set of columns, and so on until the entire sheet is printed. To widen the printed sheet out to the full TI-Writer width, you would need to read in the first line of the first segment from the print file, then the first line of the second segment, and so on until you have read in the entire first line of the final document. You would then edit this line to use proper column spacing, then repeat the process with the second line, and so on until you have reformatted the entire print file. You would then print

the reformatted print file under the ti-writer format.

Reading of the document segments involves using the LF command under Editor, specifying the line number on the screen after which you want the new data to appear and the starting and ending line numbers of the document from which you are reading. This method allows you to cut and paste the new information into an existing document and select the data lines to be inserted all in one move.

Note that under MULTIPLAN you can set printing top and left margins, print widths, and number of print lines as desired; if you know what your TI-Writer document format will be, you can match it with MULTIPLAN print format commands. Since MULTIPLAN does have this flexibility, you would have much less editing to do under TI-Writer than you might have with other spreadsheet systems. Unfortunately, the one thing TI forgot to put into their manual on MULTIPLAN is how to enter the printer commands to change the character pitch and line spacing with the setup field of the PRINT OPTIONS command. That is another whole story however.

So that pretty well discusses the capability of wide documents with TI-Writer. Play around with this a while; you may find, once again, that getting away from using the PF function under Editor can open up a whole world of neat things you can do with the program.

The Three Musketeers == =====

by Steve Mehr

Barry Traver, Doug Moore, and Norm Weiss. What, do you say, do these three TI junkies have in common? Well, let's start with Barry Traver. Did you miss the June meeting? Too bad! What a night!

Barry was out on the west coast and was able to attend the meeting, lucky for us. Thanks to Barry, I no longer feel I have to make excuses for programming in Extended BASIC. Barry defended XB with a passion, explaining how an XB programmer can take advantage of Assembly Language within the XB environment, simply and easily. Wow! I have been using this technique for some time now, and can say that it is truly amazing what you can accomplish by tapping the power of Assembly Language within an XB program. Some XB programs look and run just like Assembly. Truly amazing! Thanks Barry, for the informative talk and for your TOTAL support of the TI community. Wish I had a tape recorder at the meeting.

Now we come to Doug. I have been interested in Binary Cellular Automata for a long time, and have even written a program long ago to produce them. I have also "played around" with Steve Langguth's Fractal Explorer program and was amazed at the complexity of its output. So along comes Doug, and at the June meeting, puts Fractal Explorer through its paces. His "playing around" involved creating around 10 or 15 Fractals, which I don't have to remind you take from 1 to 4 hours to create...

EACH! To research a subject like he has requires a strong desire to share the fruits of his effort with everyone. Thanks Doug, we all learned a lot and were quite entertained by your demo.

Someone has to be last Norm, sorry. "Where were you in '82?" I can't say where I was because I can't remember. But I DO know where Norm was, Olympic Sales buying his first computer... Guess! O.K., so what about Norm, what makes him so special. Let me tell you. Norm is the ONLY one I know, (and when I say "ONLY", I'm talking about out of the 6 Users Groups I'm a member of, so I think I talk from experience), who has taken all his personal notes from teaching himself to program along with the experience he gained along the way, and started teaching BASIC classes. I didn't say C, Assembly, Forth, or any other lower level language, I said BASIC, the language we all too often ignore. Didn't we all, or almost all of us, start there? I know I did. If we are to continue to encourage people to program for the TI, don't we have to try and "Keep The Dream Alive"? Well, Norm sure does with his classes. For more information on his classes, please call Norm at (818) 894-7559. Thanks Norm, you're a true TI GUY!

P.S. Terrie, I might be persuaded to make this a regular column. (EN: Steve, your wish is my command.)

Did you know that...?

by Chick De Marti



PASSWORD anyone?

Have you ever wondered how they create that PASSWORD routine? You know! You type in the word but nothing shows up on the screen? I did and decided to try to create one. The following is my version.

```

10 REM *****
20 ! - PASSWORD Routine -!
30 ! ( Like the BBSs) !
40 ! !
50 CALL CLEAR
60 DISPLAY AT(3,1):"Enter password ###"
70 GOSUB 1000+(ACCEPT AT)+
80 IF A$="CDM" THEN 100
90 CALL SOUND(500,1000,5):: A$="" :: GOTO 70
100 DISPLAY AT(5,1):"Hi Chick, welcome to":"OUR BOARD"
110 END
1000 FOR I=1 TO 3
1010 DISPLAY AT(3,15+I)SIZE(-1):" "
1020 FOR DX=1 TO 10 :: NEXT DX
1030 DISPLAY AT(3,15+I)SIZE(-1):" "
1040 FOR DX=1 TO 10 :: NEXT DX
1050 CALL KEY(0,K,S):: IF S=0 THEN 1010
1060 A$=A$&CHR$(K)
1070 NEXT I
1080 RETURN

```

<*><*><*><*><*><*><*>

HIGH_LOW

Here's a short routine to receive random numbers between an indicated High and Low.

```

10 PRINT "Enter lowest and highest number"
20 INPUT "Seperated by a comma ":N1,N2
30 FOR I=1 TO 10
40 R=INT(RND*(N2-N1+2)+N1)
50 PRINT R;
60 NEXT I
70 PRINT :: INPUT "Another (Y/N)":YN$
80 IF YN$="Y" THEN 10 ELSE END

```

<*><*><*><*><*><*><*>

PLATO anyone?

A friend of mine has the full compliment of PLATO programs. I was playing around with one of the programs. In PLATO's "MAKE SENTENCE", level - beginner, try:
"THE BICYCLE CARRIES THE BOY"
SURPRISE! SURPRISE!!

CATALOGER

The next item was found in the INTERFACE newsletter. I use one like this as a subroutine in several of my programs. I use two systems to activate it. One is at any INPUT or ACCEPT AT command I enter a line "IF Q\$="C" THEN GOSUB 17000". The other is part of an error routine. "ON ERROR GOTO 10000" and at line 10000 I use: PRINT "Check the disk in drive two or press "C" to read the catalog" :: INPUT Q\$:: IF Q\$="C" etc. It might also be entered via a menu.

```

100 ! FAST LOOK !
110 CALL CLEAR
120 OPEN #1:"DSK1.",RELATIVE,INTERNAL,INPUT ,FIXED
130 FOR I=1 TO 127
140 INPUT #1,REC I:A$
150 IF A$="" THEN 180
160 PRINT I:A$
170 NEXT I
180 CLOSE #1
190 END

```

(Try changing line 160 to: 160 PRINT I;A\$,)

<*><*><*><*><*><*><*>

PUZZLE_TIME

Mathematically use the numbers...1,2,3,4,and 5 so it will equal 6. No! Parenthesis' are NOT allowed

(This was given to me by Mike Lessinger, a young piano student of mine. Thanx Mike.

<*><*><*><*><*><*><*>

A POINT TO PONDER by Dick Daus

"Have you ever thought about what a Users Group really is?" ... Ans. "A place to share ideas and information."
WE ARE ALL MEMBERS! We all have something to share!!

From TIGAR CUB:

Worth repeating...at least once a year.
A great debugging aide.

```
10 ! Turns all numeral and
punctuation white! by Harry
Wilhelm in Twin Tiers User
Group Newsletter
20 ! Turn it on by CALL LOAD
(-31804,63):: Turn it off by
CALL LOAD(-31804,0)
100 CALL INIT
110 CALL LOAD(16128,2,224,38
,0,2,0,8,17,2,1,63,36,2,2,0,
3,4,32,32,36,2,224,131,192,3
,128)
120 CALL LOAD(16164,240,240,
240)
130 CALL LOAD(-31804,63)
```

<*><*><*><*><*><*><*>

From "LITI" newsletter of Long Island,
the President's corner:

Goodbye, DANNY MICHAEL!

Danny Michael (SCREEN DUMP, NEATLIST, GK
UTILITY) has announced in the Shoals 99er
Newsletter that he is selling his TI and
buying an IBM> The TI community will miss him.

<*><*><*><*><*><*><*>

ANOTHER HURTLE

This article was download and forwarded
to us by Danny Nelson. I believe it to be
in an area where we can help. Either make
photocopies of the letteres of just and the
copies we made for you...but MAIL THEM!

PHONE MAKE

PHONE MAKE 1 2 3

From the time that Stu Olson Released
MASS-TRANS Version 4.0, the most asked
question was, "How do you get the PHONEMAKE
TO WORK?". Well we are now at version 4.2
(The last one according to Stu), and the
question is still being Asked. Just follow
my instructions and you should have no
problems. To put it in the words of the
TV DETECTIVE HUNTER, "It works for me!!".

- 1) ... Load PHONEMAKE (RUN"DSK1.PHONEMAKE").
- 2) ... Press #1 (Leave Disk in drive).

- 3) ... Press #2 to make sure the book is Clear.
- 4) ... Press #3, then press the letter for the position at which you want the name to appear on your Phone Book MENU. i.e. "A".
- 5) ... Type the name of the BBS or the person you wish to Call. <ENTER>.
- 6) ... Type YOUR Modems Attention Dial Tone Command, and the number you wish to call. i.e. ATDT9238433 for local calls. ATDT1818923433 for calls to other Service areas.
- 7) ... When you have completed one phone book or any part of it, you may review it by typing "N" at the 'Change Another(Y/N)'. Press #2 at the main menu.

8) ... Press #5 and assign YOUR MODEM COMMANDS PER THE MENU. At CONNECT 300. i.e. type ATZ

9) ... If you are on PC-PURSUIT, Press #6 and enter your ID,PW,Baud Rate and the Area Code you wish to call. This will set up a file on the disk for each area code that you enter.

10) ... DO NOT REMOVE THE DISK YET!!!!. Press #7. When the Disk stops, you may remove it and Boot your Disk Manager. Find the file named "PHONE". Change the name of this file to "PHONE1".

THAT IS ALL THERE IS TO IT.

If you want to EDIT the "PHONE1" file, you only have to remove the "1" and run the PHONEMAKE file using #2 and #3 to look at and change Names and Numbers.

GOOD LUCK. Danny Nelson

<*><*><*><*><*><*><*>

Answer_to_puzzle

$$1 \times 2 + 3 - 4 + 5 = 6$$

Well...I'm out of coffee
see you next month ...

Chick

Passed on from Capricorn BBS.

To all concerned BBS users:

On June 10th, the FCC proposed another rule change that would radically change BBS and computer systems use as we know it. This rule change would assess an access charge for all business AND residential computer users amounting to between \$5-\$10 an hour. If the thought of losing your ability to use your modem bothers you, now is the time to act! I have prepared a FORM LETTER for you to use to contact the FCC and express your feelings. Feel free to modify/enlarge/alter these letters to suit your tastes. But whatever you, PLEASE SEND THESE LETTERS.

This is the second challenge to our freedom in just a few months. LETS SHOW THEM WE MEAN BUSINESS THIS TIME!

Thank you for your concern and willingness to attempt to block this threat to BBS and computer communications systems.

Originated by --> Vince Castelli
and thanks to SYSOP, Chesapeake RBBS
(301) 267-4930 (24 hrs)
300,1200,2400 baud

July , 1987

Honorable Dennis R. Patrick
Chairman, Federal Communications Commission
1919 M. Street, N.W.
Washington, D.C. 20554

Re: "Interstate Access Charges Exemption for Enhanced Service
Providers" FCC Docket 87-208

Dear Sir:

I am strongly opposed to the subject rule change. I believe that it would provide an unfair advantage for the regional Bell telephone companies at the expense of both the current communications network suppliers and current computer users.

I request that this proposed rule change be abolished and that the current system be maintained.

Also, please do not consider any other changes of this sort in the near future.

Sincerely,

July , 1987

Honorable James H. Quello
Commissioner, Federal Communications Commission
1919 M. Street, N.W.
Washington, D.C. 20554

Re: "Interstate Access Charges Exemption for Enhanced Service
Providers" FCC Docket 87-208

Dear Sir:

I am strongly opposed to the subject rule change. I believe that it would provide an unfair advantage for the regional Bell telephone companies at the expense of both the current communications network suppliers and current computer users.

I request that this proposed rule change be abolished and that the current system be maintained.

Also, please do not consider any other changes of this sort in the near future.

Sincerely,

July , 1987

Honorable Patricia Diaz Dennis
Commissioner, Federal Communications Commission
1919 M. Street, N.W.
Washington, D.C. 20554

Re: "Interstate Access Charges Exemption for Enhanced Service
Providers" FCC Docket 87-208

Dear Sir:

I am strongly opposed to the subject rule change. I believe that it would provide an unfair advantage for the regional Bell telephone companies at the expense of both the current communications network suppliers and current computer users.

I request that this proposed rule change be abolished and that the current system be maintained.

Also, please do not consider any other changes of this sort in the near future.

Sincerely,

Honorable Mimi Danforth Dawson
Commissioner, Federal Communications Commission
1919 M. Street, N.W.
Washington, D.C. 20554

Re: "Interstate Access Charges Exemption for Enhanced Service
Providers" FCC Docket 87-208

Dear Sir:

I am strongly opposed to the subject rule change. I believe that it would provide an unfair advantage for the regional Bell telephone companies at the expense of both the current communications network suppliers and current computer users.

I request that this proposed rule change be abolished and that the current system be maintained.

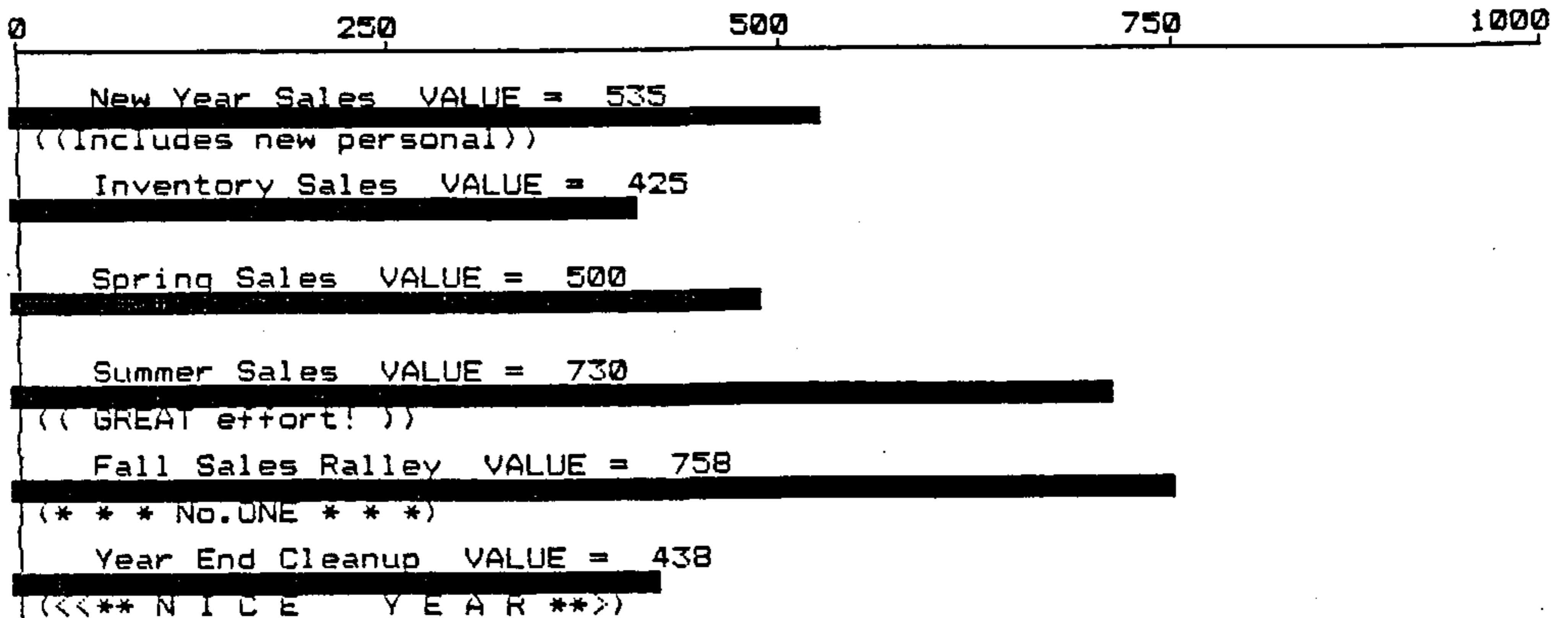
Also, please do not consider any other changes of this sort in the near future.

Sincerely,

We have in our library, an excellent Bar-graphing program. It is called BARGRAPHER. This sample was printed using an EPSON FX80 and I am sure it will work equally as well on the TI with Graphx. It comes with many options. Main Title, sub-title, max value, Number of bars (up to 20), and Name,value & comments for each bar. Give this one a try ...you'll like it. Chick

SAMPLE BAR CO.

Sales of BARS (per 2 Months)



RANDOM WORD GENERATOR

There are times, when trying to write an adventure game, you at a loss for unique and appropriate names for either your monsters or your heros. My first try was by hand (or I should say, by hand). But this took much too long...and besides, was no fun! So I wrote this program. I was rewarded with much more than I bargained for. Not only did I come up with acceptable names, but I also created an appropriate foreign language I could teach my characters. Here are some examples:

MONSTERS: DIHA-PIXE (a loveable rascal). An XORTI (a HUGH bat) The OZA PU, the DIWU KAVA and the ZULU QUYE, are strange vampire like creatures.

Of course, a GOTOAUNA, a MUWAROLO as well as the ZENIBUBA need no explanation.

OUR HEROS ARE: ATONO HAZODABO (a 6'6" adonis)
 MEYUVI AADA (the clown)
 The princess CINA EBULAVA
 Her friend VIVI LUWAJADE
 Emperor HUZD MAJEDESA
 Doctor ZABERUHI ... and of course,
 ANOGE SU DAH, the wizard.

"XUME ACIYI AJAEI KOHEMIMA", shouted the evil Zenibuba. As they carried my friend away he pleaded, "JEYA, JEYA !" But the EFUCO (Cad) only sneered back, " FALA PEPIJE TETUJUZU !"

Other not so obvious words are:

A "FUFEUCUPO" is a clown, an "AJOKI" is a pun. "CINE MEYALEDI" is a beautiful woman and "ODODO TUEA" is a delicious fruit from, as you know, the island of OZAPU

The program will print 30 rows of words in 3 columns. This can be changed in line 90 and if you don't want a printed copy, REM out lines 80,200,210,220 and 230. Also shorten line 280 to ... 280 END.

I hope you enjoy using this program as much as I did writing it.

<*> <*> <*> <*> <*> <*> <*>

```

10 !-----! !099
20 ! * RANDOM WORD GENERATOR * ! !092
30 ! ! !004
40 ! -- by Chick De Marti -- ! !219
50 ! ! !004
60 REM RANDOMIZE !099
70 FOR I=1 TO 5 :: READ V(I):: NEXT I !194
80 OPEN #2:"PIO" !254
90 FOR WORD=1 TO 90 !100
100 R=INT(RND*10)+1 :: IF R<4 THEN GOSUB 290 :: P=2 !142
110 FOR PHRASE=1 TO P !008
120 IF R>5 THEN X=2 ELSE X=1 !230
130 FOR Y=1 TO X !166
140 C=INT(RND*26)+65 :: IF C=65+C=69+C=73+C=85 THEN 140 !042
150 NAME$=NAME$&CHR$(C):: GOSUB 290 ! vowel !075
160 NEXT Y !239
170 GOSUB 290 ! add a vowel between phrases !045
180 NEXT PHRASE !089
190 CT=CT+1 :: PRINT NAME$,!189
200 ON CT GOTO 210,220,230 !013
210 PRINT #2:"Name #";WORD;" ";NAME$:: GOTO 240 !212
220 PRINT #2:TAB(30);"Name #";WORD;" ";NAME$:: GOTO 240 !030
230 PRINT #2:TAB(60);"Name #";WORD;" ";NAME$ :: CT=0 !124
240 NAME$="" ! prepare for new name !238
250 NEXT WORD :: P=3 :: NAME$="" !203
260 PRINT : !"Press Q to quit";: INPUT Q$ !125
270 IF Q$="Q" THEN 280 ELSE CALL CLEAR :: GOTO 90 !223
280 CLOSE #2 :: END !165
290 NAME$=NAME$&CHR$(V(INT(RND*5)+1))!149
300 RETURN !136
310 DATA 65,69,73,79,85
    
```

LI TOPICS

== =====

by Howie Rosenberg

REMINISCING.. Shortly after TI made the PCODE card available in a plug in version, I discovered, from a review in 99er magazine, that on power up, control was taken over by the card and if one wanted to run anything but PASCAL one would need wait for a minute or so. This and a rather healthy price made me and many others decide to forgo PASCAL. Shortly afterward the PCODE card was modified so that by setting a switch one could bypass the card. Both of my sons had studied Pascal in school. Indeed Pascal, for a long time, was the language used in virtually all schools which taught programming. I asked my sons a number of questions about the language to aid me in deciding if I would make the investment. My older son wrote the following "poem", ostensibly to show me the syntax, which, while it may not teach you much about Pascal code, may be of some interest.

HAPPINESS by Craig Rosenberg

```
PROCEDURE HAPPINESS ;
begin while unhappy do
  begin
    relax ;
    smile ;
    unwind ;
    think of those who love you ;
  end
end [happiness]
```

```
PROCEDURE fun ;
begin while bored and restless do
  begin
    be friendly ;
    be active ;
    be energetic ;
  end
end [fun]
```

```
PROCEDURE love ;
begin repeat show others you love and respect them ;
  care about other people ;
  keep being sensitive to feelings ;
until the end of time
end [love]
```

```
begin
REPEAT
  happiness ;
  fun ;
until cheered up
repeat
  love ;
until the end of time
```

FORTH BITS 7
On Defining Words

Defining words are simply FORTH words which are used to create new definitions. The simplest example is the word : (colon). Colon creates a new dictionary entry consisting of the next word in the text stream and containing the code which follows. The word is so important that it is introduced on page 2 of Starting Forth and probably most other introductory works on the language. Just a reminder, if you felt a little puzzled as to why I mentioned such a trivial fact, then keep reading! Otherwise, please, read Brodie's book prior to these articles. It will certainly teach you a lot more

about the language than these bits are intended to do! There are other simple defining words ; CONSTANT VARIABLE and CREATE. (note again the absence again of punctuation in the last sentence. Since , and . are perfectly valid FORTH words I sometimes find it difficult to be grammatically correct and at the same time unambiguous). Several other useful Defining Words exist in the version of FORTH used in Starting Forth which are not available in the TI version namely 2VARIABLE and 2CONSTANT which simply define a new double length variable and constant respectively. Perhaps the most useful defining word in TI FORTH is the word <BUILDS. <BUILDS when used in conjunction with the FORTH Compiler word DOES> enables creation of new compiler words. The TI FORTH word <BUILDS is equivalent to the Brodie CREATE. The TI word CREATE may have uses. I have not really found any, <BUILDS does fine for all applications I have tried.

The word <BUILDS is used in a structure : NEWWORD <BUILDS (FORTH words go here) DOES> (FORTH words go here) ; The two sets of FORTH words are optional. The series of words after <BUILDS specifies what is to be done during compile time. In other words the word NEWWORD which will be a defining word can have compiling characteristics that we specify. The words after DOES> are executed during run time, ie the code that the new compiler word places in the dictionary entry the same way that : does. A few examples should illustrate the above as well as the power of this much neglected capability unique to FORTH

: NUMWORD <BUILDS , DOES> @ . ;

The word NUMWORD when executed will compile a new definition into the dictionary (the word which follows NUMWORD and then compile the top number on the stack into the dictionary entry. On execution the new word will print the previously stored value. Thus 3 NUMWORD THREE creates a new dictionary word THREE. Typing in THREE will now print out 3.

: LOADER <BUILDS , DOES> @ LOAD ;

: BLOADER <BUILDS , DOES @ BLOAD ;

These two words enable loading screens or BSAVED applications with a single word. Thus 50 BLOADER MYGAME will allow the single word MYGAME to run an application called MYGAME from screen 50.

Finally those two missing words 2VARIABLE and 2CONSTANT can easily be created as shown in appendix C of your TI FORTH manual.

An extremely powerful function indeed is the ability to change the function of the compiler so easily!

HOPE TO SEE YOU!

I'm heading west tomorrow (July 8) for a rather hectic tour of California from San Diego to San Francisco. All this will take place after a small research project I have planned on the subject of probability and statistics to take place in Las Vegas. I hope to see some old TI friends as well as meet a few that I have heard about and never met. Unfortunately I will just miss the LA group meeting. Barry Traver seems to have planned his vacation a bit better than I did. Anyway I had to struggle to meet an early deadline for this column in the face of threats from Teresa which are too horrible to repeat. I hope to meet a few of you in person during an all too brief stay in Los Angeles.

QUADCOLUMN, REVISITED IN ASSEMBLY

=====

by Mike Dodd

Tom Freeman's QUADCOL program, published in the May 1986 LA TopIcs, is an excellent program for converting text to multiple columns. It had only one drawback: it was slow. Now, with the following assembly language subprogram, QUADCOL will read a two-column page off of a disk in mere seconds. First, you'll need to type in the assembly language program with the Editor/Assembler. Assemble it with the R option with the object file named QUADCOL/O. Now go into Extended Basic, and load QUADCOL. Type the following changes:

```

90 CALL PEEK(8194,X,Y):: IF X#256+Y<>10472 THEN CALL INIT
T :: CALL LOAD("DSK1.QUADCOL/O")
230 CALL LINK("COL",X,A$( ),Y,C( ),X1,X2,X3,CL,COL( ),COLUMN
N,EOFFLG):: IF EOFFLG THEN 350
240 ! **DELETED LINE**
250 ! **DELETED LINE**
260 ! **DELETED LINE**
270 ! **DELETED LINE**
280 ! **DELETED LINE**
290 ! **DELETED LINE**
300 ! **DELETED LINE**
310 ! **DELETED LINE**
320 ! **DELETED LINE**
350 A$(X),A$(X+1),A$(X+2),A$(X+3)=" :: EX=0 :: FOR Z=1
TO X :: EX=EX+C(Z):: IF Z-EX=INT((Y+COLUMN-1)/COLUMN)THE
M X1=Z :: IF COL(3)=0 THEN 390
    
```

For lines that read ! **DELETED LINE**, type the line number and press ENTER. Now re-save QUADCOL to disk. Line 90 will load the object file called QUADCOL/O off drive 1 if it is not loaded. It is possible (although VERY unlikely) that it may think it is loaded when it is not yet. If this happens, merely break with FCTN 4 and type CALL INIT :: RUN <ENTER>

The directions for using QUADCOL are the same.

I have also found a way to make running QUADCOL a little easier. Instead of counting the number of lines for the PL command, include a .PL32000 command at the start of your document. Then type .PL1 at the end of the document. The formatter will then suppress ALL form feeds, and will immediately stop at the end.

Thanks to Craig Miller and Doug Warren of MG for the use of their GPLLNK and DSRLNK subroutines, without which this program would have been impossible to write.

* QUADCOL assembly subprogram
 * written by Mike Dodd, 87.0315

```

TITL 'QUADCOL update - by Mike Dodd'
IDT 'QUADCOL'
DEF COL
    
```

```

* EQU
* XB
NUMASG EQU >2008      numeric assignment
NUMREF EQU >200C      numeric reference
STRASG EQU >2010      string assignment
XMLLNK EQU >2018      XML link
VSBW EQU >2020        VDP single byte write
    
```

```

VMBW EQU >2024        VDP multiple byte write
VSBW EQU >2028        VDP single byte read
VMBR EQU >202C        VDP multiple byte read
ERR EQU >2034        XBasic ERRor routine
* system
FAC EQU >834A
    
```

```

* DATA
* constant
D1 DATA >0001
* variable
X DATA 0
Y DATA 0
X1 DATA 0
X2 DATA 0
X3 DATA 0
CL DATA 0
COLUMN DATA 0
EOF DATA 0
DSRADD DATA >0000
VDPBUF DATA >0000
    
```

```

* BYTE
H01 BYTE >01
H0A BYTE >0A
EVEN
    
```

```

* buffers
MYWS BSS >20
STRLEN BYTE >00
STRBUF BSS >FF
EVEN
    
```

```

* Get number
* access with:
* BL @GETN reads from
* DATA parameter# non-array
* -OR-
* BL @GETNUM
* R1=parameter# - reads from non-array
* -OR-
* BL @GETNM1
* R0=array element number
* R1=parameter number
*
    
```

```

* OUT: FAC=integer number
GETN MOV #R1+,R1
GETNUM CLR R0
GETNM1 BLWP @NUMREF
LWPI >83E0
BL @>12B8
LWPI MYWS
RT
    
```

```

* write number
* access with
* BL @WRTN writes to non-array
* DATA address of number
* DATA parameter number
* -OR-
* BL @WRTNUM
* R1=parameter number
* FAC=integer number
* -OR-
    
```



```

* BL @WRTNM1
* R0=array element number
* R1=parameter number
* FAC=integer number
WRTN  MOV  *R11+,R0
      MOV  *R0,@FAC
      MOV  *R11+,R1
WRTNUM CLR  R0
WRTNM1 BLWP @XMLLNK
      DATA >20
      BLWP @NUMASG
      RT
* DSRLNK - by M6
* requires GPLLNK
* 186 bytes, including DSRLNK & GPLLNK
DSRLNK DATA DSRMS,DLINK1
DSRMS  EQU  $
DLINK1 MOV  R12,R12
      JNE  DLINK3
      LWPI >83E0
      MOV  @>50,R4
      BL  *R4
      LI  R4,>11
      MOVB R4,@>402(R13)
      JMP  DLINK2
      DATA 0
      DATA 0,0,0
DLINK2 MOVB @>83E9,@>402(R13)
      MOV  @>166C,R5
      MOVB *R13,@DLINK5
      INCT @DLINK4
      BL  *R5
      LWPI DSRMS
      LI  R12,>2000
DLINK3 INC  R14
      MOVB *R14+,@>836D
      MOV  @>8356,R3
      AI  R3,-8
      BLWP @GPLLNK
DLINK4 BYTE >03
DLINK5 BYTE >00
      MOVB @DSRMS+7,@>8C02
      MOVB R3,@>8C02
      SZCB R12,R15
      MOVB @>8800,R3
      SRL  R3,5
      MOVB R3,*R13
      JNE  DLINK6
      COC  @>837C,R12
      JNE  DLINK7
DLINK6 SOCB R12,R15
DLINK7 RTWP
* GPLLNK - by M6
* 70 bytes
GPLLNK DATA GLNKWS
      DATA GLINK1
      DATA GLINK2
      DATA >176C
      DATA >50
GLNKWS EQU  $->18
      BSS  8
GLINK1 MOV  *R11,@>83E8
      MOV  *R14+,@>83EC
      MOV  @>200E,R12
      MOV  R9,@>200E
      LWPI >83E0
      BL  *R4
      MOV  @GLNKWS+20,@>8302(R4)

```

```

      INCT @>8373
      B  @>60
GLINK2 MOV  @>166C,R4
      BL  *R4
      LWPI GLNKWS
      MOV  R12,@>200E
      RTWP
* DSR access subroutine
DSR  MOV  @DSRADD,@>8356  set address of PAB
      BLWP @DSRLNK        access DSR
      DATA 8              code for normal file I/O
      JEQ  DSR1            if error then DSR1
      RT                  return
DSR1 MOV  @DSRADD,R0      get address of length byte
      AI  R0,-8            point to error byte
      BLWP @VSBR           get it
      SRL R1,13           shift out other flags
      CI  R1,5             is it an I/O error?
      JNE  IOERR           no-give an error
      MOV  @D1,@EOF        indicate EOF
      B  @RET              return to Basic
IOERR DEC  R0             point to start of PAB
      MOV  R0,@>831C       set PAB pointer
      LI  R0,>2400         I/O error
      BLWP @ERR            give the error
IOERR1 LI  R0,>2200       file error
      BLWP @ERR            give the error
* START OF MAIN PROGRAM
* format:
* CALL LINK("COL",X,A$( ),Y,C( ),X1,X2,X3,CL,COL( ),COLUMN,
*   EOFFLG)
COL  LWPI MYNS            load main workspace
* find file number 1
      MOV  @>833C,R5        get address of first PAB
COL1 MOV  R5,R0           get address in R0
      JEQ  IOERR1          if 0, then give FILE ERROR
      INCT R0              point to file number
      BLWP @VSBR           get it
      CB  @H01,R1          is it file #1?
      JEQ  COL2            YES!
      DECT R0              back up to next link
      BLWP @VSBR           get first byte in MSBy R1
      MOVB @>8800,R5        get LSBY in MSBy R1
      SWPB R5              shift
      MOVB R1,R5           put MSBy in MSBy
      JMP  COL1            try next file
COL2 AI  R5,13            point to length byte
      MOV  R5,@DSRADD      move to DSRADD
      AI  R5,-7            back up to addr of VDP buffer
      MOV  R5,R0           VDP address to read
      LI  R1,VDPBUF        place to put it
      LI  R2,2             2 bytes
      BLWP @VMBR           read from VDP
* clear variables
      CLR  @X
      CLR  @Y
      CLR  @X1
      CLR  @X2
      CLR  @X3
      CLR  @EOF
* get CL and COLUMN from XBasic
      BL  @GETN            get a number
      DATA 8              8th parameter
      MOV  @FAC,@CL        move to CL
      BL  @GETN            get a number

```

```

DATA 10      10th parameter
MOV @FAC,@COLUMN move to COLUMN

* start of Basic conversion
*
* 230 X=X+1 :: LINPUT #1:A$(X):: B=POS(A$(X),LF$,1):: IF
* B THEN A$(X)=SEG$(A$(X),1,B-1):: Y=Y+1 :: C(X)=0
* ELSE C(X)=1
*
* X=X+1
L230 INC @X
* LINPUT#1:A$(X)
BL @DSR      read a record from the disk
MOV @DSRADD,R0
AI R0,-4    point to char count byte
BLWP @VSRB  get it
MOVB R1,@STRLEN write to string length
SRL R1,8    put in LSBy
MOV R1,R2   copy to R2 for VDP read
MOV @VDPBUF,R0 address of VDP buffer
LI R1,STRBUF string buffer
BLWP @VMBR  read from VDP
* B=POS(A$(X),LF$,1) ! LF$=CHR$(10)
* R3=B
CLR R3      0
L230A CB *R1+,@H0A is it CHR$(10)?
JEB L230B  yes!
AB @H01,R3 no - add to length
DEC R2     length
JNE L230A  still more characters
* since B=0,
* C(X)=1
MOV @D1,@FAC move 1 to FAC
JMP L230B1  write number
* since B=1,
*
* A$(X)=SEG$(A$(X),1,B-1)
L230B MOVB R3,@STRLEN reset length
* Y=Y+1
INC @Y     +1
* C(X)=0
CLR @FAC   move 0 to FAC
* set C(X)=FAC
L230B1 MOV @X,R0    get X
LI R1,4     4th parameter
BL @WRTNM1 write it to XBasic
* set A$(X)=string
MOV @X,R0    get X
LI R1,2     2nd parameter
LI R2,STRLEN address of string length byte
BLWP @STRASB write to string
* we won't run line 240 PRINT X;Y

* 250 IF X1<>0 THEN 270
L250 MOV @X1,R0 does X1=0?
JNE L270    no - GOTO 270
* 260 IF Y=CL THEN X1=X :: GOTO 310
L260 C @Y,@CL does Y=CL?
JNE L270    no - GOTO 270
MOV @X,@X1  X1=X

```

```

JMP L310      GOTO 310
* 270 IF X2<>0 AND COL(3)<>0 THEN 290
* or IF X2=0 OR COL(3)=0 THEN 280 ELSE 290
L270 MOV @X2,R0 does X2=0?
JEB L280     yes - GOTO 280
LI R0,3     3rd array element
LI R1,9     9th parameter
BL @GETNM1  get number
MOV @FAC,R0 does COL(3)=0?
JNE L290    no - GOTO 290
* 280 IF Y=2*CL THEN X2=X :: GOTO 310
L280 MOV @CL,R0 get CL
SLA R0,1    *2
C R0,@Y     does 2*CL=Y?
JNE L290    no - GOTO 290
MOV @X,@X2  X2=X
JMP L310    GOTO 310
* 290 IF X3<>0 AND COL(4)<>0 THEN 310
* or IF X3=0 OR COL(4)=0 THEN 300 ELSE 310
L290 MOV @X3,R0 does X3=0?
JEB L300     yes - GOTO 300
LI R0,4     4th array element
LI R1,9     9th parameter
BL @GETNM1  get number
MOV @FAC,R0 does COL(4)=0?
JNE L310    no - goto 310
* 300 IF Y=3*CL THEN X3=X
L300 MOV @CL,R0 get CL
LI R2,3     3
MPY R2,R0   R1=CL*3
C @Y,R1     does Y=3*CL?
JNE L310    no - goto 310
MOV @X,@X3  X3=X
* 310 IF Y<COLUMN*CL THEN 230
* note - didn't include EOF check, since the input
* record section handled that
L310 MOV @COLUMN,R0 get COLUMN
MOV @CL,R2   get CL
MPY R2,R0   R1=COLUMN*CL
C @Y,R1     is Y<COLUMN*CL?
JL L230     yes - GOTO 230
* it's time to get out of here!
* now, we have to write a LOT of variables back to XB
RET BL @WRTN write
DATA X,1    X,1st
BL @WRTN   write
DATA Y,3    Y,3rd
BL @WRTN   write
DATA X1,5   X1,5th
BL @WRTN   write
DATA X2,6   X2,6th
BL @WRTN   write
DATA X3,7   X3,7th
BL @WRTN   write
DATA EOF,11 EOF,11
LWP! >83EB load GPL workspace
B @>6A    return to GPL interpreter

END

```

JOYPAINT AND JOYPAINT PAL REVIEWED

=====

by Jerry Steinberg, LA 99ers

Let me begin this review by telling you that I own just about every graphic program that has come out for the 99/4A, and I have come to a very interesting conclusion. I have been asked many times, "which graphic program do you think I should buy" ... I have arrived at the perfect answer.. You should buy all of them .. I have yet to see a graphics program that does not have some facility that the others lack ... If however somebody said to me, "I can only afford to buy three or four programs, then I would have to say, "Buy T.I. Artist, Graphx, and Joypaint 99 and Pal."

One big reason for my choice of these four is that the files from one can easily be used with the others, thereby giving you a flexibility not available from any single one. If money is not lacking I would also recommend Dave Rose's CS6D Program... But now to get down to the review of Joypaint in particular... In my estimation this one is just about the best.. It has most of the features all the others have with some added ones that are breathtakingly original.. Speaking broadly before I get into details let me say this; Joypaint is the easiest to use of all the others and has many features that none of the rest have.. For example, you have an assortment of pulldown windows that enable you to easily select with your joystick a myriad of functions, some of which are entirely new to graphic program users.

First of all.. How many times have you had to save a picture to disk so that when you add the next lines you can recall the old picture in case you make a mistake?.. You don't have to do that with Joypaint... If you make an error all you need do is use the undo function provided, and the last thing you did will be undone.. This of course is a miraculous time saver... But this is only one of the miracles this program affords.. Another is the "Smart Fill".. A fill that works better than any I have seen... A little ball skirts the perimeter of the area you wish to fill, and then proceeds to do so.. It wanders about and looks for unfilled areas so it can fill them... Also the fill target is not a cross that is so big you can't fit it into what you want to fill, but a sign that says "fill" with a dot below it one pixel large, so that you can fill the tiniest of areas.. No doubt about it.. This is the best fill I've seen.

Now lets talk about textures.. You have your choice of twenty six available and if you say that's not enough..O.K. There are provision for you to make your own in any quantity and save them to disk.. So much for textures.. But the real miracle of this program is the function called Spray Can.. This is another of these great choices which are made by choosing an icon just like in T.I. Artist. The spray can allows you to take any texture and spray paint with it... The leaves in the accompanying graphic were made using the Spray Can, as was the foliage.. Everything in this graphic with the

exception of the girl was made with Joypaint 99.. The girl was borrowed from another graphics program. If you wanted for example to make a brick wall.. You would first make the wall with the boxmaker, and then use the fill using the brick texture and you would have it.. Simple as that.

When I give examples of what can be done, I am talking about both Joypaint 99 and the Joypaint Pal, since these two work hand in glove to perform many of these unheard of functions... Now to go on.. There are eight different brush shapes which can be used in your drawing .. This also gives you great facility in erasing small or large areas.. We also have a directory feature which allows you to see the files on any disk without leaving the graphics program... You can also draw on an area larger than the screen you are viewing.. There is a function for moving the whole screen sideways and up and down. There is also a feature which allows you to magnify or reduce any area chosen of your picture.. Also, when you save a graphic it doesn't use much disk space because it only saves the picture information and not the blank areas. It will also do all the things the others will do such as Invert, Rotate, Flip Vertically or Horizontally, Copy, Move, Store on a clipboard, Circles, Ovals, Boxes, Etc.

As a few afterthoughts, we have printout in two sizes and two densities that works like a dream, and a delightful thing called Fatpixels, which is a zoom option which allows single pixel editing. The list price on these two disks is \$39.95 for Joypaint 99 and \$9.95 for the Joypaint pal.. The two disks together come to just under fifty bucks.. Worth it? .. A bargain as far as I'm concerned.. Right now it is necessary to have a Gemini 10x,15x,T.I. Impact, or any Epson compatible.. If you also have the Pal it will work with the GP100 by Axion.. Other printers are being worked on at the present time... Use of Joypaint 99 requires the following equipment.. T.I. 99/4A, Joystick, Extended Basic, E/A, or Mini-Memory.



Steinberg

THE LA99er USERS GROUP LIBRARY

Is changing the cost of obtaining programs from the LIBRARY as follow

1. Programs numbered from 1 to 1999 are \$1.00 each (disk are NOT included) Disk are \$1.00 each
2. Program Disks numbered from 2000 to 4999 are \$3.00 each or 4 for \$10.00. (disk included)
3. Mailing cost = 2 disk for \$1.00
4. Two copies if all program disks will be made available to the members at the regular meetings. This will help those who attend the meeting get ready access to the LIBRARY.

NEW PROGRAM DISKS FOR JULY

4141 **THE BIG TEXAS SPY ADVENTURE** Freeware by Sydney Micheal: This Adventure game has 158 locations! Can you find the submarine - and the Tiger moth airplane. Requires X/B,32K, (SSSD)195

4142 **MXT BBS** Freeware by Stewaet Olson and Harry Owen: An Assembly Language program, loads with X/B. This bulletin board will run on one DSSD disk but would greatly limited the board up and down loading. Has complete remote system operation and allows multi transfer without an attendance. An excellent BBS when two disks are used. 2(DSSD)971

2656 **GAMES #56 "JEUX"** 7 new games from France. Better know your French to play some of these games. Good action, color and graphic. AIRPORT, BILLIARDS, BULDOZER, ENVAHISS, FERNANDO, JUMPER, RANDONNEE. (SSSD)297

2657 **GAMES #57** 12 Games in Basic and X/B ! 32KCHECKERS, DUNGEON, ENCOUNTERS, ESCAPE, GOLDRUSH, GOLD#2, HOUSE, LUNER LANDING#2, MINEFIELD, SILENT, SILENTNIGHT, SUBMARINE, TERMITE. (SSSD)353

2429 **EDUCATION #29** From Amnion 30 "ENGINEERING" type programs :
MOHR'S CIRCLE, FFT 256 FFT POINT, ANTENNA,
DATA FORECASTING DIVERGENCE, TEMPERATURE CONVERSION, BLACK BOX,
CIRCUIT, STATS, ELECTRONICS, 3-D PLOT, BORE,
CROSS CORRELATION FOR CURVE, HARMONIC CURVE FITTING,
ALTERNATOR AND TRANSFORMERS, MATRIX(99/4), CURVE FIT, AC VOLT,
AMP CIRCUIT DESIGN, POWER, REACT, ELECTRONICS SOLVER,
ATOMIC WEIGHT CALCULATOR, STATISTICAL ANALYSIS, MOVING TIME DATA,
TRIGOMETRIC CALCULATOR, NEWTONIAN INTERPOLATOR, ANTENNA ALIGNER.
(SSSD)258

2430 **EDUCATION #30** From Amnion 27 "ENGINEERING" type program. :
555 TIMER DESIGN, DECIBEL CALCULATOR, GEOMETRY CALCULATOR,
STATISTICAL ANALYSIS, OIL WELL CASING DESIGN,
RESISTIVE ATTENUATOR DESIGN, FACTORIAL II, HYPNOTIC INDUCTION,
WIND CHILL FACTOR, CLOSED LOOP SYSTEM ANALYSIS, PRIME NUMBER CRUNCHER,
STATISTICS TABLES, COMET CALCULATOR, PLANET WATCH, FIBONACCI'S NUMBERS,
PIPING SYSTEMS ANALYSIS, TELESCOPE SPECS, UNIVERSAL TIME,
POLYNOMIAL CURVE, CONVERSION II, MATRIX DETERMINANT EVALUATOR,
VECTOR ADDITION/SUBSTRATION, PARABOLIC REFLECTOR, AIRCRAFT DESIGN,
MINI MUF, MOON WATCH, NCS OP-AID (SSSD)356

2514 **DIAL A WORD** Improved by Chick DeMarti and Tom Freeman input your 7 digit telephone number and the computer will print the many words possible combinations. (SSSD)26

NEW ADDS THANKS TO "STEVE MEHR"

- 4144 **TRACKER** Freeware by Will McGovern 215 Grinsell St. Kotara, NSW 2299 Australia : Track copier for MYARC Disk Controller. Track SSSD disk only. Use E/A Run Program File. (SSSD)31
- 4145 **FUNPLUS 4.1** Freeware by Jack Sughrue Box 459 East Douglas, MA 01515 : A disk of templates, documentation, articles, codes, programs, and a great companion to FUNLWRITER and DM1000. 2(SSSD)715
- 4146 **TOURIST/VCR** Freeware by Bill Knecht 915 Yorkshire Pasadena, TX 77503 : **Tourist Information Services** is a program that gives the name, address and phone number of who to contact in any state including Washington D.C. to obtain info and brochures. Great for vacation planning. **VCR Movies Guide** ver2.0 aids in keeping of movies on video cassettes, prints title, rating, number, counter for up to 175 movies. (SSSD)197
- 4147 **X/B TOOLS** Freeware by Jim Swedlow 7901 Kirby Way Stanton, CA 90680 : The XB/Tools programs take a program that has been saved on disk in merge format and either extract information or modify it. REFERENCE, COMPRESS, NAMECHANGE, LINEMOVE, DATAPRESS, PRINTER, ASHART, INSTALL LOAD. (SSSD)335
- 4148 **UTILITY #6** Freeware by Karl Romstedt 2543 Cranford Rd. Columbus Ohio 43221-1105 Disk full of Misc. programs : ADDRESS MACHINE create maintain address list, prints, labels, search, edit, auto phone dial. ALPHASPEAK teach your child the alphabet, speech & graphic. COMPRESSOR automaticall compress program. CONTRABAND outer space game. CONVERT convert numbers from binary, hexadecimal, or decimal. COSMIC DUEL a space duel game. DEBT CALCULATOR a spreadsheet interest rate & payments. DRILL create load edit disk file for queations and answers. FREEZER a disk base inventory of your freezer. GRADEBOOK create save esit class records. GUMMED LABEL MAKER make 3*5 X 15/16 labels each line can have a different fonts Epson type. HOUSE NUMBER a number game for children speech and graph. LABORATORY CALCULATOR biochemical % changes, gravitational forces, radioactive, ultrafiltration, etc. LOAD 2 load programs. NESTING ERROR ANALYSIS finds the unfinish loop. PANORAMA 2 create high resolution graphic merge. PRINTER INITIALIZATION select font combination for variety printers. SPRITE DEFINITION IV define sprites & merge up to 30 sprites. SUPER JOT guessing game of 3 letters words 2(SSSD)710
- 4149 **SEGREGATION** Freeware by Chris Lang 1906 Jackson Rd. Baltimore, MD 21222 : Fascinating game of putting back a mixed up crazy quited design pattern into it original condition. Great color and graphic. Good for all ages. (SSSD)238
- 4150 **FLIP-FLAP** Freeware by Chris Lang 1906 Jackson Rd. Baltimore, MD 21222 : This program prints disk sleeves and displays a disk catalog on front of the envelope. (SSSD)350
- 4151 **GENEOLOGY III** version 2.1 Freeware by Walter Davies 17718 Orchard Lane Salinas, CA 93907 : Keep track of your ancestrals for 4 generation. Print a family group sheet, locate files with ease. (SSSD)204

4152 **GAMES OF WIT** Freeware by Chris Lang 1906 Jackson Rd. Baltimore, MD 21222 : 5 Excellent educational games of wit. NIT, NITWIT, WIT-OUT, WITTLE TAGS, WIT'S END These games are like crossword puzzle, scramble, teaches spelling and word forming. For all ages. play=(SSSD)290 instruction=2(SSSD)518

4153 **MUSIC PREPROCESSOR V1.2** Freeware by Norm Sellers 15 Dorset Drive Broomall, PA 19008 : Allows you to simply and quickly code up music. You may code, run, change, rerun and play the song countless different ways (change keys,coard, minors, etc. has several songs prerecorded. 2(SSSD)616

4154 **THE USABLE DISK CATALOGUER V2.0** Freeware by Weatern New York User Group, 2753 Main St. Newfane, NY 14108 : A data base disks cataloger sorting out such file names as LOAD, A, FILES2 etc. programs can be sorted alphabetically by your key word (Games, Music, etc.), file name, or disks name. Number of sectors, kind of program access (D/V,I/F,PRM ETC.), access (E/A,X/B,TIW,etc.), and COMMENTS (your own words). (SSSD)139

4155 **EDUCATIONAL #B** Freeware by Marty Knoll Jr. 218 Kaplan Ave. Pittsburgh, PA 15227 : 3 Trailware programs 1. COOTERBUG A non-violent game for young children in Basic. 2. GEOMETRIC SHAPES in X/B illustrates many geometric shapes, labels the dimensions, displays formulars, and calculates volume, perimeter, area, or circumference [call files (1)]. great for Jr.high student. 3. MORSE CODE TUTOR X/B both begineers and advanced Morse Code users. Learn and improve code recognition and speed. Can be used to transmission over the air. (SSSD)213

4156 **LOTTO** Freeware by Phil Barnalyzer : A Lotto analyzer to help you keep track of your Lotto numbers. Keep track and plots graphs on winning numbers, bets, games etc. Prints results (SSSD)90

4157 **TIMP PRINT COMMAND V1.5** Freeware by J&B Mathis 5941 E 26th Tuscon, AZ 85711 : This file will allow you to place printer commands into any BLANK cell on your multiplan spreadsheet. You can turn off or on different commands at any point. Covers 12 of the proper printers. A must for anyone useing TI Multiplan (SSSD)176

4158 **FILEMASTER** Freeware : Create your own NAME list reports, address, city, state, zip, phone. program will Create, Change, Display, purge, save, load, print, sort, and catalog. (SSSD)49

4159 **MONTE CARLO** Freeware by Bob Guellinity 15 Wyda Court Colonia, NJ 07067 : Play Roulette as in a Monte Carlo Casino. Has great color, graphic and play. Instruction are in english betting in French (SSSD)247

4160 **THE WORKS** Freeware by Howard Uman 3913 Sybil Rd. Randallstown, MD 21133 : A Unitily disk of 8 popular programs TI-WRITER, DM-1000, FAST TERM, EDITOR ASSEMBLY, LABEL MAKER, DISK CATALOG, ARCHIVER, X-LATE. Simalar to FUNLWRITER. 2(SSSD)653

4161 SUPER SAVE Freeware by Erik Olson 6305 Rabbit Ears Circle, Colorado Springs, CO 80919-2125 : Program is used to make program files from assembly object code. Do not need the source code to save no changes are needed to the source code program. Use E/A cartridge's load & run. Disk also includes some good freebie games (SSSD)360

4162 TOTAL PROSPECT LIST Freeware by Leo DuBry 325 So Center St. Longview, TX 75601 : A label printing program for a prospect list. List is stored on disk under a file name. Labels can search and printed by number, name, city, state, zip, or phone. (SSSD)100

4163 SUPER BETTER BANNERS V2.7 Freeware (donorware) by B Falkin & J Asenas 326 Glen Way Fillmore, CA 93015 : A banner program that can work with all TI compatible printer. Fonts are loaded from disk so you can creat your own character sets. You can also create your own graphic 8X8. 16 predefined graphic are also on disk. 2(SSSD)372

4164 INSTANCES Freeware by Howard Uman 3913 Sybil Rd. Randallstown, MD 21133 : 57 Artist Instances APPLE BASEBALL BIPLANE BIRD BRUSH CANNON CAR CASSETTE CHAMPAGN CLOCK COKE COMB COPTER CROSS DISK DRUM FLAG FORK FROG GLOBE HELMET HORSE HOUSE INGRAM IRON JET KNIFE LIGHT MINI99 NOTEPAD KNIFE PAW PIN RIBBON SAILBOAT SHIP SHUTTLE SIGNAL SMILE SNOOPY SPOON STAR STERO STOP SWITCH TANK TAPE TARGET TELESCOPE TENNIS TRUCK TRUMPET TV UFO VISA (SSSD)345

4117 DISK UTILITIES V3.3 Freeware by John Birdwell 7052 Spring Hill Circle Eden Prarie, MN 55340 : A DSSD disk with the following menu. COMPARE DISKS, PRINTS SECTORS, SECTOR EDITOR, FIND STRING, DISK REPORT, DIRECT COMMENTS, FILE UTILITIES, PRINTS SETUP, SCREEN COLOR. Provides the user with a means to study how data is stored on a disk. Good for changing and editing the data to suit their own purpose. Auto-Dialer II for Fast term also has been added. (DDSD)665

4165 ADDRESS & PHONE FILE Freeware by Walt Davies 17718 Orchard Ln. Salinas, CA 93907 : Keeps your addresses and phone numbers in a file on a disk. Can sort by Name, search by name, address, city, state, zip, or telephone. Prints labels or envelope (large or small) with both TO & FROM addresses. (SSSD)164.

LIBRARIAN

FRED MOORE 7730 EMERSON AVE LOS ANGELES, CA 90045 213-670 4293

REMEMBER NEXT MEETING - Wednesday July 22, Torrance Public Library, 7 PM

***** CLUB OFFICERS *****

- | | | |
|----------------------------------|---------------------------------|----|
| ** | * | ** |
| ** President: | * Membership Chairman: | ** |
| ** Tom Freeman (213) 454-1943 | * Ed May (213) 644-6241 | ** |
| ** Vice President: | * Librarian: | ** |
| ** Terrie Masters (213) 271-6930 | * Fred Moore (213) 670-4293 | ** |
| ** Secretary: | * Library Assistants: | ** |
| ** Doug Moore (213) 451-1069 | * Alan Whiteman (213) 379-8031 | ** |
| ** Treasurer: | * Chick De Marti (213) 532-8499 | ** |
| ** Alan Whiteman (213) 379-8031 | * Equipment Chairman: | ** |
| ** Topics Editors: | * Joe Fierstein (213) 377-9834 | ** |
| ** Terrie Masters (213) 271-6930 | * Hospitality Chairman: | ** |
| ** Tom Freeman (213) 454-1943 | * Myron Harms (213) 675-3959 | ** |
| ** | * | ** |

Membership in the LA 99ers, including subscription to Topics is \$20.00 per year

MARKETPLACE

=====

(the marketplace is a fund raiser for the club, that is, the "profit" goes to maintain the quality of this Newsletter. In general the price listed splits the difference between cost and retail. Please help your club.)

MILLERS GRAPHICS	
DISASSEMBLER	10.00
ORPHAN CHRONICLES (PRICELESS)	9.95
ADVANCED DIAGNOSTICS	10.00
NIGHT MISSION	10.00
SGK UTILITY I	10.00
SMART PROGRAMMING FOR SPRITES	6.25
NEW RELEASES	
KRACKER FACTS	5.00
JOYPAINT	7.00
JOYPAINT PAL	7.00
UTILITIES DISK/DOCS (T. FREEMAN)	8.00
PRE-SCAN IT! (J. PETER HODDIE)	10.00
GRAM PACKER	10.00
FONT WRITER	19.00
PRINTER'S APPRENTICE (M. McCANN)	19.00
MYARC	
RS232	Check
D/D DISK CONTROLLER	for
128K RAM DISK/SPOOLER	discount
512K RAM DISK/SPOOLER	prices
EXTENDED BASIC II LEVEL IV	
128K RAM DISK W/XBASIC II	
512K RAM DISK W/XBASIC II	
GENEVE 9640 COMPUTER	
INSCEBOT	
TI-ARTIST	15.00
DISPLAY MASTER	12.00
ARTIST EXTRAS	6.00
GENIAL COMPUTERWARE	
XBASHER (MIKE DODD)	9.00
XB:BUG (J. PETER HODDIE)	12.00
MEGATRONICS	
EXTENDED BASIC II PLUS	72.50
INTERN (BOOK ON GPL)	16.50
128K GRAM CARD	227.50
HARDWARE & SUPPLIES	
TEAC USBV DSDD DRIVES	90.00
DISKETTES DSDD	.50
BACK ISSUES	
SUPER 99 MONTHLY	1.25
MICROPENDIUM	1.25
SMART PROGRAMMER JUNE 1986	1.00
BEST OF NEWSLETTERS W/DISK	10.00
FORTH NOTES VOL 1-6 (2.50 EA)	10.00
BEGINNER'S FORTH NOTEBOOK	2.00
ASSEMBLY NOTES VOL 1	2.00
TECHNICAL AND BUSINESS BOOKS	7.00
SAMS BOOKS (VARIOUS)	7.00
SAMS BOOKS WITH CASSETTES	7.00

(please send your order to the CLUB address, not the Librarian, and add \$1.00 per disk for postage and handling. CA residents add 6.5% tax).