

## PRESIDENT'S NOTES...by Jay Seaberg

This past month has been busy with lots of information coming in. The latest batch of newsletters from across the country reveal several hardware modifications that can be made to your computer.

One article details the modifications you can make to the disk controller card to allow you to run DS/DD drives. Another article explains how you can mount the speech synthesizer inside the console, while another tells how to add memory to your computer.

There are plans for a home built surge suppressor, including wiring diagrams and parts list. One intrepid user out there published a method that allows the combination of the E/A and TI-Writer modules in one housing.

There is a treasure chest of information in every month's collection of newsletters. There are software listings in Basic, X-Basic, Assembly, Forth, and even in 'C'. We have begun a system to index our newsletter library. Once this is in place, it should be quite easy to find articles on whatever topics may interest you. You can access the library at the meetings or on request.

Harry Livergood will be demonstrating TI-Artist at the next meeting. This software allows your artistic tendencies full rein, and also sports a number of different fonts for fancy printing. He will also be demonstrating his new printer, a Star SG-10.

Speaking of printers, there is a plug-in chip available on the market that can be added to the Gemini 10-X that will give you NLQ printing. The source and other pertinent info will be available at the meeting or you can call me at home.

If you have an interest in Forth, we have recently started a SIG to learn a little more about this language. If you need more information, call Jesse Jolly or myself.

## INSIDE...

TIPS from the TIGERCUB by Jim Peterson

CARDTRICK from the PUG Pittsburgh U.G.

ASSEMBLY FASTLOAD from the PUG Pittsburgh U.G.

DISK LIBRARY from the Edmonton 99'er

NEW YORK from the Milwaukee Area 99/4 U.G.

INSANITY from the San Fransisco 99ers Newsletter

By john willforth

This is one of my early attempts to take a program written in Microsoft Basic and convert it to TI Basic. The trick is really not a trick of course, but the effect appears as if the computer can really read your mind. If the program is a little too long for you, just start at line 400, skipping the title and instructions. Good luck and have fun. OH BY THE WAY, there is a bug that I should warn you about, and that is occasionally the computer will randomly select TWO JACK OF DIAMONDS, in that event dont select a JACK OF DIAMONDS, as neither you or the computer will know which is correct. Now I never went back to fix that bug. I beleive that I saved it for YOU! I will offer a reward for the one who gives me the first FIX.

```

100 REM THIS PROGRAM WAS ALTERED IN ORDER TO RUN ON THE
      TI-99/A COMPUTER
110 REM BY JOHN WILLFORTH
120 REM HOPE YOU ENJOY IT.
130 REM 6/02/83
140 CALL CLEAR
150 CALL SCREEN(2)
160 FOR Q=1 TO 10
170 PRINT TAB(11);"COLUMN"
180 NEXT Q
190 PRINT
200 PRINT TAB(5);"CREATIVE COMPUTING";
210 PRINT TAB(3);"MORRISTOWN, NEW JERSEY"
220 PRINT :TAB(5);"JOHN F. WILLFORTH"
230 FOR Q=1 TO 8
240 PRINT TAB(11);"COLUMN"
250 NEXT Q
260 CALL SCREEN(12)
270 FOR DELAY=1 TO 1000
280 NEXT DELAY
290 FOR M=1 TO 30
300 PRINT TAB(11);"COLUMN"
310 NEXT M
320 CALL SCREEN(2)
330 CALL CLEAR
340 PRINT "THIS PROGRAM WILL SHOW YOU ACARD TRICK. AFTER THE FIRST DEAL, PICK A
CARD AND TYPE THE NUMBER OF THE THE COLUMNCONTAINING IT."
350 REM
360 PRINT "THE DEALER WILL THEN PICK UP THE CARDS, A COLUMN AT A TIME, AND WIL
L DEAL THEM OUTAGAIN HORIZONTALLY. WHEN HE"
370 REM
380 PRINT "FINISHES EACH TIME, TYPE THENUMBER OF THE NEW COLUMN CONTAINING YO
UR CARD.FOLLOW-ING THE LAST DEAL THE DEALER"
390 PRINT "WILL TURN OVER THE CARDS ONEAT A TIME, UNTIL HE REACHES THE ONE CAR
D THAT..... YOU PICKED!!!!"
400 CALL SCREEN(10)
410 FOR DELAY=1 TO 4500
420 NEXT DELAY
430 CALL COLOR(9,2,10)
440 CALL COLOR(10,7,10)
450 CALL CLEAR
460 RANDOMIZE
470 PRINT TAB(2);"ONE";TAB(10);"TWO";TAB(19);"THREE":;
480 DIM A(21),B(21)
490 FOR I=1 TO 21
500 J=0
510 T=INT(528RND)+1
520 REM
530 FOR Y=1 TO I-1
540 IF A(Y)=T THEN 510
550 NEXT Y
560 A(I)=T
570 NEXT I
580 M=0
590 FOR I=1 TO 3
600 FOR Z=1 TO 21
610 IF A(Z)=4*(INT(A(Z)/4))THEN 730
620 IF A(Z)-2=4*(INT(A(Z)/4))THEN 700
630 IF A(Z)-3=4*(INT(A(Z)/4))THEN 670
640 CALL CHAR(100,"1818DBFFF7E7E3C18")
650 P=100
660 GOTO 750
670 CALL CHAR(104,"56FFFF7E7E3C1818")
680 P=104
690 GOTO 750

```

```

860 GOTO 890
870 PRINT
880 GOTO 890
890 IF J=5 THEN 1370
900 IF J=10 THEN 1510
910 GOTO 1100
920 IF INT(A(Z)/4)=9 THEN 1010
930 IF INT(A(Z)/4)=10 THEN 990
940 IF INT(A(Z)/4)=11 THEN 970
950 AS=" JK"
960 GOTO 1020
970 AS=" QM"
980 GOTO 1020
990 AS=" KG"
1000 GOTO 1020
1010 AS=" AC"
1020 PRINT TAB((M-1)*9);AS;" ";CHR$(P);
1030 CALL SCREEN(10)
1040 IF M=3 THEN 1060
1050 GOTO 1080
1060 PRINT
1070 GOTO 1080
1080 IF J=5 THEN 1370
1090 IF J=10 THEN 1510
1100 NEXT Z
1110 PRINT :;
1120 PRINT "WHICH COLUMN CONTAINS YOUR CARD";
1130 INPUT K
1140 CALL CLEAR
1150 CALL SCREEN(10)
1160 IF K<1 THEN 1190
1170 IF K>3 THEN 1190
1180 GOTO 1210
1190 PRINT " (1-3)"
1200 GOTO 1120
1210 PRINT :;
1220 T=1
1230 S=K+2-3*INT((K+1)/3)
1240 GOSUB 1460
1250 S=K
1260 GOSUB 1460
1270 S=K+1-3*INT(K/3)
1280 GOSUB 1460
1290 FOR C=1 TO 21
1300 A(C)=B(C)
1310 NEXT C
1320 NEXT I
1330 J=5
1340 FOR Z=1 TO 11+INT(108RND)+1
1350 M=0
1360 GOTO 610
1370 PRINT :
1380 NEXT Z
1390 PRINT
1400 PRINT " OOPS !!! YOUR CARD IS THE
";
1410 PRINT
1420 M=1
1430 J=10
1440 Z=11
1450 GOTO 610
1460 FOR R=S TO S+18 STEP 3
1470 B(T)=A(R)
1480 T=T+1
1490 NEXT R
1500 RETURN
1510 PRINT
1520 PRINT "DO YOU WANT TO SEE IT AGAIN? TYPE <YES> AND PRESS ENTER";
1530 INPUT T$
1540 IF T$="YES" THEN 450
1550 END

```

NEW YORK

100 A=40 :: DIM B(40),C(40),  
D(40),E(37),F(37),A\$(37),S(3  
,H(C),I(9),J(9)

110 RESTORE :: CALL CLEAR ::  
CALL CHARSET :: CALL SCREEN  
(15) :: K=100 :: FOR L=0 TO 4  
0 :: D(L)=0 :: NEXT L :: M=0  
:: N=0 :: O=0 :: FOR L=32 T  
O 44 :: READ B\$ :: CALL CHAR  
(L,B\$) :: NEXT L

120 DATA "55AA55AA55AA55AA",  
"55AA55AA55AA55AA", "55AA5500  
FF191919", "55AA55A4757A454A4"  
,"191919FF00AA55AA"

130 DATA "1919191919191919",  
"000000FFFF", "191919FFFF1919  
19", "FFFFFFFFFFFFFFFF"

140 DATA "0F0F0F0F0F0F0F0F",  
"0F0F0F0F0F0F0F0F", "00000000  
FFFFFFFF", "FFFFFFFF"

150 FOR L=48 TO 95 :: CALL C  
HARPAT(L,B\$) :: CALL CHAR(48+  
L,B\$) :: NEXT L :: CALL COLOR  
(1,16,2) :: CALL COLOR(2,6,2)  
:: FOR L=3 TO 9 :: CALL GOLD  
R(L,2,7) :: CALL COLOR(5+L,2,  
4) :: NEXT L

160 CALL HCHAR(5,1,39,32) ::  
CALL HCHAR(13,5,39,24) :: CAL  
L HCHAR(20,1,39,32)

170 CALL VCHAR(1,5,37,24) ::  
CALL VCHAR(5,13,37,9) :: CALL  
VCHAR(5,20,37,15) :: CALL VC  
HAR(1,29,37,24) :: FOR L=1 TO  
17 :: READ P,Q,R :: CALL HC  
HAR(Q,P,R) :: NEXT L

180 DATA 13,4,34,20,4,34,6,6  
,39,13,5,39,20,5,39,29,5,39

190 DATA 4,13,35,5,13,39,13,  
13,39,20,13,39,29,13,39,29,1  
3,33,13,14,36

200 DATA 5,20,39,20,20,39,29  
,20,39,20,21,36

210 FOR L=0 TO 37 :: READ E(I  
L),F(L),A\$(L) :: CALL HCHAR(F  
(L),E(L),ASC(A\$(L))) :: NEXT  
L

220 DATA 4,J,A,27,3,9,7,4,C,  
15,4,D,22,4,E,30,4,F,3,5,5,1  
1,3,H,18,3,1,25,3,J

230 DATA 6,7,K,14,7,L,21,7,M  
,29,7,N,4,11,0,12,11,P,19,11  
,0,27,11,R

240 DATA 7,12,6,15,12,T,22,1  
2,U,11,14,V,12,14,W,25,14,X,  
5,15,Y,21,15,3,29,15,0

250 DATA 4,19,1,19,19,2,27,1  
9,3,7,19,4,25,19,5,30,19,5,3  
,21,7,19,21,9,26,21,9,6,22,4  
,29,22,3

260 B\$="NEW YORK" :: DISPLAY  
AT(2,6):B\$ :: FOR L=1 TO 9  
:: CALL HCHAR(2,17+L,ASC(SEE  
\$(B\$,L,1))+48) :: NEXT L :: D  
ISPLAY AT(16,7):"YOU HAVE"  
: DISPLAY AT(17,6):"100 POIN  
TS" :: RANDOMIZE :: DEF B(P)  
=INT(RND\*P)

270 G(0),H(2)=1 :: G(1),H(3)  
=-1

280 FOR L=0 TO A :: T=D(L) ::  
GOSUB 820 :: IF T=0 THEN 37  
0

290 P=9(L) :: Q=C(L) :: IF T A  
ND 1 THEN P=P+1 :: U=43 :: V  
=44 :: W=39 :: X=0 :: Y=1 ::  
IF P>32 THEN 400

300 IF T AND 2 THEN P=P-1 ::  
U=44 :: V=45 :: W=38 :: X=0  
:: Y=-1 :: IF P<1 THEN 400

310 IF T AND 4 THEN Q=Q+1 ::  
U=42 :: V=41 :: W=37 :: X=-  
1 :: Y=0 :: IF Q>24 THEN 400

320 IF T AND 8 THEN Q=Q-1 ::  
U=41 :: V=42 :: W=37 :: X=1  
:: Y=0 :: IF Q<1 THEN 400

330 IF T AND 16 THEN 390

340 IF T AND 32 THEN 480

350 IF T AND 64 THEN W=39 ::  
T=T-64 :: D(L)=T :: K=K+10

360 GOTO 410

370 NEXT L :: GOTO 570

380 CALL GCHAR(C(L)+Y,B(L)+X  
,I) :: IF I>95 THEN D(L)=(T O  
R 32)AND NOT 16 :: GOTO 410

390 K=K-1 :: GOTO 370

400 K=K+25 :: D(L)=0 :: GOTO  
470

410 CALL GCHAR(Q,P,C) :: IF C  
=U OR C=40 THEN K=K-1 :: GDT  
O 370

420 CALL GCHAR(Q+Y,P+X,C) ::  
IF C<>32 THEN D(L)=T OR 16

430 IF C=V THEN U=40

440 CALL GCHAR(C(L),B(L),C) ::  
IF C=40 THEN W=V

450 IF C=32 THEN D(L)=0 :: K  
=K-100 :: GOTO 370

460 CALL HCHAR(Q,P,U)

470 CALL HCHAR(C(L),B(L),W) ::  
B(L)=P :: C(L)=Q :: GOTO 3  
70

480 CALL GCHAR(Q,P,C) :: IF C  
=U THEN 370

490 IF C<>39 THEN 550

500 C=0 :: FOR I=0 TO 3 :: B  
=2\*I :: CALL GCHAR(Q+H(I),P+  
G(I),C) :: IF NOT C=39 OR C=3  
7) THEN C=C OR 9

510 NEXT I

520 =2\*S(4) :: IF ( AND C)=  
0 THEN D(L)= OR 64 :: GOTO  
460

530 IF C=15 THEN 370 ELSE 52  
0

540 D(L)=0 :: IF C<>32 THEN  
I(I)=F :: J(I)=0 :: D=0-1 ::  
K=K-100

550 U=32 :: CALL SOUND(4000,  
-2,0) :: GOTO 480

570 IF D(N)=0 THEN 600

580 N=N+1 :: IF N>A THEN N=0  
ELSE 700

600 B=S(4) :: D(N)=2\*B :: DN  
B+1 GOTO 510,520,530,540

610 B(N)=1 :: GOTO 650

620 B(N)=32 :: GOTO 650

630 C(N)=1 :: GOTO 670

640 C(N)=24 :: GOTO 670

650 IF S(2)=1 THEN C(N)=5 ::  
GOTO 700

660 C(N)=20 :: GOTO 700

670 IF S(2)=1 THEN B(N)=5 ::

GOTO 700

680 B(N)=29

700 IF K<0 THEN K=0

710 DISPLAY AT(17,5):USING "  
####":K :: IF K=0 THEN 760

720 M=M+1 :: IF M>100 THEN 7  
50

730 IF S(25)=0 OR D=0 THEN 7  
50

740 CALL HCHAR(J(I),I(I),39)  
:: FOR L=0 TO 9 :: J(L)=J(L)  
+1 :: I(L)=I(L)+1 :: NEXT L  
:: O=0-1

750 GOTO 290

760 IF K=0 THEN CALL SOUND(4  
000,-1,0) :: GOTO 780

770 FOR L=1 TO 5 :: CALL SOUN  
D(1000,599,0) :: CALL SOUND(1  
000,784,0) :: CALL SOUND(100  
0,890,0) :: NEXT L

780 DISPLAY AT(2,6):"PLAY AG  
AIN ? Y N Y " :: ACCEPT AT(2  
,23)SIZE(1)VALIDATE("YN"):C  
\$ :: IF C\$="Y" THEN 110

900 CALL CLEAR :: CALL CHARS  
ET :: PRINT "YOU HAVE":K:"PO  
INTS" :: END

820 CALL KEY(0,I,R) :: IF R<1  
1 THEN RETURN

830 IF I=64 AND I<91 THEN I=  
I-65 :: GOTO 880

840 IF I=47 AND I<59 THEN I=  
I-22 :: GOTO 880

850 IF I=44 THEN I=36 :: GOT  
O 880

860 IF I=46 THEN I=37 :: GOT  
O 880

870 RETURN

880 CALL GCHAR(F(I),E(I),C) ::  
IF C>95 THEN C=C-48 ELSE I  
=I+48

900 CALL HCHAR(F(I),E(I),C) ::  
CALL SOUND(-200,1047,0) ::  
RETURN



HOW TO CONVERT ASSEMBLY PROGRAMS TO PROGRAM FORM FOR FASTER LOADING AND LESS DISK SPACE.

Written by Darren Leonard PUG on an idea by Marty Kroll Jr.

*Convert to hex and add A+B*

If you have ever loaded an assembly program with editor/assembler option #3 you may have noticed that it takes quite a while to load. With some programs this can take over 2 minutes. These types of programs are in Display/Fixed 80 format which we are going to change to PROGRAM format to load with OPTION #5. In addition to loading 3 to 5 times faster, programs stored in program format, i.e. Memory Image, take as little as 1/4 the disk space of D/F 80 files.

The method outlined in this article will work on 95% of all Assembly D/F 80 programs. Prior to writing this, I tried it on 20 programs and it worked on 19 of them. It will even allow you to save a ASSEMBLY program to cassette. Thus people with and E/A and 32K can run assembly programs!

To begin with read page 420 of the Editor/Assembler manual. Try your program the way they outline it. If you get an error then read on and I will explain in detail how to get around it.

This section describes the procedure for D/F 80 files that DO NOT AUTOSTART!, if your program does autostart read down a few paragraphs on how to remove it with DISKO.

- 1) Plug in your E/A and call up TI-BASIC, your E/A must be plugged in!
- 2) Type " CALL INIT"  
" CALL LOAD("DSK1.FILENAME")
- 3) If your program has more than one file type in all the remaining files in order as follows:  
" CALL LOAD("DSK1.GAME#1")  
" CALL LOAD("DSK1.GAME#2")  
" CALL LOAD("DSK1.GAME#3")  
get the idea?
- 4) Type "CALL PEEK(8228,A,B)"  
PRINT A,B
- 5) Now 2 numbers will appear on the screen, one on the left and one in the middle of the screen. This number corresponds to the first free address in the memory which is also the last address of your program.

number. Since your program is normally loaded in memory from addresses >A000->FFD7 if you get A000 for A+B then your program has an Absolute Origin statement (AORG) and you will not be able to convert it with this method. Similarly, if A+B is A780 or smaller then the program is loaded in a unusual manner since it cannot fit in the small area from >A000-A780. But if you come up with A+B=B000 or greater then this method will work 99% of the time.

- 7) Type " BYE " and call up the editor. Now type in the small assembly program listed here:

```
DEF SFIRST,SLAST,SLOAD
SFIRST EQU >A000
SLOAD EQU >A000
SLAST EQU >A780 (the value of A+B)
END
```

NOTE!! PUT THE HEX NUMBER OF A+B IN THE PLACE WHERE A780 IS!!!!

Hit Fctn 9 twice and save to disk.

- 8) Load the Assembler.

For source file enter what you save in step 7.

For object file type DSK1.GAME#4 or what you want.

Hit return for the printer output.

TYPE "RC" when it prompts for assembler directives.

It will then assemble the program. You shouldn't get any errors.

- 9) Now load E/A option 3.

Enter your filename DSK1.GAME#1  
DSK1.GAME#2

Then enter the assembled filename from DSK1.GAME#4 step 8.

- 10) Insert E/A disk #2 into drive one and load file " DSK1.SAVE". Hit enter and type "SAVE" for the program name. Follow the screen input prompts.
- 11) Now hit FCN + and call up E/A option #5 and type DSK1.YOURFILE and wala!

## DISK LIBRARY BY R.K. LOTEN

JUNE 20, 1985

THIS IS MY VERSION OF A SELF  
LOADING DISK PROGRAM THAT  
CATALOGUES ALL YOUR FLOPPIES  
AND EITHER DISPLAYS OR  
PRINTS ALL RELEVANT DATA.

NOV. 11, 1985 REVISION

10 ! DISK FILE LIBRARY-  
SAVED AS PROGRAM FILE:LOAD\*\*

20 ! DEFINE VARIABLES

30 OPTION BASE 1

40 DIM DSKNAME\$(20),FILENAME\$(20,20),SECTORSFREE(20),FILESECTORS(20,20),FILETYPE(20,20),RECLEN(20,20)

50 WHERE\$="OUTSIDE"

60 ! MODIFY SCREEN COLOURS

70 CALL CLEAR :: FOR A=1 TO 9 :: CALL COLOR(A-1,16,2):: NEXT A :: R,S=0 :: START=1

80 CALL VCHAR(2,4,30,4):: CALL DELAY(400):: CALL HCHAR(2,4,30,26):: CALL DELAY(400):: CALL VCHAR(2,29,30,4):: CALL DELAY(400)

90 CALL HCHAR(6,4,30,26):: CALL DELAY(400)

100 MSG\$="DISK LIBRARY MENU"  
:: CALL SLOWPRINT(MSG\$,4,"")

110 IF WHERE\$="INSIDE" AND LEN(NDATE\$)&lt;&gt;0 THEN 160

120 ! BRING UP DATE OF LATEST ADDITION/REVISION

130 OPEN #1:"DSK1.DATEFILE",RELATIVE,INTERNAL,FIXED 20

140 RESTORE #1 :: IF EOF(1)=1 THEN CLOSE #1 :: GOTO 180

150 INPUT #1,REC 0:NDATE\$,DSKNUM :: CLOSE #1

160 CALL CALENDER(NDATE\$,SDATE\$):: MSG\$="DATE OF LAST REVISION:" :: CALL SLOWPRINT(MSG\$,8,"")

170 MSG\$=SDATE\$ :: CALL SLOWPRINT(MSG\$,10,"")

180 IF WHERE\$="INSIDE" THEN 350

190 MSG\$="SPEECH SYNTHESIZER ON-LINE?" :: CALL SLOWPRINT

(MSG\$,13,"")

200 MSG\$="PRESS Y OR N" :: CALL SLOWPRINT(MSG\$,14,"")

210 CALL KEY(0,R,S):: IF S&lt;&gt;1 THEN 210

220 IF R&lt;&gt;89 AND R&lt;&gt;78 THEN 210 ELSE IF R=89 THEN SS\$="YES" ELSE SS\$="NO"

230 IF SS\$="NO" THEN 260

240 MSG\$="REMEMBER TO INCREASE THE" :: CALL SLOWPRINT(MSG\$,17,""):: MSG\$="MONITOR/T VOLUME FOR THE" :: CALL SLOWPRINT(MSG\$,18,"")

250 MSG\$="VERBAL COMMANDS TO BE HEARD" :: CALL SLOWPRINT(MSG\$,19,""):: CALL DELAY(500):: FOR BB=17 TO 19 :: DISPLAY AT(BB,1):" " :: NEXT BB

260 MSG\$="SINGLE OR TWIN DISK DRIVES?" :: CALL SLOWPRINT(MSG\$,16,"")

270 MSG\$="PRESS 1 OR 2" :: CALL SLOWPRINT(MSG\$,17,"")

280 CALL KEY(0,R,S):: IF S&lt;&gt;1 THEN 280

290 IF R&lt;49 OR R&gt;51 THEN 280

300 DRIVE\$=CHR\$(R)

310 MSG\$="PRINTER ON-LINE?" :: CALL SLOWPRINT(MSG\$,19,""):: MSG\$="PRESS Y OR N" :: CALL SLOWPRINT(MSG\$,20,"")

320 CALL KEY(0,R,S):: IF S&lt;&gt;1 THEN 320

330 IF R&lt;&gt;89 AND R&lt;&gt;78 THEN 320 ELSE IF R=89 THEN PRINTER\$="YES" ELSE PRINTER\$="NO"

340 FOR A=13 TO 20 :: DISPLAY AT(A,1):" " :: NEXT A

350 MSG\$="PRESS" :: CALL SLOWPRINT(MSG\$,13,"")

360 MSG\$="1-DISPLAY LISTINGS" :: CALL SLOWPRINT(MSG\$,15,"")

370 MSG\$="2-EDIT/ENTER LISTINGS" :: CALL SLOWPRINT(MSG\$,16,"")

380 MSG\$="3-STOP EXECUTION" :: CALL SLOWPRINT(MSG\$,17,"")

390 ! LINES 350-430 ARE THE TIMING LOOP FOR PROMPTS.

400 Q=0 :: DISPLAY AT(22,2):

"IDLE FOR 8";TAB(18);"% SECONDS"

410 K=0 :: IF SS\$="NO" THEN 430 ELSE 420

420 FOR IX=1 TO 2 :: CALL SOUND(400,500,15):: CALL SOUND(400,300,15):: NEXT IX :: CALL SAY("PLEASE PRESS THE KEY OF YOUR CHOICE"):: GOTO 450

430 DISPLAY AT(19,1):" " :: DISPLAY AT(20,1):" " :: MSG\$="PLEASE PRESS THE KEY OF" :: CALL SLOWPRINT(MSG\$,19,"")

440 MSG\$="YOUR CHOICE" :: CALL SLOWPRINT(MSG\$,20,"")

450 CALL KEY(0,R,S):: IF S=1 THEN 470

460 K=K+1 :: Q=Q+1 :: IF K&lt;500 THEN DISPLAY AT(22,12)SIZE(5):INT(Q/8):: GOTO 450 ELSE 410

470 IF R&gt;48 AND R&lt;52 THEN 480 ELSE 450

480 ON R-48 GOTO 1050,510,490

490 CALL CLEAR :: STOP

500 REM SEARCH ALL YOUR DISKS AND THE PROGRAM NAMES

510 DISPLAY AT(5,1)ERASE ALL:"INPUT TODAY'S DATE, USING THE FORMAT-YMMD" :: ACCEPT AT(8,10)SIZE(6)VALIDATE(DIGIT)BEEP:DATE\$

520 CALL DATECHECK(DATE\$,OK\$):: IF OK\$="N" THEN 510 ELSE 530

530 DISPLAY AT(10,8):"PRESS" :: DISPLAY AT(12,2):"1 FOR COMPLETE REVISION" :: DISPLAY AT(13,2):"2 TO REVISE ONE DISK ONLY"

540 DISPLAY AT(14,2):"3 TO ADD ANOTHER DISK"

550 CALL KEY(0,R,S):: IF S&lt;&gt;1 THEN 550

560 IF R&lt;49 AND R&gt;51 THEN 550

570 ON R-48 GOTO 620,580,600

580 IF SS\$="YES" THEN CALL SAY("WHAT IS DISKETTE NUMBER")

590 DISPLAY AT(18,2):"INPUT DISK # " :: ACCEPT AT(18,16)SIZE(3)VALIDATE(DIGIT):START :: IF START&lt;1 OR START&gt;DSKNU

M THEN 590 ELSE 620

600 START=DSKNUM+1

610 REM OPEN CATALOG

620 OPEN #1:"DSK"&amp;DRIVE\$&amp;".\* &amp;"" ,INPUT ,RELATIVE,INTERNAL

630 FOR X=START TO 20

640 CALL SPGET(STR\$(X),X\$)

650 ON VAL(DRIVE\$)GOTO 660,680

660 IF SS\$="YES" THEN CALL SAY("TAKE OUT PROGRAM DISKETTE"):: CALL SAY("PUT IN DISKETTE",X\$)

670 DISPLAY AT(6,1)ERASE ALL:"REMOVE LIBRARY PROGRAM DISKET" :: DISPLAY AT(8,1):"INSERT DISK";TAB(15);" " :: DISPLAY AT(8,13)SIZE(2):X :: GOT 0 700

680 IF SS\$="YES" THEN CALL SAY("PUT DISKETTE",X\$,"IN NUMBER 2 DEVICE"):: CALL DELAY(1000):: CALL SAY("THEN PRESS ANY KEY TO START")

690 DISPLAY AT(6,1)ERASE ALL:"INSERT DISK";TAB(15);" INT 0 THE 2ND" :: DISPLAY AT(7,1):"DISK DRIVE." :: DISPLAY AT(6,12)SIZE(3):X

700 DISPLAY AT(12,1):"PRESS ANY KEY TO START THE DISK LISTING PROCESS"

710 CALL KEY(0,R,S):: IF S&lt;&gt;1 THEN 710

720 INPUT #1,REC 0:DSKNAME\$(X),DUMMY1,DUMMY2,SECTORSFREE(X)

730 REM RECORD PROGRAM FILE NAMES

740 DISPLAY AT(8,1)ERASE ALL:"CHECKING DISK #";X

750 FOR Y=1 TO 20

760 INPUT #1:FILENAME\$(X,Y),FILETYPE(X,Y),FILESECTORS(X,Y),RECLEN(X,Y)

770 IF LEN(FILENAME\$(X,Y))=0 THEN 790

780 NEXT Y

790 DISPLAY AT(12,1):"DISK #";X;" LISTING DONE" :: CALL DELAY(750)



800 IF R=48=2 THEN 860

810 DISPLAY AT(15,1):"SEARCH  
ANOTHER DISK OR STORE ALL DI  
SK NAMES & PROGRAM FILENA  
MES FOUND SO FAR?"

820 DISPLAY AT(19,1):"PRESS  
1 TO CONTINUE SEARCH" :: DIS  
PLAY AT(21,1):"PRESS 2 TO ST  
ORE LISTINGS"

830 CALL KEY(O,R,S):: IF S<  
1 THEN 830

840 IF R<49 AND R>50 THEN 83  
0 ELSE IF R=50 THEN 860

850 NEXT X

860 CLOSE #1

870 IF VAL(DRIVE\$)=2 THEN 90  
0

880 DISPLAY AT(8,1)ERASE ALL  
:"INSERT LIBRARY PROGRAM DIS  
K" :: DISPLAY AT(10,1):"PRES  
S ANY KEY WHEN FINISHED"

890 CALL KEY(O,R,S):: IF S<  
1 THEN 890

900 DISPLAY AT(8,5)ERASE ALL  
:"PLEASE WAIT-IN PROCESS"

910 OPEN #1:"DSK1.DATEFILE",  
RELATIVE,INTERNAL,FIXED 20

920 PRINT #1,REC 0:DATE\$,X :  
: CLOSE #1

930 FOR DSK=START TO X

940 OPEN #1:"DSK1.DFNAME"&ST  
R\$(DSK),RELATIVE,INTERNAL,FI  
XED 60

950 PRINT #1,REC 0:DSKNAME\$(  
DSK),SECTORSFREE(DSK)

960 FOR FILE=1 TO 20

970 IF FILENAME\$(DSK,FILE)=""  
 THEN 1000

980 PRINT #1,REC FILE:FILENA  
ME\$(DSK,FILE),FILESECTORS(D  
SK,FILE),FILETYPE(DSK,FILE),R  
ECLN(DSK,FILE)

990 NEXT FILE

1000 CLOSE #1

1010 IF START>1 THEN 1030

1020 NEXT DSK

1030 WHERE\$="INSIDE" :: GOTO  
70 ! RETURN TO MENU

1040 REM ACCESS AND DISPLAY  
DATA

1050 DISPLAY AT(10,1)ERASE A  
LL:"DO YOU WANT TO REVIEW AL  
L OR JUST ONE OF THE LISTED D  
ISKS"

1060 DISPLAY AT(13,2):"PRESS  
1 FOR ALL THE DISKS" :: DIS  
PLAY AT(14,8):"2 FOR ONE SEL  
ECT DISK"

1070 CALL KEY(O,R,S):: IF S<  
>1 THEN 1070

1080 IF R<49 AND R>50 THEN 1  
070 ELSE IF R=49 THEN DNUM=1  
:: GOTO 1100

1090 DISPLAY AT(17,2):"TYPE  
DISK #" :: ACCEPT AT(17,14)S  
IZE(2)VALIDATE(DIGIT):DNUM :  
: CHOICE\$="ONE"

1100 FOR DSK=DNUM TO 20

1110 OPEN #1:"DSK1.DFNAME"&S  
TR\$(DSK),RELATIVE,INTERNAL,FI  
XED 60

1120 RESTORE #1 :: IF EOF(1)  
=1 THEN CLOSE #1 :: GOTO 171  
0

1130 INPUT #1,REC 0:DSKNAME\$(  
DSK),SECTORSFREE(DSK)

1140 FOR Z=1 TO 20 :: FILENA  
ME\$(DSK,Z)="" :: NEXT Z ! R  
ESET BUFFER

1150 FOR NAME=1 TO 20

1160 IF EOF(1)=1 THEN 1190

1170 INPUT #1,REC NAME:FILENA  
ME\$(DSK,NAME),FILESECTORS(D  
SK,NAME),FILETYPE(DSK,NAME),  
RECLN(DSK,NAME)

1180 NEXT NAME

1190 DISPLAY AT(2,6)ERASE AL  
L:"DISKNAME=";DSKNAME\$(DSK)

1200 DISPLAY AT(3,4):"DATE:  
";SDATE\$

1210 DISPLAY AT(4,4):"AVAILA  
BLE=";TAB(19);"USED=" :: DIS  
PLAY AT(4,14)SIZE(4):SECTORS  
FREE(DSK)

1220 DISPLAY AT(4,24)SIZE(4)  
:358-SECTORSFREE(DSK)

1230 DISPLAY AT(5,4):"FILENA  
ME";TAB(13);"SIZE";TAB(19);"  
TYPE" :: DISPLAY AT(6,4):RPT  
\$( "-" ,8);TAB(13);"----";TAB(  
19);"----"

1240 B=1

1250 FOR A=1 TO 20

1260 IF LEN(FILENAME\$(DSK,A)

)<2 THEN 1400

1270 IMAGE "#####  
#####"

1280 RECSIZE\$=STR\$(RECLN(D  
SK,A)):: IF RECLN(DSK,A)=0 T  
HEN RECSIZE\$=""

1290 ON FILETYPE(DSK,A)GOTO  
1300,1310,1320,1330,1340

1300 TYPE\$="DIS/FIX" :: GOTO  
1350

1310 TYPE\$="DIS/VAR" :: GOTO  
1350

1320 TYPE\$="INT/FIX" :: GOTO  
1350

1330 TYPE\$="INT/VAR" :: GOTO  
1350

1340 TYPE\$="PROGRAM"

1350 DISPLAY AT(8+7,1):USING  
1270:FILENAME\$(DSK,A),FILES  
ECTORS(DSK,A),TYPE\$,RECSIZE\$

1360 B=B+1 :: IF B>7<20 THEN  
1390 ELSE B=1

1370 DISPLAY AT(21,1):" PRES  
S ANY KEY TO CONTINUE "

1380 CALL KEY(O,R,S):: IF S<  
>1 THEN 1380

1390 NEXT A

1400 IF CHOICE\$="ONE" THEN D  
ISPLAY AT(21,1):" PRESS ANY  
KEY FOR MENU" ELSE 1420

1410 CALL KEY(O,R,S):: IF S<  
>1 THEN 1410 ELSE CLOSE #1 :  
: WHERE\$="INSIDE" :: GOTO 70

1420 FOR XX=(B+7)TO 20 :: DI  
SPLAY AT(XX,1):" " :: NEXT X  
X

1430 DISPLAY AT(21,1):"PRESS  
";TAB(8);"1 FOR REVIEW" :: D  
ISPLAY AT(22,8):"2 TO RUN PR  
OGRAM"

1440 DISPLAY AT(23,8):"3 TO  
PRINT LISTING" :: DISPLAY AT  
(24,8):"4 RETURN TO MENU"

1450 CALL KEY(O,R,S):: IF S<  
>1 THEN 1450

1460 IF R<49 OR R>52 THEN 14  
50

1470 ON R-48 GOTO 1650,1660,  
1490,1480

1480 CLOSE #1 :: WHERE\$="INS  
IDE" :: GOTO 70

1490 IF PRINTER\$="YES" THEN

OPEN #5:"PI0" ELSE 1430

1500 PRINT #5 :: PRINT #5 ::  
PRINT #5:TAB(16);"DISKNAME=  
";DSKNAME\$(DSK):: PRINT #5:T  
AB(14);"DATE:";SDATE\$

1510 PRINT #5:TAB(14);"AVAIL  
ABLE=";SECTORSFREE(DSK);TAB(  
34);"USED=";358-SECTORSFREE(  
DSK)

1520 PRINT #5:TAB(14);"FILEN  
AME";TAB(26);"SIZE";TAB(34);  
"TYPE" :: PRINT #5:TAB(14);R  
PT\$("-" ,8);TAB(26);"----";TA  
B(34);"----"

1530 FOR A=1 TO 20

1540 IF LEN(FILENAME\$(DSK,A)  
)<2 THEN 1640

1550 RECSIZE\$=STR\$(RECLN(D  
SK,A)):: IF RECLN(DSK,A)=0 T  
HEN RECSIZE\$=""

1560 ON FILETYPE(DSK,A)GOTO  
1570,1580,1590,1600,1610

1570 TYPE\$="DIS/FIX" :: GOTO  
1620

1580 TYPE\$="DIS/VAR" :: GOTO  
1620

1590 TYPE\$="INT/FIX" :: GOTO  
1620

1600 TYPE\$="INT/VAR" :: GOTO  
1620

1610 TYPE\$="PROGRAM"

1620 PRINT #5:TAB(14);FILENA  
ME\$(DSK,A);TAB(26);FILESECTO  
RS(DSK,A);TAB(34);TYPE\$;TAB(  
41);RECSIZE\$

1630 NEXT A

1640 PRINT #5:CHR\$(12):: CLO  
SE #5 :: GOTO 1420

1650 CLOSE #1 :: NEXT DSK

1660 CLOSE #1 :: DISPLAY AT(  
10,1):"INSERT DISK ##";DSKNA  
ME\$(DSK);"##" :: DISPLAY AT(  
12,1):" PRESS ANY KEY TO CON  
TINUE"

1670 CALL KEY(O,R,S):: IF S<  
>1 THEN 1670

1680 DISPLAY AT(14,1):"RUN P  
ROGRAM BY TYPING THIS:"

1690 DISPLAY AT(16,1):" RU  
N ""DSK1.XXXXX""

```

1700 STOP
-----
1710 MSG$="NO DATA AVAILABLE
DUE TO : " :: CALL SLOWPRINT
(MSG$,20,"")
-----
1720 MSG$="1-NO MORE DISKS T
O REVIEW " :: CALL SLOWPRINT
(MSG$,21,"")
-----
1730 MSG$="OR" :: CALL SLOWP
RINT(MSG$,22,"")
-----
1740 MSG$="2-NO DISKS LISTED
YET " :: CALL SLOWPRINT
(MSG$,23,"")
-----
1750 CALL DELAY(1500):: WHER
E$="INSIDE" :: GOTO 70
-----
1760 SUB SLOWPRINT(MSG$,LINE
NUM,DIRECTION$)
-----
1770 IF LEN(MSG$)>=26 AND DI
RECTION$="" THEN START=1 ::
X,B=1 :: GOTO 1810
-----
1780 IF LEN(MSG$)>=26 AND DI
RECTION$="" THEN START=LEN(
MSG$):: X=-1 :: B=LEN(MSG$):
: GOTO 1810
-----
1790 IF DIRECTION$="" THEN
START=1+INT((28-LEN(MSG$))/2)
:: X,B=1 :: GOTO 1810
-----
1800 IF DIRECTION$="<" THEN
START=INT((28+LEN(MSG$))/2):
: X=-1 :: B=LEN(MSG$)
-----
1810 FOR A=START TO (START+X
*LEN(MSG$))STEP X
-----
1820 IF A=0 OR B=0 THEN SUBE
XIT
-----
1830 DISPLAY AT(LINENUM,A)SI
ZE(1):SEG$(MSG$,B,1):: CALL
DELAY(1)
-----
1840 IF DIRECTION$=">" THEN
B=B+1 ELSE IF DIRECTION$="<"
THEN B=B-1
-----
1850 NEXT A
-----
1860 SUBEND
-----
1870 SUB DELAY(N)
-----
1880 FOR A=1 TO N :: NEXT A
-----
1890 SUBEND
-----
1900 SUB CALENDER(NDATE$,SDA
TE$)
-----
1910 DIM MONTH$(12)
-----
1920 DATA JANUARY,FEBRUARY,M
ARCH,APRIL,MAY,JUNE,JULY,AUG
UST,SEPTEMBER,OCTOBER,NOVEMB
ER,DECEMBER
-----
1930 RESTORE 1920 :: FOR A=1
TO 12 :: READ MONTH$(A):: M
EXT A
-----
1940 FOR B=1 TO 12
-----
1950 IF B=VAL(SEG$(NDATE$,3,
2))THEN 1970
-----
1960 NEXT B
-----
1970 SDATE$=MONTH$(B)&"%SE
G$(NDATE$,5,2)&"%"19"%SEG$
(NDATE$,1,2)
-----
1980 SUBEND
-----
1990 SUB DATECHECK(DATE$,OK$
)
-----
2000 DIM MTHLEN(12)
-----
2010 DATA 31,28,31,30,31,30,
31,31,30,31,30,31
-----
2020 OK$="Y" :: RESTORE 2010
:: FOR A=1 TO 12 :: READ MT
HLEN(A):: NEXT A
-----
2030 IF VAL(SEG$(DATE$,1,2))
/4=INT(VAL(SEG$(DATE$,1,2)))
THEN MTHLEN(2)=29
-----
2040 IF VAL(SEG$(DATE$,3,2))
=0 OR VAL(SEG$(DATE$,3,2))>1
2 THEN 2060
-----
2050 IF VAL(SEG$(DATE$,5,2))
=0 OR VAL(SEG$(DATE$,5,2))>M
THLEN(VAL(SEG$(DATE$,3,2)))T
HEN 2070 ELSE 2100
-----
2060 DISPLAY AT(10,1):"ERRO
R* DATE INPUT INCLUDED VALUE
OF ZERO OR 2-DIGIT * GREAT
ER 12 FOR THE MONTH * .TRY A
GAIN." :: GOTO 2080
-----
2070 DISPLAY AT(10,1):"ERRO
R* DD VALUE IS GREATER THAN
THE NUMBER OF DAYS FOR THIS
MONTH.TRY AGAIN"
-----
2080 OK$="N" :: MSG$="PRESS
ANY KEY TO CONTINUE" :: CALL
SLOWPRINT(MSG$,18,"")
-----
2090 CALL KEY(O,R,S):: IF S<
>1 THEN 2090 ELSE 2100
-----
2100 SUBEND

```

The following program may drive you crazy. It requires Extended Basic and a joystick. The object is very simple: try to align four colored blocks.

```

1 REM INSANITY BY G. MINEO
5 REM WESTMEGO. LA
15 CALL CLEAR
20 DISPLAY AT(12.1):"INSANITY"
25 DISPLAY AT(13.1):"LIKE THE
NAME SAYS. USE YOUR"
30 DISPLAY AT(17.1):"JOY STICKS
TO PUT 'EM LIKE"
35 DISPLAY AT(19.1):"YOU FOUND
'EM. GOOD LUCK!"
40 DISPALT AT(23.1):"PRESS ANY
KEY TO BEGIN"
45 CALL KEY(O,K,S):: IF S=0 THEN 13
50 CALL CHAR(42."FFFFFFFFFFFF
FFF")
55 CALL MAGNIFY(2)
60 CALL CLEAR
65 DISPLAY AT(1.12):"INSANITY"
70 CALL SPRITE(#1,42,3,96,128)
75 CALL SPRITE(#2,42,9,112,128)
CALL SPRITE(#3,42,11,80,128)
CALL SPRITE(#4,42,16,128,128)
80 CALL JOYST(1,Y,X)
85 CALL MOTION(#2,-20*X,20*Y)

```

```

100 CALL MOTION(#1,-20*X,20*Y)
105 CALL MOTION(#3,-20*X,20*Y)
110 CALL MOTION(#4,-20*X,20*Y)
115 GOTO 90
120 REM IT CAN BE DONE!
125 REM REPRINTED FROM SAN FRAN-
CISCO 99ERS NEWSLETTER

```

```

100 ! *****
110 ! * SCROLL DOWN *
120 ! *****
130 CALL INIT
140 CALL LOAD(8196.63.248)
150 CALL LOAD(16376.83.67.82.76.68.
78.48.0)
160 CALL LOAD(12288.2.224.131.224.4
,192,2,1,37,20,2,2.224,4,32.32.44)
170 CALL LOAD(12306.2.1,36.244,2.2.
3,0,4,32,32.36,4.91)
180 FOR C=9460 TO 9492 :: CALL LOAD
(C,128)::NEXT C
190 !.....
195 ! ** PROGRAM TEST **
200 CALL CLEAR
210 PRINT "* TEST *"
220 FOR UP=1 TO 20 :: PRINT :: NEXT
UP
230 FOR DOWN=1 TO 20 :: CALL LINK("
SCRLDN"):: NEXT DOWN
240 GOTO 220

```



