HARDWARE CONSTRUCTION

by Richard E. Esposito and Jessie W. Jackson

Own the "biggest" CoCo on the block with this do-it-yourself 256K upgrade.

The Fat CoCo



Although it has been feasible to upgrade your CoCo to 256K of memory for some time, the cost of the necessary chips has put the project out of reach. But now you can buy 256K RAM chips for less than $5 each, and a new SAM (synchronous address multiplexor) chip with a 256K refresh cycle for $14.95. You can perform the 256K upgrade on any eight-RAM-chip CoCo or CoCo 2 that is already configured for 64K of memory for a cost of about $100. If your CoCo isn't configured to receive 64K RAMs, refer to "64K Modification Revisited," *HOT CoCo*, June 1985, p. 40.

You do not need a disk drive or Extended Color Basic to use this upgrade. With it, you have either four banks of 64K or eight banks of 32K. You switch memory banks using PEEK commands to the following addresses:

$FFC0-$FFC7–64K Bank Select
$FFC8-$FFCF–32K Bank Select
$FFD0-$FFD7–VDG (video display generator) Select
$FFD8-$FFDF–Reserved for Future 512K Upgrade

Possible uses for the extra memory include placing different DOSes (disk operating systems) in each bank, giving you concurrent Basic; animation using 30 pages of PMODE 4; and unleashing the power of the Flex and OS-9 operating systems. We've provided some software with this article to get you started. Additional software is being developed by J & R Electronics; possible future offerings include a RAM virtual disk and support for both Flex and OS-9 users.

J & R's upgrade design is, as far as we know, compatible with the stock 64K CoCo with one exception. A problem arises if you program the SAM using the CLR command in machine language (e.g., CLR $FFC0), because a CLR command PEEKs and then POKEs 00s to memory. You can fix this by replacing all offending CLR commands with STA commands (e.g., STA $FFC0).

You do not need any software or firmware drivers to refresh the RAM, so you do not lose any speed and there is no conflict with existing CoCo software. Since the hardware is set using PEEKs and the addresses used are the same as the original SAM, no address conflict is likely to arise when you use hardware add-ons such as an 80-column card, a parallel printer interface, or a voice synthesizer.

The parts list is short. (See Table 1.) Figure 1 presents a schematic for dedicated hardware hackers who are capable of wire-wrapping and building their own project boards. Design the board layout so that it plugs into your original SAM socket. Photo 1 shows a completed unit.

For those of you less inclined to do it all yourself, J & R Electronics (P.O. Box 2572, Columbia, MD 21045) is offering the board, called The Banker, in various stages of completion as listed below:

- Bare board—$29.95
- Bare board and parts kit (no memory chips)—$59.95
- Board assembled and tested (no memory chips)—$69.95
- Board assembled and tested with memory chips—$99.95

A $3 shipping and handling fee is charged for each order ($7 for foreign orders). All versions come etched and drilled with software and

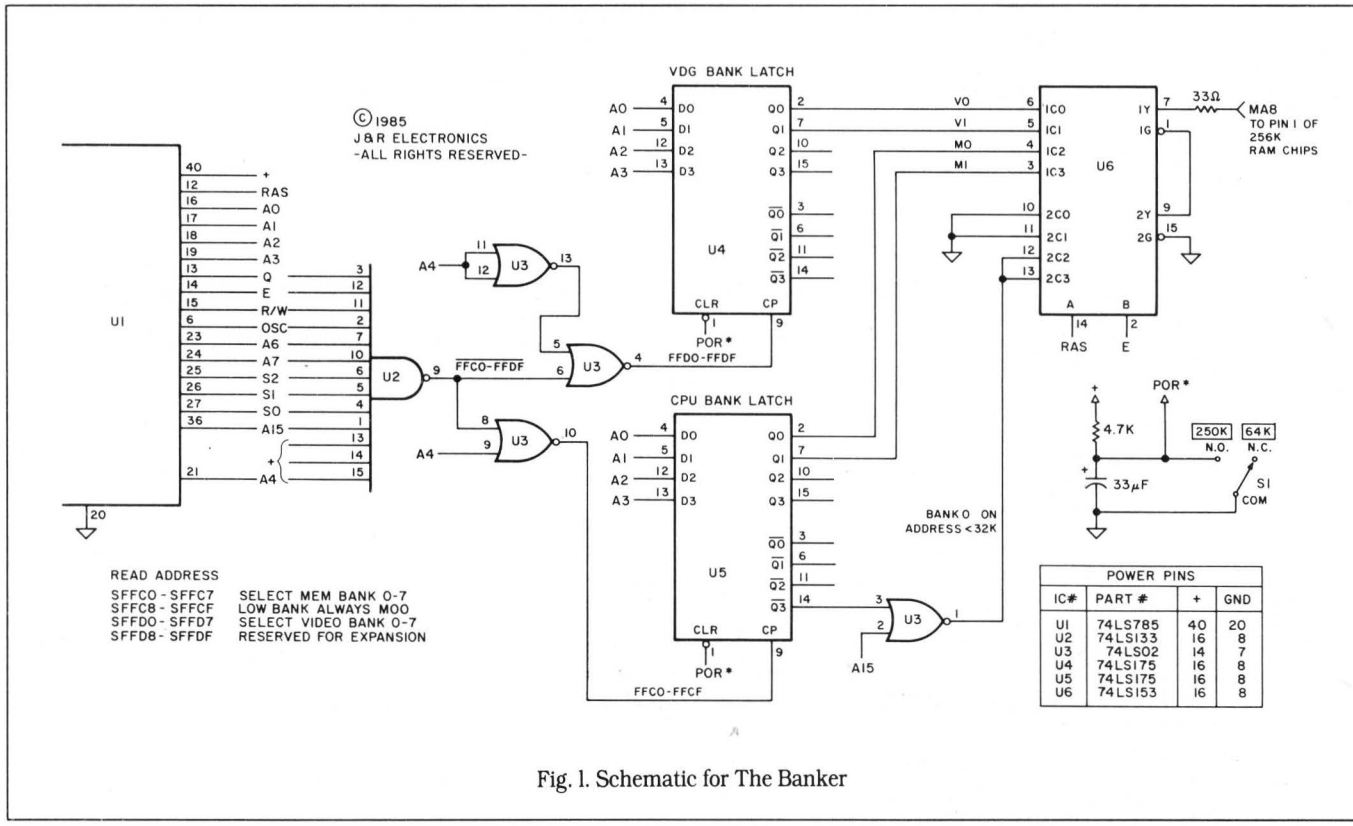


Fig. 1. Schematic for The Banker

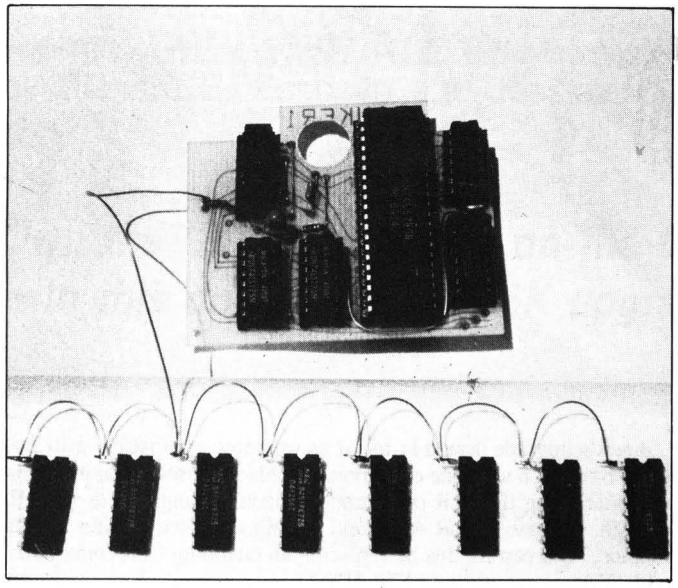


Photo 1. The Completed Unit

documentation. A software package is available separately for $19.95. It comes on tape or disk and contains the software from this article, RAM disk software, and demos. If you order from J & R, indicate whether you have a disk or tape system, original CoCo or CoCo 2.

Assuming you have a 64K-configured CoCo, the upgrade procedure is quite simple:

- Remove the eight 4164 dynamic RAMs.
- Remove the 6883 SAM chip.
- After bending the pin 1's of the eight 41256 chips so that they are in a horizontal position, place the chips in conductive foam and solder a 30-guage insulated wire to each of the bent pins, connecting them together so that you can install them in the sockets vacated by the 4164s.
- Install The Banker into the 6883 SAM socket, lining up the notches with each other.
- Install the 41256 RAMs into the 4164 sockets.
- Solder a wire so that it connects the point indicated on the schematic with the wire connecting the pin 1's of the new 41256 memory chips.

Be sure to orient the new chips with the notches in the sockets. A backward chip can wreak havoc upon your SAM, PIAs, and CPU. Solder splashes in strategic places can do likewise. Photo 2 shows the complete unit installed.

With The Banker installed, the SAM is programmed as usual by POKEing values into the range $FFC0 to $FFDF. (The actual values POKEd are irrelevant.) The Banker is programmed in much the same way as the SAM, except PEEK is used instead of POKE according to

74LS785 (new SAM chip): Radio Shack no. MX-6433—$14.95
74LS133: available at most electronics stores
74LS02: available at most electronics stores
74LS175 (two needed): available at most electronics stores
74LS153: available at most electronics stores
41256 (150 ns) dynamic RAMs (eight needed) *

* The best price we've seen for these chips as of June 19, 1985, was $4.32 each from Microprocessors Unlimited, 24000 S. Peoria Ave., Beggs, OK 74421, Telephone 918-267-4961. Since the prices on these RAMs are quite volatile, call for the latest price before placing an order.

Table l. 256K Upgrade Parts List.

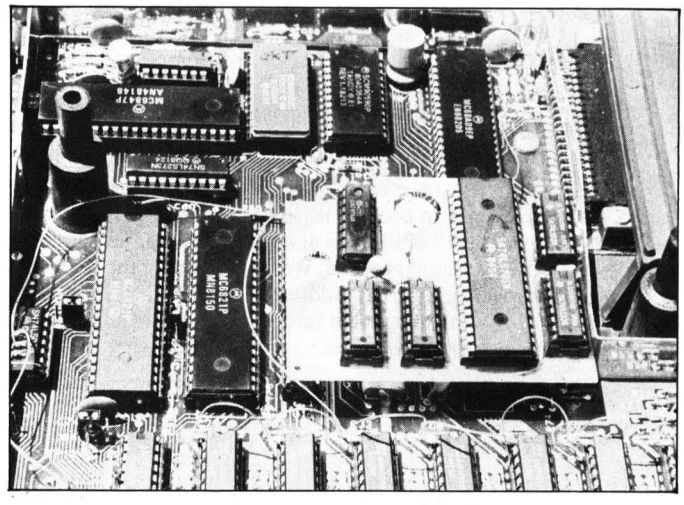


Photo 2. The Complete Unit Installed

$FFFF $FF00	I/O								
$FEFF $8000	Map Type 0 ROM	Map Type 1 >0	Map Type 1 >1	Map Type 1 >2	Map Type 1 >3	ROM	ROM	ROM	ROM
$7FFF $0000	Page = 0 <0	Page = 0 <0	Page = 0 <1	Page = 0 <2	Page = 0 <3	Page = 1 >0	Page = 1 >1	Page = 1 >2	Page = 1 >3

> = Upper Bank ($8000–$FEFF)
< = Lower Bank ($0000–$7FFF)

4 Banks of 32K in Upper RAM—Step 1
4 Banks of 64K—Steps 1 and 2
8 Banks of 32K in Lower RAM—Steps 1, 2, and 3

Step 1. Using Map Type 1 and $FFC8–$FFCF, transfer from <0 to >0 through >3
Step 2. Using Map Type 1 and $FFC0–$FFC7, transfer from >1 to <1, >2 to <2, >3 to <3
Step 3. Using Map Type 0, Page Bit and $FFC0–$FFC7

Fig. 2. The Banker Memory Map

the memory map in Table 2. Figure 2 displays a memory map that shows the possibilities available, given the combinations of PEEKs to the addresses in Table 2.

Below are the PEEKs needed to change the graphics memory page accessed by the VDG:

PEEK	VDG Memory Bank
$FFD0	0
$FFD1	1
$FFD2	2
$FFD3	3
$FFD4	4
$FFD5	5
$FFD6	6
$FFD7	7
$FFD8–$FFDF	Reserved for future expansion

PEEK	Lower bank ($0000–$7FFF)	Upper bank ($8000–$FEFF)
$FFC0	0	0
$FFC1	1	1
$FFC2	2	2
$FFC3	3	3
$FFC4*	4	4
$FFC5*	5	5
$FFC6*	6	6
$FFC7*	7	7
$FFC8	0	0
$FFC9	0	1
$FFCA	0	2
$FFCB	0	3
$FFCC*	0	4
$FFCD*	0	5
$FFCE*	0	6
$FFCF*	0	7

* Reserved for 512K upgrade

Note: must be in Map Type I to use upper banks

Table 2. CPU Memory

Software

Program Listing 1, Banker Test, is a POKE routine that gives you a machine-language program that tests your new memory. It also works for 64K upgrades; just specify the tests for bank 0 only.

Program Listing 2, Banker 32K, is an Assembly-language routine that demonstrates how to initialize the upper four 32K banks of RAM memory.

Program Listing 3, Bank Switcher, switches upper banks and alters Basic's prompts to indicate which bank is being used. Use it in conjunction with the assembled version of Listing 2.

Program Listing 4, Banker Backup, generates a machine-language program that lets you copy an entire Tandy-format disk on a single-drive machine without swapping disks. It copies the entire disk into RAM, and after the first copy, you have the option of copying the contents of RAM to additional disks.

If you have difficulty securing parts or have questions concerning assembly, write J & R Electronics. Please enclose a stamped, self-addressed envelope for a reply.■

See program listing on page 46

Address correspondence to J & R Electronics, P.O. Box 2572, Columbia, MD 21045.

Circle Reader Service card #70

NEW!



MAROONED!
By Steve Hartford
Sitting on the back porch one afternoon, you see a strange, flashing UFO descend from the clouds & land out in the corn field. Being the curious type, you run out to investigate and find a spaceship with it's hatch open...as you step inside, the hatch closes and the ship takes off! You must find a way to get back home. A great graphics adventure! 32K & one disk drive required.
Disk or Amdek $29.95



Blackjack Dealer
Feeler Dealer
These two programs help you develop your Blackjack skill and strategy. In **Blackjack Dealer,** the computer deals the cards and plays the dealer's hand against you. **Feeler Dealer** enables you to test your strategy by playing the desired number of hands using your techniques & tendencies. A great teacher for new Blackjack players and a valuable tool for the veteran player. Both programs included. 32K extended.
Tape $24.95 Disk $29.95



EAGLE
A graphic-enhanced lunar simulator. The pilot breaks out of lunar orbit and attempts a soft landing on the lunar surface. Joysticks control thrust and craft altitude and information is continually displayed on horizontal and vertical velocities, acceleration values, vertical and horizontal distances from target, fuel consumption and much more. Disk version allows choice of landing site between Mars and Earth's moon. A great tool for the future astronaut or physicist. 32K, 2 joysticks required.
Tape - $24.95 Disk - $29.95

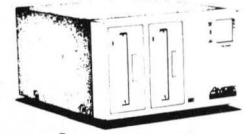
Sketchpad $19.95
Graphics Drawing Program - 32K Disk
ALPHA -40+ $19.95
Formats 40+ Tracks, Much More! 32K Disk
Testmaker $29.95
Makes T/F, Mult. Choice Tests - Disk Only
Maycode $24.95
6809 Disassembler - Add $5.00 For Disk

P51 Mustang 32K	$23.75
Worlds Of Flight	$23.75
Sailor Man 64K	$23.75
Trekboer 32K	$19.75
Tut's Tomb 32K	$19.75
Zookey Typing Tutor	$19.75
To Preserve Quandic Disk • 32K	$29.75
LIZPACK Disk • Stat. Anal.	$145.00
Super Screen Machine	$33.75

Add $3.00 For Disk

This Month's Special!
VIP Software
VIP Calc $49.75
VIP Terminal $34.75
VIP Speller $34.75
Limited To Stock On Hand.
Send For Our Free Catalog!

Factory Closeout!
Amdek Dual 3" Disk Drive



While Supply Lasts! $199 While Supply Lasts!
Includes 10 Diskettes & Cable
See This Month's Special!

Monitors
Amdek Color 1+ $199
Reconditioned - 90 Day Warranty
Amdek Color 300 $239
Composite, 300x260 Res.
Amdek Color 500 $339
Composite/RGB
Video Driver $27.95
Mark Data • Works On All Models
With Monitor Purchase - $25.95
Reverse Video Switch $11.95
Works On All Models With
Plug-In VDG Chip

$1.00 Shipping!
We'll Ship Your Order
To Anywhere In The USA
For $1.00
Software Only - Hardware Shipped At Cost
Outside USA - $2.00/ Program ($6.00 Max.)
Authors! Ask About Our Royalty Program

Saguaro Software
We've Moved!
(303) 728-4937
C.O.D. Orders Add $3.00
P.O. Box 1864 • Telluride, CO 81435
Colorado Residents Add 7% Tax

```
7Ø2Ø PRINT#-2,"C";P(P)
7Ø3Ø RETURN
8ØØØ '***DEFINE NO PRINT COLOR**

8ØØ1 'FOUR COLOR CHECK
8Ø1Ø IF R=Ø OR R=2 OR R=4 THEN 8
5ØØ
8Ø2Ø Z=Ø:IF CS=1 THEN Z=4
8Ø3Ø B1=99:B2=99:B3=99:B4=99
8Ø4Ø IF P(Z+1)=1Ø THEN B1=Z+1
8Ø5Ø IF P(Z+2)=1Ø THEN B2=Z+2
8Ø6Ø IF P(Z+3)=1Ø THEN B3=Z+3
8Ø7Ø IF P(Z+4)=1Ø THEN B4=Z+4
8Ø8Ø RETURN
85ØØ 'TWO COLOR CHECK
8505 B=99
851Ø IF CS=Ø AND P(Ø)=1Ø THEN B=
Ø
8515 IF CS=Ø AND P(1)=1Ø THEN B=
1
852Ø IF CS=1 AND P(5)=1Ø THEN B=
5
8525 IF CS=1 AND P(Ø)=1Ø THEN B=
Ø
854Ø RETURN
9ØØØ '*SCREEN GRAPHIC DISPLAY*
9Ø1Ø SCREEN1,CS
9Ø2Ø LINEINPUT Q$
9Ø3Ø CLS
9Ø4Ø SCREENØ
9Ø5Ø GOTO 4ØØØ
1ØØØØ SOUND1ØØ,1ØØ
1ØØ1Ø PRINT#-2,CHR$(17)
1ØØ2Ø END
```

The Fat CoCo

(Article on page 28)

Program Listing 1. Banker Test

```
1Ø 'BANKRTST.BAS V1.Ø
2Ø CLEAR 2ØØ,&H77FF
3Ø LA=&H78ØØ:EA=LA+&H33Ø:TA=LA
4Ø CLS
5Ø PRINT " "STRING$(3Ø,"*")
6Ø PRINT " *          BANKER TEST
      *"
7Ø PRINT " *       MEMORY BANK TE
ST     *"
8Ø PRINT " *
      *"
9Ø PRINT " *     J & R ELECTRONI
CS      *"
1ØØ PRINT " *          (C) 1985
      *"
11Ø PRINT " *   ALL RIGHTS RESER
VED     *"
12Ø PRINT " "STRING$(3Ø,"*")
13Ø INPUT" PRESS A KEY ";XX$
14Ø CLS
15Ø PRINT "      MENU SELECTIONS
"
16Ø PRINT"1. TEST UPPER BANK
17Ø PRINT"2. TEST LOWER BANK
18Ø PRINT"3. LOAD TEST PROGRAM
19Ø PRINT"4. QUIT
2ØØ INPUT" YOUR CHOICE <1,2,3,4>
 ";YC
21Ø IF YC<1 OR YC>4 THEN 14Ø
22Ø IF (YC = 1 OR YC= 2) AND (PE
EK(LA) <> &H16) THEN GOSUB 44Ø
23Ø ON YC GOSUB 26Ø,29Ø,44Ø,25Ø
24Ø GOTO13Ø
25Ø END'**********************
26Ø GOSUB32Ø'GET BANK #
27Ø POKE LA+17,255:EXEC LA
28Ø RETURN
29Ø GOSUB32Ø'GET BANK #
3ØØ POKE LA+&H11,Ø:EXEC LA
31Ø RETURN
32Ø INPUT"SELECT BANK <Ø,1,2,3>
";BK
33Ø IF BK<Ø OR BK>3 THEN 32Ø
34Ø POKE LA+9,BK
35Ø RETURN
36Ø INPUT"FILE NAME TO SAVE ";FL
$
37Ø INPUT"TAPE OR DISK <T*/D> ";
TD$
38Ø PRINT"SAVE "FL$ "TO";:IF TD$
="D" THEN PRINT"DISK" ELSE PRINT
"TAPE"
39Ø INPUT"IS THAT CORRECT ";XX$
4ØØ IF XX$ <> "Y" THEN 36Ø
41Ø IF TD$="D" THEN SAVEM FL$,LA
,EA,TA ELSE INPUT "READY TAPE";X
X$:CSAVEM FL$,LA,EA,TA
42Ø RETURN
43Ø END
44Ø PRINT"POKEING PROGRAM INTO M
EMORY"
45Ø RESTORE
46Ø FOR I=LA TO EA
47Ø READ X$
48Ø POKE I,VAL("&H"+X$)
49Ø IF I/16 = INT(I/16) THEN PRI
NT".";
5ØØ NEXT I
51Ø PRINT:PRINT" LOADED "
52Ø RETURN
53Ø DATA16,Ø1,ØB,16,ØØ,53,16,ØØ
54Ø DATA21,ØØ,FF,FF,FF,ØØ,ØØ,7F
55Ø DATA  FE,ØØ,ØØ,ØØ,7F,FE,8Ø,Ø
Ø
56Ø DATA  FE,FE,2Ø,DF,DF,ØØ,ØØ,D
F
57Ø DATA  FF,ØØ,ØØ,DF,FF,ØØ,ØØ,D
F
58Ø DATA  FF,ØØ,1A,5Ø,AE,8C,E7,A
F
59Ø DATA8C,DB,AE,8C,E3,AF,8C,D7
6ØØ DATA7F,FF,4Ø,B7,FF,DF,CC,FF
61Ø DATA  FF,ED,8C,C7,A7,8C,C3,C
E
62Ø DATA  FF,CØ,A6,8C,BC,84,Ø7,8
A
63Ø DATAØ8,A5,C6,8D,62,B7,FF,DE
64Ø DATA39,1A,5Ø,AE,8C,B4,AF,8C
65Ø DATA  AC,AE,8C,BØ,AF,8C,A8,7
F
66Ø DATA  FF,4Ø,B7,FF,DF,CC,FF,F
F
67Ø DATA  ED,8C,98,A7,8C,94,CE,F
F
68Ø DATA  CØ,A6,8C,8D,84,Ø7,8A,Ø
8
69Ø DATA  A5,C6,17,ØØ,5F,3Ø,8D,Ø
Ø
7ØØ DATAØ6,3Ø,89,8Ø,ØØ,6E,84,A6
71Ø DATA8D,FF,76,84,Ø7,CE,FF,CØ
72Ø DATA  A5,C6,8D,1B,CE,FF,CØ,A
6
73Ø DATA8D,FF,66,84,Ø7,8A,Ø8,A5
74Ø DATA  C6,3Ø,8D,ØØ,Ø6,3Ø,89,8
Ø
75Ø DATAØØ,6E,84,B7,FF,DE,39,AE
76Ø DATA8D,FF,52,A6,84,43,B7,Ø4
77Ø DATAØØ,A7,84,A1,84,26,14,43
78Ø DATA  B7,Ø4,ØØ,A7,84,A1,84,2
6
79Ø DATAØA,3Ø,Ø1,AC,8D,FF,38,2F
8ØØ DATA  E2,2Ø,Ø8,6F,8D,FF,2B,A
F
81Ø DATA8D,FF,28,39,3Ø,8D,ØØ,24
82Ø DATA  AF,8D,ØØ,2Ø,3Ø,8D,FF,1
Ø
83Ø DATA  CE,FF,CØ,A6,8D,FF,12,8
4
84Ø DATAØ7,8A,Ø8,A5,C6,A6,84,A7
85Ø DATA89,8Ø,ØØ,3Ø,Ø1,AC,8D,ØØ
86Ø DATAØ3,2F,F2,39,ØØ,ØØ,3Ø,8D
87Ø DATAØØ,DC,17,ØØ,9Ø,6D,8D,FE
88Ø DATA  F8,27,ØC,17,FF,ØC,3Ø,8
D
89Ø DATAØ1,C8,17,ØØ,8Ø,2Ø,ØA,17
9ØØ DATA  FF,2F,3Ø,8D,Ø1,C4,17,Ø
Ø
91Ø DATA74,3Ø,8D,Ø1,C5,17,ØØ,6D
92Ø DATA  A6,8D,FE,CD,17,ØØ,3B,A
6
93Ø DATA8D,FE,C7,43,27,ØC,3Ø,8D
94Ø DATAØ1,C1,17,ØØ,58,17,ØØ,ØD
95Ø DATA2Ø,Ø7,3Ø,8D,Ø1,AB,17,ØØ
96Ø DATA4C,17,ØØ,3C,39,A6,8D,FE
97Ø DATA  AA,17,ØØ,16,A6,8D,FE,A
4
98Ø DATA17,ØØ,ØF,17,ØØ,Ø5,39,8D
99Ø DATAØØ,8D,ØØ,86,2Ø,AD,9F,AØ
1ØØØ DATA Ø2,39,34,Ø2,46,46,46,4
6
1Ø1Ø DATA 17,ØØ,Ø6,35,Ø2,17,ØØ,Ø
1
1Ø2Ø DATA 39,84,ØF,81,ØA,2D,Ø2,8
B
1Ø3Ø DATA Ø7,8B,3Ø,AD,9F,AØ,Ø2,3
9
1Ø4Ø DATA 86,ØD,AD,9F,AØ,Ø2,86,Ø
A
1Ø5Ø DATA AD,9F,AØ,Ø2,39,A6,8Ø,2
7
1Ø6Ø DATA Ø6,AD,9F,AØ,Ø2,2Ø,F6,3
9
1Ø7Ø DATA 34,3Ø,6F,A4,17,FF,EE,1
7
1Ø8Ø DATA ØØ,13,26,Ø2,6C,A4,3Ø,8
D
1Ø9Ø DATA Ø1,56,17,FF,EØ,17,ØØ,Ø
5
11ØØ DATA 35,3Ø,26,E4,39,17,ØØ,1
7
111Ø DATA 81,59,27,Ø6,81,4E,27,Ø
2
112Ø DATA 2Ø,F3,AD,9F,AØ,Ø2,81,5
9
113Ø DATA 39,AD,9F,AØ,ØØ,27,Ø6,A
D
114Ø DATA 9F,AØ,ØØ,27,FA,39,ØD,Ø
D
115Ø DATA ØD,ØD,ØD,ØD,2Ø,2A,2A,2
A
116Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
117Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
118Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
119Ø DATA 2A,2Ø,ØD,2Ø,2A,2Ø,2Ø,2
Ø
```

```
12ØØ DATA 2Ø,2Ø,2Ø,42,41,4E,4B,5
4
121Ø DATA 45,53,54,2Ø,5Ø,52,4F,4
7
122Ø DATA 52,41,4D,2Ø,2Ø,2Ø,2Ø,2
A
123Ø DATA 2Ø,ØD,2Ø,2A,2Ø,2Ø,2Ø,2
Ø
124Ø DATA 2Ø,2Ø,43,4F,5Ø,59,52,4
9
125Ø DATA 47,48,54,2Ø,31,39,38,3
5
126Ø DATA 2Ø,2Ø,2Ø,2Ø,2Ø,2Ø,2A,Ø
D
127Ø DATA 2Ø,2A,2Ø,2Ø,2Ø,2Ø,2Ø,2
Ø
128Ø DATA 4A,2Ø,26,2Ø,52,2Ø,45,4
C
129Ø DATA 45,43,54,52,4F,4E,49,4
3
13ØØ DATA 53,2Ø,2Ø,2Ø,2A,ØD,2Ø,2
A
131Ø DATA 2Ø,2Ø,2Ø,2Ø,2Ø,2Ø,5Ø,2
E
132Ø DATA 4F,2E,42,4F,58,2Ø,32,3
5
133Ø DATA 37,32,2Ø,2Ø,2Ø,2Ø,2Ø,2
Ø
134Ø DATA 2Ø,2Ø,2A,ØD,2Ø,2A,2Ø,2
Ø
135Ø DATA 2Ø,2Ø,2Ø,2Ø,43,4F,4C,5
5
136Ø DATA 4D,42,49,41,2C,2Ø,4D,4
4
137Ø DATA 2E,2Ø,32,31,3Ø,34,35,2
Ø
138Ø DATA 2A,ØD,2Ø,2A,2Ø,2Ø,2Ø,2
Ø
139Ø DATA 2Ø,2Ø,41,4C,4C,2Ø,52,4
9
14ØØ DATA 47,48,54,53,2Ø,52,45,5
3
141Ø DATA 45,52,56,45,44,2Ø,2A,Ø
D
142Ø DATA 2Ø,2A,2A,2A,2A,2A,2A,2
A
143Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
144Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
145Ø DATA 2A,2A,2A,2A,2A,ØD,ØD,Ø
D
146Ø DATA ØD,ØØ,2Ø,55,5Ø,5Ø,45,5
2
147Ø DATA 2Ø,ØØ,2Ø,4C,4F,57,45,5
2
148Ø DATA 2Ø,ØØ,2Ø,42,41,4E,4B,2
Ø
149Ø DATA ØØ,2Ø,5Ø,41,53,53,45,4
4
15ØØ DATA 2Ø,ØD,ØØ,2Ø,66,61,69,6
C
151Ø DATA 65,64,2Ø,41,54,2Ø,24,Ø
Ø
152Ø DATA 2Ø,41,52,45,2Ø,59,4F,5
5
153Ø DATA 2Ø,53,55,52,45,2Ø,3C,5
9
154Ø DATA 2F,4E,3E,2Ø,3F,2Ø,ØØ,Ø
Ø
155Ø DATA ØØ,FF,DF,ØØ,ØØ,DF,FF,Ø
Ø
156Ø END 'BANKRTST.BAS
```

Program Listing 2. Banker 32K
(Not available on the Instant CoCo cassette.)

```
10 NAM B256
20 *BANKER 256K
30 *COPYRIGHT 1985
40 *J & R ELECTRONICS
50 *ALL RIGHTS RESERVED
60 *
70 ORG $0E00
80 ORCC #$50
90 LDX #$8000
100 LDY #$FFC0 *BANK 0
110 COPY STB $FFDE
120 LDB 0,Y
130 LDA 0,X
140 LDB $08,Y *BANK 00/00
150 STB $FFDF
160 STA 0,X
170 LDB $09,Y *BANK 01/00
180 STB $FFDF
190 STA 0,X
200 LDB $0A,Y *BANK 02/00
210 STB $FFDF
220 STA 0,X
230 LDB $0B,Y *BANK 03/00
240 STB $FFDF
250 STA 0,X
260 LDB 0,Y *BANK 00
270 STB $FFDF
280 LEAX 1,X
290 CMPX #$FF00
300 BNE COPY
310 LDX #$ABEE *OK PROMPT
320 LDB 0,Y *BANK 00
330 STB $FFDF
340 LDA #'>
480 STA 0,X
490 LDA #'0 BANK #
500 STA 1,X
510 LDB $09,Y *BANK 01/100
520 STB $FFDF
530 LDA #'>
540 STA 0,X
550 LDA #'1
560 STA 1,X
570 LDB $0A,Y *BANK 02/00
580 STB $FFDF
590 LDA #'>
600 STA 0,X
610 LDA #'2
620 STA 1,X
630 LDB $0B,Y *BANK 03/00
640 STB $FFDF
650 LDA #'>
660 STA 0,X
670 LDA #'3
680 STA 1,X
690 LDB 0,Y *BANK 00
700 STB $FFDF
710 RTS
720 END
```

Program Listing 3. Bank Switcher
(Not available on the Instant CoCo cassette.)

```
1Ø REM NEEDS "B256/BIN" IN RAM
2Ø CLS
3Ø A=&HFFC8:V=&HFFDØ
4Ø I=PEEK(A)
5Ø FOR I=Ø TO 3
6Ø CLS
7Ø IF PEEK(A+I) THEN
8Ø IF PEEK(V) THEN
9Ø PRINT "BANK"CHR$(PEEK(&HABEF)
)" ";
1ØØ PRINT:PRINT"PRESS A KEY"
11Ø IF INKEY$="" THEN 11Ø
12Ø NEXT
13Ø GOTO5Ø
```

Program Listing 4. Banker Backup

```
1Ø 'BANKRBAK.BAS V1.Ø
2Ø CLEAR 2ØØ,&H77FF
3Ø LA=&H78ØØ:EA=LA+&H48F:TA=LA
4Ø CLS
5Ø PRINT " "STRING$(3Ø,"*")
6Ø PRINT " *          BANKER BACK
UP       *"
7Ø PRINT " *       MEMORY BANK TE
ST       *"
8Ø PRINT " *
         *"
9Ø PRINT " *      J & R ELECTRONI
CS       *"
1ØØ PRINT " *          (C) 1985
         *"
11Ø PRINT " *   ALL RIGHTS RESER
VED      *"
12Ø PRINT " "STRING$(3Ø,"*")
13Ø INPUT" PRESS A KEY ";XX$
14Ø CLS
15Ø PRINT "      MENU SELECTIONS
"
16Ø PRINT"1. POKE PROGRAM TO MEM
ORY
17Ø PRINT"2. MACHINE LANGUAGE SA
VE
18Ø PRINT"3. QUIT
19Ø INPUT" YOUR CHOICE <1,2,3> "
;YC
2ØØ IF YC<1 OR YC>3 THEN 14Ø
21Ø IF YC= 2 AND (PEEK(LA) <> &H
16) THEN GOSUB 32Ø
22Ø ON YC GOSUB 32Ø,25Ø,24Ø
23Ø GOTO13Ø
24Ø END'**********************
25Ø INPUT"FILE NAME TO SAVE ";FL
$
26Ø INPUT"TAPE OR DISK <T*/D> ";
TD$
27Ø PRINT"SAVE "FL$ "TO";:IF TD$
="D" THEN PRINT"DISK" ELSE PRINT
"TAPE"
28Ø INPUT"IS THAT CORRECT ";XX$
29Ø IF XX$ <> "Y" THEN 25Ø
3ØØ IF TD$="D" THEN SAVEM FL$,LA
,EA,TA ELSE INPUT "READY TAPE";X
X$:CSAVEM FL$,LA,EA,TA
31Ø RETURN
32Ø PRINT"POKEING PROGRAM INTO M
EMORY"
33Ø RESTORE
34Ø FOR I=LA TO EA
35Ø READ X$
36Ø POKE I,VAL("&H"+X$)
37Ø IF I/16 = INT(I/16) THEN PRI
NT".";
38Ø NEXT I
```

Listing continued

```
39Ø PRINT:PRINT" LOADED "
4ØØ RETURN
41Ø FOR I=&H78ØØ TO &H7C8F
42Ø READ X$
43Ø POKE I,VAL("&H"+X$)
44Ø NEXT I
45Ø DATA16,ØØ,64,16,ØØ,1D,ØØ,ØØ
46Ø DATAØ1,ØØ,ØØ,ØØ,ØØ,FE,ØØ,ØØ
47Ø DATAØØ,ØØ,ØØ,DF,FF,ØØ,ØØ,DF
48Ø DATA  DF,2Ø,ØØ,FF,DF,ØØ,ØØ,D
F
49Ø DATA  FF,ØØ,ØØ,1A,5Ø,AE,8C,E
1
5ØØ DATA  EE,8C,E6,1Ø,8E,FF,CØ,A
6
51Ø DATA8C,D4,84,Ø7,31,A6,A5,28
52Ø DATA  A6,C4,81,Ø2,27,Ø7,81,Ø
3
53Ø DATA27,13,A5,28,39,EE,44,C6
54Ø DATAØØ,A6,CØ,A5,A4,A7,8Ø,A5
55Ø DATA28,5A,26,F5,39,EE,44,C6
56Ø DATAØØ,A5,A4,A6,8Ø,A5,28,A7
57Ø DATA  CØ,5A,26,F5,39,ØØ,ØØ,1
A
58Ø DATA5Ø,7F,FF,4Ø,B7,FF,DE,FE
59Ø DATA  CØ,Ø6,EF,8C,9C,3Ø,8D,Ø
2
6ØØ DATA3A,17,Ø1,E6,3Ø,8D,Ø3,2F
61Ø DATA17,Ø1,DF,3Ø,8D,Ø3,58,17
62Ø DATAØ1,D8,17,Ø2,1F,17,Ø1,C1
63Ø DATA  FE,CØ,Ø6,CC,Ø6,ØØ,ED,4
4
64Ø DATA86,Ø2,A7,C4,17,ØØ,51,3Ø
65Ø DATA8D,Ø2,1Ø,17,Ø1,BC,3Ø,8D
66Ø DATAØ3,1B,17,Ø1,B5,3Ø,8D,Ø3
67Ø DATA2E,17,Ø1,AE,17,Ø1,F5,17
68Ø DATAØ1,97,FE,CØ,Ø6,86,Ø3,A7
69Ø DATA  C4,17,ØØ,2C,3Ø,8D,Ø1,E
B
7ØØ DATA17,Ø1,97,3Ø,8D,Ø3,1F,17
71Ø DATAØ1,9B,6D,8D,Ø1,BB,1Ø,27
72Ø DATAØØ,12,3Ø,8D,Ø3,28,17,Ø1
73Ø DATA8C,6D,8D,Ø1,AC,1Ø,26,FF
74Ø DATA  B6,16,FF,7B,B7,FF,DE,3
9
75Ø DATA  FE,CØ,Ø6,86,ØØ,A7,42,A
7
76Ø DATA8D,FF,14,3Ø,8D,ØØ,E4,C6
77Ø DATAØ2,A1,85,2C,Ø3,5A,2Ø,F9
78Ø DATA5C,E7,8D,FE,F9,5A,AØ,85
79Ø DATA  A7,8D,FE,F3,4C,8E,ØØ,Ø
Ø
8ØØ DATA4A,27,Ø6,3Ø,89,12,ØØ,2Ø
81Ø DATA  F7,AF,8D,FE,E4,3Ø,89,Ø
Ø
82Ø DATA  FF,AF,8D,FE,DE,86,Ø1,A
7
83Ø DATA43,A7,8D,FE,DB,A6,C4,81
84Ø DATAØ2,27,4D,81,Ø3,1Ø,26,ØØ
85Ø DATA9B,DC,88,ED,8D,Ø3,46,3Ø
86Ø DATA8D,Ø3,Ø8,17,Ø1,14,17,ØØ
87Ø DATA  C5,EC,8D,Ø3,38,DD,88,1
7
88Ø DATAØØ,8C,AE,8D,FE,AF,AD,Ø3
89Ø DATA  B7,FF,DE,AD,9F,CØ,Ø4,1
A
9ØØ DATA5Ø,FE,CØ,Ø6,A6,46,84,4Ø
91Ø DATA1Ø,27,ØØ,3F,3Ø,8D,Ø2,F4
92Ø DATA17,ØØ,E7,3Ø,8D,Ø2,6Ø,17
93Ø DATAØØ,EØ,17,Ø1,27,16,ØØ,54
94Ø DATA  B7,FF,DE,DC,88,ED,8D,Ø
2
95Ø DATA  FC,3Ø,8D,Ø2,A5,17,ØØ,C
A
96Ø DATA17,ØØ,7B,EC,8D,Ø2,EE,DD
97Ø DATA88,AD,9F,CØ,Ø4,1A,5Ø,17
98Ø DATAØØ,3C,AE,8D,FE,5F,AD,Ø3
99Ø DATA  B7,FF,DE,AE,8D,FE,52,3
Ø
1ØØØ DATA 89,Ø1,ØØ,AF,8D,FE,4A,3
Ø
1Ø1Ø DATA 89,ØØ,FF,AF,8D,FE,44,F
E
1Ø2Ø DATA CØ,Ø6,A6,43,4C,81,12,1
Ø
1Ø3Ø DATA 2F,FF,5C,A6,42,4C,81,2
2
1Ø4Ø DATA 1Ø,2F,FF,19,B7,FF,DE,7
F
1Ø5Ø DATA FF,4Ø,39,ØØ,ØE,1C,34,7
6
1Ø6Ø DATA 3Ø,8D,FE,79,AF,8D,FE,7
5
1Ø7Ø DATA 3Ø,8D,FE,ØC,1Ø,AE,8D,F
E
1Ø8Ø DATA 14,CE,FF,CØ,A6,8D,FE,Ø
6
1Ø9Ø DATA 84,Ø7,8A,Ø8,A5,C6,B7,F
F
11ØØ DATA DF,A6,8Ø,A7,AØ,AC,8D,F
E
111Ø DATA 54,2F,F6,35,76,39,A6,8
D
112Ø DATA FD,F5,17,ØØ,16,A6,8D,F
D
113Ø DATA EF,17,ØØ,ØF,17,ØØ,Ø5,3
9
114Ø DATA 8D,ØØ,8D,ØØ,86,2Ø,AD,9
F
115Ø DATA AØ,Ø2,39,34,Ø2,46,46,4
6
116Ø DATA 46,17,ØØ,Ø6,35,Ø2,17,Ø
Ø
117Ø DATA Ø1,39,84,ØF,81,ØA,2D,Ø
2
118Ø DATA 8B,Ø7,8B,3Ø,AD,9F,AØ,Ø
2
119Ø DATA 39,34,Ø2,86,ØD,AD,9F,A
Ø
12ØØ DATA Ø2,86,ØA,AD,9F,AØ,Ø2,3
5
121Ø DATA Ø2,39,A6,8Ø,27,Ø6,AD,9
F
122Ø DATA AØ,Ø2,2Ø,F6,39,34,1Ø,6
F
123Ø DATA 8D,ØØ,1E,17,FF,EC,17,Ø
Ø
124Ø DATA 19,26,Ø4,6C,8D,ØØ,12,1
7
125Ø DATA FF,CF,3Ø,8D,Ø1,9C,17,F
F
126Ø DATA D9,17,ØØ,Ø6,35,1Ø,26,D
D
127Ø DATA 39,ØØ,17,ØØ,17,81,59,2
7
128Ø DATA Ø6,81,4E,27,Ø2,2Ø,F3,A
D
129Ø DATA 9F,AØ,Ø2,81,59,39,AD,9
F
13ØØ DATA AØ,ØØ,27,Ø6,AD,9F,AØ,Ø
Ø
131Ø DATA 27,FA,39,ØD,ØD,ØD,ØD,Ø
D
132Ø DATA ØD,2Ø,2A,2A,2A,2A,2A,2
A
133Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
134Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
135Ø DATA 2A,2A,2A,2A,2A,2A,2Ø,Ø
D
136Ø DATA 2Ø,2A,2Ø,2Ø,2Ø,2Ø,2Ø,2
Ø
137Ø DATA 42,41,4E,4B,52,42,41,4
B
138Ø DATA 2Ø,5Ø,52,4F,47,52,41,4
D
139Ø DATA 2Ø,2Ø,2Ø,2Ø,2A,2Ø,ØD,2
Ø
14ØØ DATA 2A,2Ø,2Ø,2Ø,2Ø,2Ø,2Ø,4
3
141Ø DATA 4F,5Ø,59,52,49,47,48,5
4
142Ø DATA 2Ø,31,39,38,35,2Ø,2Ø,2
Ø
143Ø DATA 2Ø,2Ø,2Ø,2A,ØD,2Ø,2A,2
Ø
144Ø DATA 2Ø,2Ø,2Ø,2Ø,2Ø,4A,2Ø,2
6
145Ø DATA 2Ø,52,2Ø,45,4C,45,43,5
4
146Ø DATA 52,4F,4E,49,43,53,2Ø,2
Ø
147Ø DATA 2Ø,2A,ØD,2Ø,2A,2Ø,2Ø,2
Ø
148Ø DATA 2Ø,2Ø,2Ø,5Ø,2E,4F,2E,4
2
149Ø DATA 4F,58,2Ø,32,35,37,32,2
Ø
15ØØ DATA 2Ø,2Ø,2Ø,2Ø,2Ø,2Ø,2Ø,2
A
151Ø DATA ØD,2Ø,2A,2Ø,2Ø,2Ø,2Ø,2
Ø
152Ø DATA 2Ø,43,4F,4C,55,4D,42,4
9
153Ø DATA 41,2C,2Ø,4D,44,2E,2Ø,3
2
154Ø DATA 31,3Ø,34,35,2Ø,2A,ØD,2
Ø
155Ø DATA 2A,2Ø,2Ø,2Ø,2Ø,2Ø,2Ø,4
1
156Ø DATA 4C,4C,2Ø,52,49,47,48,5
4
157Ø DATA 53,2Ø,52,45,53,45,52,5
6
158Ø DATA 45,44,2Ø,2A,ØD,2Ø,2A,2
A
159Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
16ØØ DATA 2A,2A,2A,2A,2A,2A,2A,2
A
161Ø DATA 2A,2A,2A,2A,2A,2A,2A,2
A
162Ø DATA 2A,2A,ØD,ØD,ØD,ØD,ØØ,2
Ø
163Ø DATA 49,4E,53,45,52,54,2Ø,5
3
164Ø DATA 4F,55,52,43,45,2Ø,44,4
9
165Ø DATA 53,4B,2Ø,ØD,ØØ,2Ø,49,4
E
166Ø DATA 53,45,52,54,2Ø,44,45,5
3
167Ø DATA 54,49,4E,41,54,49,4F,4
E
168Ø DATA 2Ø,44,49,53,4B,ØD,ØØ,2
Ø
169Ø DATA 5Ø,52,45,53,53,2Ø,41,2
Ø
17ØØ DATA 4B,45,59,2Ø,ØD,ØØ,ØD,2
```

```
Ø
171Ø DATA 41,4E,4F,54,48,45,52,2
Ø
172Ø DATA 43,4F,5Ø,59,2Ø,3C,59,2
F
173Ø DATA 4E,3E,2Ø,3F,2Ø,ØØ,ØD,2
Ø
174Ø DATA 43,4F,5Ø,59,2Ø,46,52,4
F
175Ø DATA 4D,2Ø,4D,45,4D,4F,52,5
9
176Ø DATA 2Ø,3C,59,2F,4E,3E,2Ø,3
F
177Ø DATA 2Ø,ØØ,ØD,2Ø,41,52,45,2
Ø
178Ø DATA 59,4F,55,2Ø,53,55,52,4
5
179Ø DATA 2Ø,3C,59,2F,4E,3E,2Ø,3
F
18ØØ DATA 2Ø,ØØ,2Ø,52,45,41,44,4
9
181Ø DATA 4E,47,2Ø,54,52,41,43,4
B
182Ø DATA 2F,53,45,43,54,4F,52,2
Ø
183Ø DATA 26,48,ØØ,2Ø,57,52,49,5
4
184Ø DATA 49,4E,47,2Ø,54,52,41,4
3
185Ø DATA 4B,2F,53,45,43,54,4F,5
2
186Ø DATA 2Ø,26,48,ØØ,ØD,2A,2A,2
Ø
187Ø DATA 44,69,73,6B,2Ø,69,73,2
Ø
188Ø DATA 57,72,69,74,65,2Ø,5Ø,7
2
189Ø DATA 6F,74,65,63,74,65,64,2
1
19ØØ DATA 2Ø,2A,2A,ØD,ØØ,Ø5,ØØ,F
F
191Ø END 'BANKRBAK,BAS
```

Matrix Maneuvers

(Article on page 40)

```
1Ø CLS8:PRINT@136,"matrix maneuv
ers";:SOUND15Ø,6:FORT=1TO25:NEXT
:SOUND15Ø,3:FORT=1TO25:NEXT:SOUN
D15Ø,3:HS=4453:SOUND194,12:PRINT
@224,"DESIGNED BY :";:PRINT@298,
"DAVE BENSON";:PRINT@33Ø,"P.O. B
OX 142";:PRINT@357,"GRANTSVILLE,
 MD. 21536";
11 PRINT@425,"(3Ø1) 895-5893";:F
ORT=1TO2ØØØ:NEXT
12 GOTO 2ØØ
9Ø CLS7:PRINT@256,"";
91 INPUT"ENTER SKILL LEVEL ";SK
99 BB$=STRING$(32,175)
1ØØ S=Ø:CLS3:FORT=1TO22
1Ø1 PRINT@T+38,CHR$(T+64);
1Ø2 NEXT
1Ø3 FORT=1TO14:IF T<1Ø THEN Y$="
Ø"+CHR$(48+T) ELSE Y$="1"+CHR$(3
8+T)
1Ø4 PRINT@(T*32)+35," ";Y$;" ";S
TRING$(22,43);" ";
1Ø5 NEXT
11Ø FORT=1TO SK+1
111 X=RND(22):Y=RND(14)
112 ADDR=(X+6)+((Y*32)+32)+1Ø24
113 GOSUB1ØØØ
114 NEXT
12Ø S=S+1
121 PRINT@Ø,"letter (a-v)";:X$=I
NKEY$
122 IF X$="" THEN 121 ELSE IF AS
C(X$)<65 OR ASC(X$)>86 THEN 91Ø
123 SOUND17Ø,5:X=ASC(X$)-64:PRIN
T@Ø,BB$;
124 PRINT@Ø,"NUMBER (Ø1-14)";:Y$
=INKEY$
125 IF Y$="" THEN 124 ELSE IF AS
C(Y$)=48 OR ASC(Y$)=49 THEN Y=VA
L(Y$)*1Ø:SOUND15Ø,1 ELSE 9ØØ
126 Y$=INKEY$:IF Y$="" THEN 126
ELSE IF ASC(Y$)=48 AND Y<>1Ø THE
N 9ØØ ELSE IF ASC(Y$)<48 OR ASC(
Y$)>57 THEN 9ØØ
127 IF Y>14 THEN 9ØØ ELSE SOUND1
7Ø,5:Y=VAL(Y$)+Y:PRINT@Ø,BB$;:AD
DR=(X+6)+((Y*32)+32)+1Ø24
128 GOSUB1ØØØ
129 GOSUB2ØØØ
13Ø GOTO12Ø
2ØØ CLS3:PRINT@256,"";:INPUT"DO
YOU WANT INSTRUCTIONS        (Y
OR N)";Q$
2Ø1 IF Q$="N" THEN 9Ø
25Ø CLS Ø:PRINT@2,"MATRIX MANEUVE
RS OPPERATES ON A22 (ACROSS) BY
14 (DOWN) GRID.";
251 PRINT@66,"THE OBJECT OF THIS
 GAME IS TO CLEAR THE BOARD OF E
VERY GREEN  +, WITH A BLACK BACK
GROUND.";
252 PRINT@162,"YOUR SCORE GETS H
IGHER EVRY    TIME YOU INCREASE T
HE SKILL      LEVEL. ALSO, THE LO
WER THE       NUMBER OF TRIES THE
 HIGHER YOUR SCORE.";
253 PRINT@322,"YOU WILL BE SHOWN
 A HIGH SCOREAT THE END OF THE G
AME, ALONG    WITH YOUR OWN SCORE
.";
254 GOSUB 8ØØ
26Ø CLSØ:PRINT@2,"TO INPUT COORD
INATES FIRST YOUENTER A LETTER (
POSITION LEFT TORIGHT). THEN YOU
 ENTER A NUMBER (POSITION TOP TO
 BOTTOM), THERE MUST BE A ZERO (
Ø) IN FRONT OF  ANY SINGLE-DIGIT
 NUMBER (1      THROUGH 9).";
261 GOSUB8ØØ
27Ø CLSØ:PRINT@2,"THE COORDINATE
 YOU ENTER WILL BE THE CENTER OF
 A 3 BY 3 GRID  WHERE REVERSING
OF COLORS TAKES PLACE, EITHER FR
OM BLACK TO     GREEN OR FROM GR
EEN TO BLACK.";
271 GOSUB 8ØØ
28Ø CLSØ:PRINT@2,"IF YOUR SQUARE
 RUNS OFF THE     SIDE OF THE GR
ID IT WILL COME   BACK ON THE OP
PISITE SIDE.";
282 PRINT@226,"IF YOUR SQUARE IS
 PLACED IN A   CORNER THE SQUARE
 WILL COME BACKAT THE THREE (3)
OTHER CORNERS.";
283 GOSUB 8ØØ
299 GOTO9Ø
799 END
8ØØ PRINT@483,"press <enter> to
continue";
81Ø IF INKEY$=CHR$(13) THEN RETU
RN ELSE 81Ø
9ØØ SOUND225,5:FORT=1TO25:NEXT:S
OUND2ØØ,5
9Ø1 PRINT@Ø,"please start again
...";
9Ø2 FORT=1TO1ØØØ:NEXT:PRINT@Ø,BB
$;
9Ø3 GOTO121
91Ø SOUND2ØØ,5:FORT=1TO25:NEXT:S
OUND175,5
911 PRINT@Ø,"PLEASE START AGAIN
...";
912 FORT=1TO1ØØØ:NEXT:PRINT@Ø,BB
$;
913 GOTO121
1ØØØ RESTORE:FOR L=1TO9:I=Ø:READ
D
1ØØ1 V=ADDR+D
1ØØ2 P=PEEK(V)
1ØØ3 I=ABS(P-15Ø)
1ØØ4 CA=Ø:CB=Ø:CX=Ø:CY=Ø:IF I=1Ø
7 OR I=43 THEN ELSE ON L GOTO 1Ø
1Ø,1Ø15,1Ø2Ø,1Ø25,1Ø3Ø,1Ø35,1Ø4Ø
,1Ø45,1Ø5Ø
1ØØ5 POKE V,I
1ØØ6 NEXT
1ØØ7 RETURN
1Ø1Ø CA=PEEK(ADDR-1):CB=PEEK(ADD
R-32)
1Ø11 IF CA<>43 THEN IF CA<>1Ø7 T
HEN CX=1
1Ø12 IF CB<>43 THEN IF CB<>1Ø7 T
HEN CY=1
1Ø13 IF CX=1 THEN V=V+22
1Ø14 IF CY=1 THEN V=V+448:GOTO1Ø
Ø2 ELSE GOTO 1ØØ2
1Ø15 V=V+448
1Ø16 GOTO1ØØ2
1Ø2Ø CA=PEEK(ADDR+1):CB=PEEK(ADD
R-32)
1Ø21 IF CA<>43 THEN IF CA<>1Ø7 T
HEN CX=1
1Ø22 IF CB<>43 THEN IF CB<>1Ø7 T
HEN CY=1
1Ø23 IF CX=1 THEN V=V-22
1Ø24 IF CY=1 THEN V=V+448:GOTO1Ø
Ø2 ELSE 1ØØ2
1Ø25 V=V+22
1Ø26 GOTO 1ØØ2
1Ø3Ø NEXT
1Ø31 RETURN
1Ø35 V=V-22
1Ø36 GOTO 1ØØ2
1Ø4Ø CA=PEEK(ADDR-1):CB=PEEK(ADD
R+32)
1Ø41 IF CA<>43 THEN IF CA<>1Ø7 T
HEN CX=1
1Ø42 IF CB<>43 THEN IF CB<>1Ø7 T
HEN CY=1
1Ø43 IF CX=1 THEN V=V+22
1Ø44 IF CY=1 THEN V=V-448:GOTO1Ø
Ø2 ELSE 1ØØ2
1Ø45 V=V-448
1Ø46 GOTO 1ØØ2
1Ø5Ø CA=PEEK(ADDR+1):CB=PEEK(ADD
R+32)
```

Listing continued