32K RAM FOR FREE!!!

By Frank Hogg

"How to run Pascal, C, and Cobal, not to mention X-FORTH, and Spelltest, on the TRS-80 COLOR COMPUTER"

Someday, as the Honeywell advertisement would say, integrated circuit processing will become so inexpensive that computer memory will be available for FREE.

That day is today, for owners of the Radio Shack TRS-80 COLOR COMPUTER.

The story begins with my early production model (with a 3-digit serial number) of the 4k COLOR COMPUTER. Its logic board had some extra wires and things on it, indicating that the design was not quite perfected when it was produced. I heard that Radio Shack would replace the board with a newer version if I purchased their 32k ram upgrade for $149.00, so I decided to give it a try.

When I took the computer to the local computer center, I was told that the upgrade would only cost $99.00. They did complete the upgrade, and indeed they did install a new logic board, containing eight memory chips with unrecognizable part numbers on them.

Various rumors have been circulating about how the 32k upgrade is accomplished. It is not done by piggybacking 16k rams! Neither is it done by installing 32k rams, as Radio Shack contends.

The 32k dynamic ram was only available for a short time. These parts were actually attempts at 64k parts that were only half-good, or they had some bad bits in one half or the other. The 32k upgrade was originally designed to take advantage of these parts - a jumper exists on revision E of the COLOR COMPUTER circuit board to select which half of the 64k dynamic ram is accessed.

Since then, memory manufacturers have learned how to produce 64k chips with sufficient yield to drive the cost lower than you or I, or Radio Shack, expected. These chips are available by mail order, in small quantities, for less than $12.00 each. Radio Shack can certainly buy them in quantity at a lower price.

The astute reader will have guessed the punch line by now. The 32k COLOR COMPUTER actually contains 64k rams! I am not in a position to guarantee this, of course, but so far it seems to be the case. I will now tell you how the "other 32K" might be useful to you.

USING THE FULL 64K RAM.

None of the versions of Radio Shack Color Basic know how to use the other 32k. As a matter of fact, this memory is not available to the cpu at all in an unmodified COLOR COMPUTER. This is due to an easily correctible omission in the design of the computer.

The dynamic memory in the COLOR COMPUTER is controlled by a chip known as the SAM, or synchronous address multiplexer. The SAM bears the Motorola part number 6883, or 74LS783. The SAM takes care of refreshing the rams and interlaces the access cycles of the cpu and the video display so that no "specks" occur on the screen. The SAM must be programmed differently for 4k, and 16k, and 64k rams. (this is why Color Basic 1.1 was written - version 1.0 didn't know about 64ks.) The SAM also provides address decoding for the three roms, as well as the I/O hardware.

As the SAM was being designed, Motorola considered the possibility that it might be useful in systems which did not use rom, but might want to use 64k of ram (minus 256 bytes for I/O, etc.) For this reason, the selection of rom in the SAM is programmable. If you whisper the right thing to the SAM (POKE &HFFDF,anything), the roms will go away, at least in theory, leaving behind nearly 32k of clean, untouhed ram.

Well, we need a more sophisticated theory, because it doesn't quite work. The SAM will still try to select the roms if the cpu writes to those addresses, regardless of how it is programmed. I guess Motorola must have thought that this decoding might be used for something - clearly it wouldn't hurt, since the system designer would have to provide logic to prevent the roms from being turned on in a write cycle anyway. (the rams are "selected" for write purposes all the time.)

Radio Shack, on the other hand, didn't see things the same way; they figured they would avoid writing to that area, so no problems would result. As a matter of fact, the first thing Color Basic does (after programming the SAM) is to test the memory from zero until it finds a byte that won't write. When this test hits address &H8000, the cpu tries to write the roms with exactly the opposite data they contain, and at the same time the roms are reading - resulting in two different chips trying to put different data onto the same bus at the same time.

The real tragedy is that a few unused NOR gates exist on the COLOR COMPUTER circuit board. You only need one of these to solve this problem. (Radio Shack designers - take note.)

THE MODIFICATION IS REVERSIBLE.

One of the extra NOR gates must be connected into the circuit as shown in figure 1. This modification disables the selection circuitry (G2B high) if a write is attempted (r/w low) and a rom is addressed (r/w low). If you have some experience with fine soldering, you can accomplish this modification in a reversible fashion, allowing you to run to Radio Shack if your COLOR COMPUTER breaks. Warning - you must remove that nasty sticker on the back, thus voiding your warranty (if you're still covered), to get inside.

The procedure is as follows. Remove the case and the top of the rf shield. On the right behind the keyboard. you should be able to find the IC's and TP1 as shown in figure 2. They are also marked on the board. U11 is a 74LS138, and U29 is a 74LS02.

You may wish to obtain a new 74LS138 and a 74LS02, so you can save the "originals" for a rainy day. In reality, Radio Shack probably doesn't remember what brand of IC it put in your computer, but precautions are cheap. Anyway, carefully remove those two IC's. (they are not especially sensitive to static.) Bend pins 4, 5 and 6 of the 74LS02 up in the air, as shown in figure 3. They must be almost straight up so they don't touch the shield. Similiarly disfigure pin 5 of the 74LS138. (be gentle!)

Next, using a short piece of 30-gauge wire, connect pin 6 of the 74LS02 to pin 8. Pin 8 must plug back in, so try not to get solder down on the pin. You should tack the wire on the very top of the pin, where it enters the package. If it doesn't come out right, buy another 74LS02 - it costs much less than a new computer.

You can do the rest of your soldering either before or after you plug the chips back in - use your own judgement. Pin 4 of the 74LS02 must be connected to pin 5 of the 74LS138, and pin 5 of the 74LS02 must be connected to TP1. I recommend that you do not solder to TP1. Just use a wire wrap tool to wrap the wire around the pin, so it can be pulled off.

After you have reinstalled the IC's, the wiring should appear as in figure 4. Check carefully for shorts!

At this point, you can turn on the computer and do a "PRINT MEM." If it says the usual number, all is probably well, so put it back together.

TESTING YOUR NEW FREE MEMORY.

The extended Color Basic program in listing 1 will test the ram which you have just made available. Save it before you try to run it, because if you mistype one of those data statements, anything can happen. The program will take about a minute to get set up, after which it will print "OK" if your memory is good. If you do have a problem, it will tell you the address and the data read from the ram, compared to what was expected. I would like to hear from you if you do find errors. If the errors occur in only one or two bit positions, they can be fixed with one or two 64k rams, for one or two ten-dollar bills. No big deal.

WHAT DO YOU DO WITH IT?

You now own a computer with almost 88k of memory, in a box no bigger than a typewriter. This fact alone may be enough for some of you. However, a large collection of software exists which can now be run on your computer.

The most important item in this collection is the popular FLEX operating system. (FLEX is a trademark of Technical Systems Consultants, Inc.) Frank Hogg Labs has developed a package which will allow FLEX to be run on the 32k COLOR COMPUTER, with the Radio Shack disc system, and the modification described above. FLEX will reside in memory at addresses &HC000-&HDfff, as always. addresses 0-&HBFFf will be available for user programs. All FLEX compatible software will run as is without patches or modifications. Addresses &HE000-&HFFEF will be available for utility programs. (We are working on an enhanced display package, using hi-res graphics to simulate a 41-by-24 screen. That's better than an apple!)

With FLEX you have a whole cosmos of software available to you. Besides the items mentioned in the subtitle, there are Basic compilers, business programs, adventure games, assemblers and text editors, word processing software, machine-language debug programs, disc

system diagnostic packages, and too much more to mention. FLEX is an excellent system which is widely supported.

If this isn't enough, we are planning to have the powerful OS-9 system on the Color Computer in the same way by early summer. Besides supporting several high level languages like Basic09, Pascal, C, and Cobal, OS-9 is Multi-User, Multi-Tasking, and Multi programming as well. The CC and OS-9 will be more powerful than all of the other TRS-80's put together!

SUMMARY

The 32k upgrade of the Radio Shack TRS-80 COLOR COMPUTER is accomplished by installing 64k dynamic ram chips. With a simple, reversible modification, nearly all 64k of this ram can be utilized. A package has been developed which will allow the FLEX operating system to be run on the modified 32k TRS-80CC with disc. You can do a lot of stuff with that.

In addition, the OS-9 operating system is being installed now and when finished (early summer '82') will allow the use of Radio Shack disk software, FLEX software, and OS-9 software on the same computer by merely changing Disks. No other computer made has those capabilities!

This article was prepared, using a preliminary version of the FLEX package, on a COLOR COMPUTER.

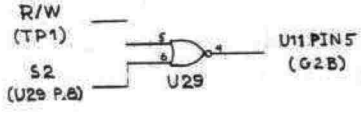


FIGURE 1. MODIFICATION TO TRS80CC FOR 64K RAM

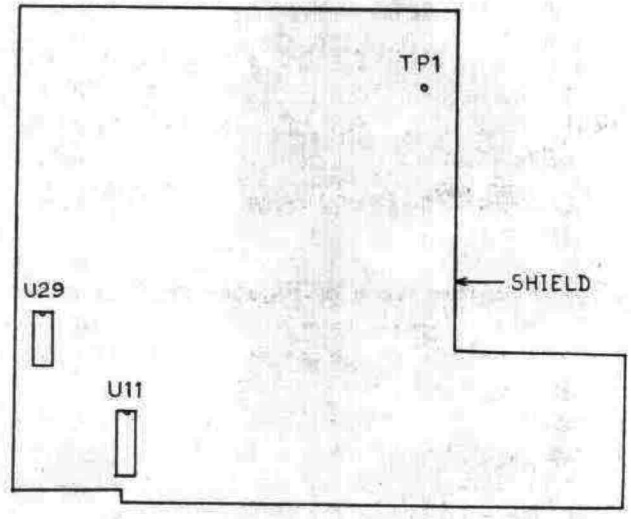


FIGURE 2. LOCATION OF COMPONENTS

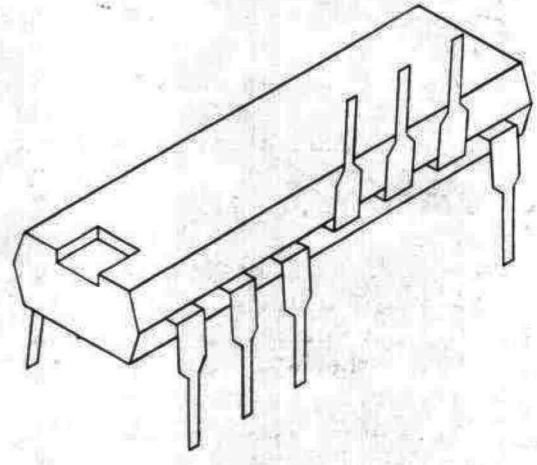


FIGURE 3. MODIFIED 74LS02 PACKAGE

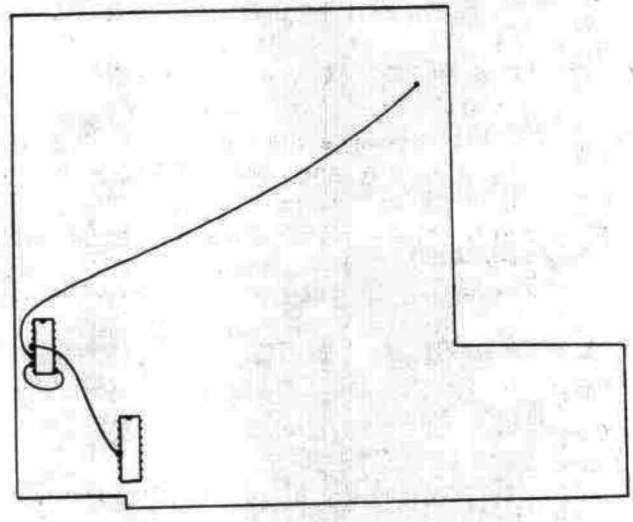


FIGURE 4. INSTALLED MODIFICATION

```
100 ' THIS PROGRAM TESTS THE
110 ' MEMORY CHIPS IN A MODIFIED
120 ' 32K TRS80 COLOR COMPUTER
130 ' FOR FUNCTIONALITY. THE
140 ' TEST DETERMINES WHETHER
150 ' FULL 64K RAM CHIPS WERE
160 ' USED IN THE RAM UPGRADE.
170 '
180 ' REQUIRES EXTENDED COLOR
190 ' BASIC.
200 '
210 ' REQUIRES ABOUT 65 SECONDS
220 ' TO SET UP THE MACHINE
230 ' LANGUAGE PROGRAM BEFORE
240 ' RUNNING.
250 '
260 ' RESERVE RAM
270 '
280 CLEAR 256,&H3FFB
290 '
300 ' SET UP MACHINE LANGUAGE
310 ' PROGRAM
320 '
330 GOSUB 1660 : SA=H
340 GOSUB 1660 : EA=H
350 GOSUB 1660 : E0=H
360 GOSUB 1660 : E1=H
370 '
380 ' SA=START ADDRESS
390 ' EA=END ADDRESS
400 ' E0=ENTRY POINT 0
410 ' E1=ENTRY POINT 1
420 '
430 FOR A=SA TO EA
440 GOSUB 1660
450 POKE A,H
460 NEXT A
470 '
480 DEFUSR0=E0
490 DEFUSR1=E1
500 '
510 ' TEST THE MEMORY
520 '
530 X=USR0(0)
540 '
550 ' CHECK FAILURE ADDRESS
560 '
570 FA=PEEK(&H3FFC)*256
    +PEEK(&H3FFD)
580 '
590 IF FA=0 THEN 760 :' END TEST
600 '
610 ' REPORT FAILURE
620 '
630 DW=PEEK(&H3FFE) :' DATA WRIT
640 DR=PEEK(&H3FFF) :' DATA READ
650 '
660 PRINT "AT ADDRESS ";HEX$(FA)
670 PRINT "WROTE ";HEX$(DW);
    "; READ ";HEX$(DR)
680 '
690 ' RESUME TEST
700 '
710 X=USR1(0)
720 '
730 GOTO 550
740 '
750 '
760 ' END OF TEST
770 '
780 END
790 '
800 ' MACHINE LANGUAGE PROGRAM
810 '
820 DATA 4000,4054
830 DATA 4000,4046
840 '
850 ' THE MACHINE LANGUAGE
860 ' MEMORY TEST PROGRAM IS AS
870 ' FOLLOWS:
880 '
890 ' E0   ORCC #$50 DIS. INTS.
900 '      STA $FFDF MAP TYPE 1
910 '      LDX #$4100
920 ' P1   CLR ,X+ CLEAR TO ZERO
930 '      CMPX #$FF00
940 '      BNE P1
950 '      CLR $3FFE DW
960 '      LDX #$4100
970 ' P2   LDA ,X
980 '      BNE RE REPORT ERROR
990 ' P2A  LDA #$FF
1000 '     STA ,X+
1010 '     CMPX #$FF00
1020 '     BNE P2
1030 '     COM $3FFE DW
1040 '     LDX #$4100
1050 'P3   LDA ,X
1060 '     CMPA #$FF
1070 '     BNE RE REPORT ERROR
1080 'P3A  LDA #$00
1090 '     STA ,X+
1100 '     CMPX #$FF00
1110 '     BNE P3
1120 '     LDX #$0000 END TEST
1130 'RE   STX $3FFC FA
1140 '     STA $3FFF DR
1150 '     STA $FFDE MAP TYPE 0
1160 '     ANDCC #$AF ENBL INTS.
1170 '     RTS
1180 '
1190 ' RESUME TESTING
1200 '
1210 'E1   ORCC #$50 DIS. INTS.
1220 '     STA $FFDF MAP TYPE 1
1230 '     LDX $3FFC FA
1240 '     TST $3FFE DW
1250 '     BEQ P2A
1260 '     BRA P3A
1270 '
1280 ' ACTUAL CODE
1290 '
1300 DATA 1A,50
1310 DATA B7,FF,DF
1320 DATA 8E,00,00
1340 DATA 8C,FF,00
1350 DATA 26,F9
1360 DATA 7F,3F,FE
1370 DATA 8E,41,00
1380 DATA A6,84
1390 DATA 26,21
1400 DATA 86,FF
1410 DATA A7,80
1420 DATA 8C,FF,00
1430 DATA 26,F3
1440 DATA 73,3F,FE
1450 DATA 8E,41,00
1460 DATA A6,84
1470 DATA 81,FF
1480 DATA 26,0C
1490 DATA 86,00
1500 DATA A7,80
1510 DATA 8C,FF,00
1520 DATA 26,F1
1530 DATA 8E,00,00
1540 DATA BF,3F,FC
1550 DATA B7,3F,FF
1560 DATA B7,FF,DE
1570 DATA 1C,AF
1580 DATA 39
1590 '
1600 DATA 1A,50
1610 DATA B7,FF,DF
1620 DATA BE,3F,FC
1630 DATA 7D,3F,FE
1640 DATA 27,C6
1650 DATA 20,D9
1660 '
1670 ' READ A HEX NUMBER TO H
1680 '
1690 READ A$
1700 LZ=LEN(A$)
1710 H=0
1720 IF LZ<=0 THEN RETURN
1730 C$=LEFT$(A$,1)
1740 FOR I=0 TO 15
1750 IF (I=0)AND(C$="0")
     THEN 1790
1760 IF C$=HEX$(I) THEN 1790
1770 NEXT I
1780 RETURN
1790 H=H*16+I:LZ=LZ-1
1800 A$=RIGHT$(A$,LZ)
1810 GOTO 1730
```