Custom Color

Give your MC-10 something to think about.

by Dennis Kitsz

WHAT? A DINKY LITTLE piece of computer junk from Radio Shack? Who cares? You might care, when you find out:

- It costs about $120 (less from some dealers, and expect occasional Radio Shack sales) with 4K of static RAM (about 3.2K, since the video calls for 512 bytes, and the system another 200 or so).
- It has a 6803 processor (one of the so-called "second generation" microprocessors), almost as powerful as the Color Computer's third-generation 6809. The 6803—part of the 6801 family—uses direct port inputs to the CPU.
- The keyboard uses these direct port inputs to the 6803, a nice touch.
- The unit is crystal-controlled from a 3.58MHz crystal in the RF modulator, which also provides the necessary clocking for the color video.
- The 6847 video display generator (VDG) is identical to the Color Computer's, meaning the characters per line (32) and lines per screen (16) are the same format and have the same 5x7 characters and low-density color graphics. The high-resolution graphics modes also work (more later).
- The MC-10 uses the same RS-232 printer and cassette ports as the Color Computer, and the same cassette and printer speeds and formats as well.
- Its Microcolor Basic is virtually identical to Color Basic. Following the Sinclair and VIC-20 leads, key words and some graphics can be typed directly from the keyboard. For me, however, the substitution of a control key for the left shift key is disturbing.
- It has a 34-pin expansion port for future who-knows-what. There is, however, no provision for internal Basic or RAM expansion, though I have heard some Basic ROMs are in sockets. If the computer becomes popular, I would imagine some outboard RAM would appear, or Radio Shack would do some piggy-backing of memories. If you're impatient about it, stay with me; secrets revealed a little later.
- There are at least five undocumented commands, four of which deal with machine language programs: CLOADM, EXEC, VARPTR, USR and OFF. The first three work just fine; USR apparently needs some direction (there is no DEFUSR to help it along), and OFF strikes me as a leftover from Color Basic's AUDIO and MOTOR commands. Want to see the keywords and lots of other stuff? Run this:

```
10 FOR X = 49152 TO 65535
20 PRINT CHR$(PEEK(X)AND127OR32);
30 NEXT
```

- Also, the instruction booklet says POKE will only work with video memory. Wrong: it will work anywhere. Want to change the cursor blinking character? POKE 17026,201... just for one example.
- The MC-10 won't load Color Computer programs directly, since the keywords are "tokenized" quite differently. But never fear; the Color Computer's "A" option CSAVE can be used if the program isn't too long in ASCII form (?OM errors abound in my tests). Some POKEing around is still needed to get line information straight, but it looks to be valid. (For another method, see Bill Barden's rundown of the MC-10 this month).
- Interference is remarkably low, even with a TV resting on the back of the computer. Heavy internal shielding is the lesson Radio Shack learned well from the Model 1, and, in my experience, they have learned it better than any manufacturer.

Listen, this is no one's dream computer, especially for $100, but I'll take it just for parts! I have always liked Microsoft's implementation of Basic for Tandy, and I like it once again here. When Radio Shack says that the MC-10 is part of their "Color Computer family," it is true: Basic, processor, video display and style.

Tear It Up

Even though the MC-10 is intended as a minimal computer, the 4K of memory is not satisfactory. Besides, I suspected that the high-resolution graphics modes could be used, but the highest resolution (256 x 192) requires 6144 bytes of memory, 2K more than the machine is designed with. Why not bring it up to 8K?

Please turn to page 80

Continued from page 78

I thought it would be an easy task, until I ran smack into the ghost—Radio Shack had used the phantom memory technique again! (Refer to "Custom Color," April, 1983, for more on phantom addressing.) In fact, memory was so incompletely decoded that there was already a change inside my MC-10 to correct an unanticipated flaw.

This analysis is just backwards engineering, mind you, but here is what it looks like to me. Judging from the unmodified circuit board, RAM was phantomed every 4K address block. Figuring that the MC-10 wasn't really being sold as an expandable machine, the hardware designers probably considered that saving an electronic part or two would be a good move. This decision resulted in memory not only appearing at $4000 to $4FFF, but also seeming to appear at $5000 to $5FFF, $6000 to $6FFF, and $7000 to $7FFF. Have a look at Figure 1, the original decoding scheme. Notice the output of the memory demultiplexer are binary 00, 01, 10, and 11. The 01 line alone was used to decode the memory, and 01 represents the most-significant two bits of $4000 to $7FFF.

Enter Microsoft, author of Tandy's Basic. Microsoft's power-up memory test technique is simple, fast and reliable.

Here's how it works: It starts at the first memory location to be tested, reads the binary information from that memory location, and saves it in the CPU. Next it complements that binary data (changing all 1's to 0's and all 0's to 1's), and stores that information in the memory location being tested. It reads it back. If it reads back exactly what it just stored, then that memory location passes the test. It finally restores the original value to memory, leaving it as found.

In case you missed what makes that a valid memory test, recall that the original number living in that memory location was read, saved, complemented, stored, read back, and re-stored. In order to pass the test, every bit of that memory location must be capable of storing a 1 or a 0 (the original value) or its complement, a 0 or a 1. That covers all the possibilities... and defines good memory.

Phantom addressing makes this scheme fail. The Microsoft memory test does not have a larger perspective, so to speak; once it has read a location, it goes on to the next one. If the address it just read was, say, $4FFF (the end of the MC-10's 4K of memory) and the next one is out of range at $5000, what happens? $5000 still tests all right because it is a functional phantom of $4000. It isn't "real," but Microsoft's test believes it is!

Please turn to page 83

MC-10 Color Computer — Parts Definition

Symbol	**Device**	**Function**
U1	MC6803P	CPU
U2	74LS373	Address/Data Latch
U3	B364	8K x 8 BASIC ROM
U4	74LS155	Memory Decoder
U5	74LS367	Address Buffer
U6	74LS367	Address Buffer
U7	74LS245	Data Buffer
U8	74LS174	General Latch
U9	D4016	2K x 8 RAM
U10	D4016	2K x 8 RAM
U11	MC6847	VDG
U12	74LS14	General Inverter
U13	74LS32	General OR Gate
U14	MC14503	Sound Output
U15	LM339	Cassette Input
U16	HA17741	
U17	7805	5-Volt Regulator
U18	74LS74	Master Clock
U19	74LS76	General Flip-Flop
Inside RF modulator unit:		
—	MC1372	Video Modulator

Table 1. Parts list and definitions of the MC-10; clock is generated inside RF modulator.

$0000–$00FF	6803 internal RAM
$0100–$3FFF	Not used
$4000–$41FF	Screen RAM
$4200–$4FFF	System and user RAM
$5000–$5FFF	Not used
$6000–$6FFF	Phantom of RAM
$7000–$7FFF	Not used
$8000	VDG and sound latch
$8001–$BFFF	4095 latch phantoms
$C000–$DFFF	BASIC ROM
$E000–$FFEF	Phantom of ROM
$FFF0–$FFFF	System vectors

Table 2. Generalized memory map of the MC-10. Areas not used as phantomed can be internally modified for expansion.

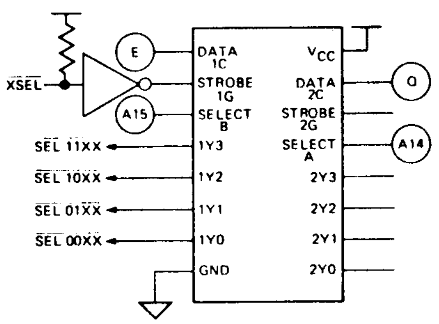
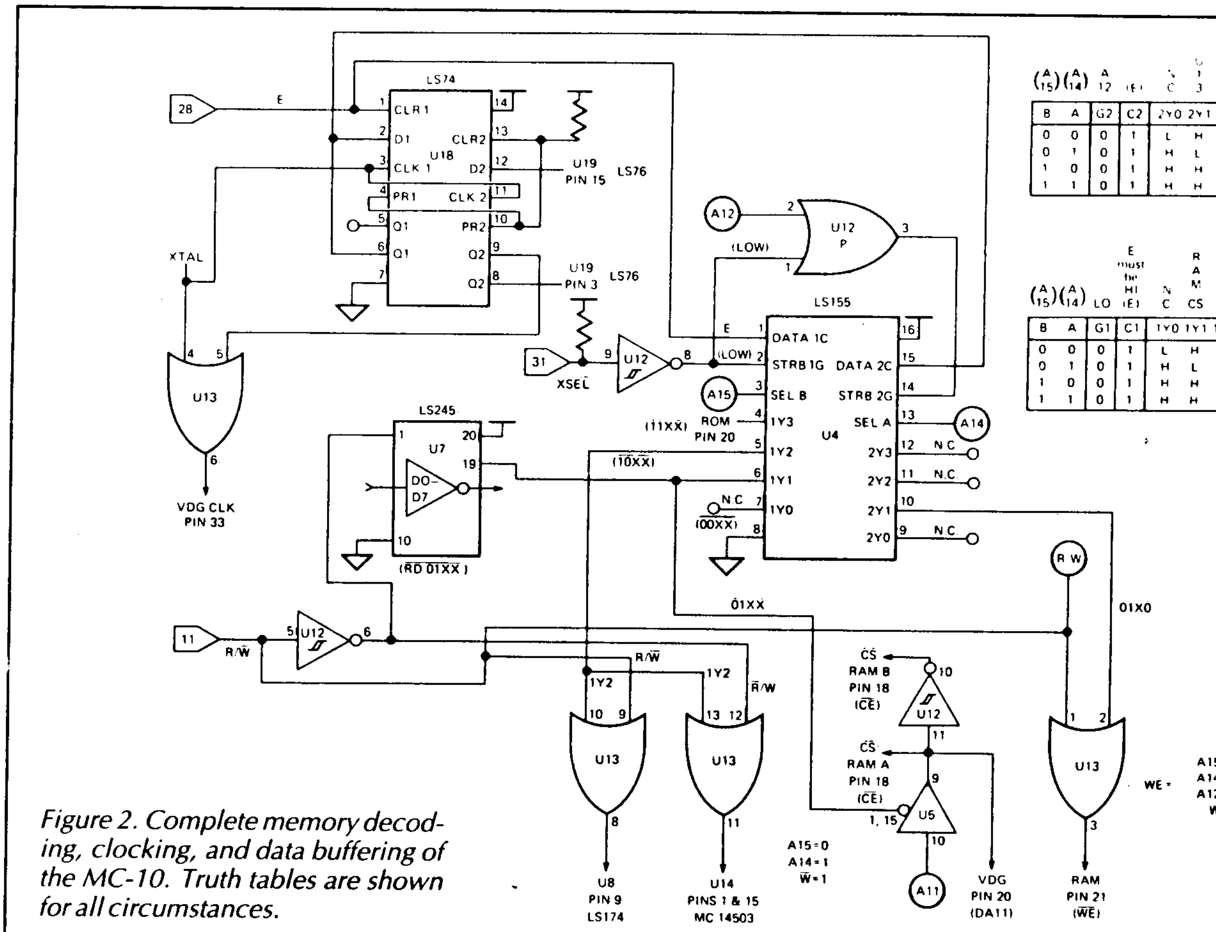


Figure 1. On-board memory decoding of the MC-10, before the Radio Shack correction.

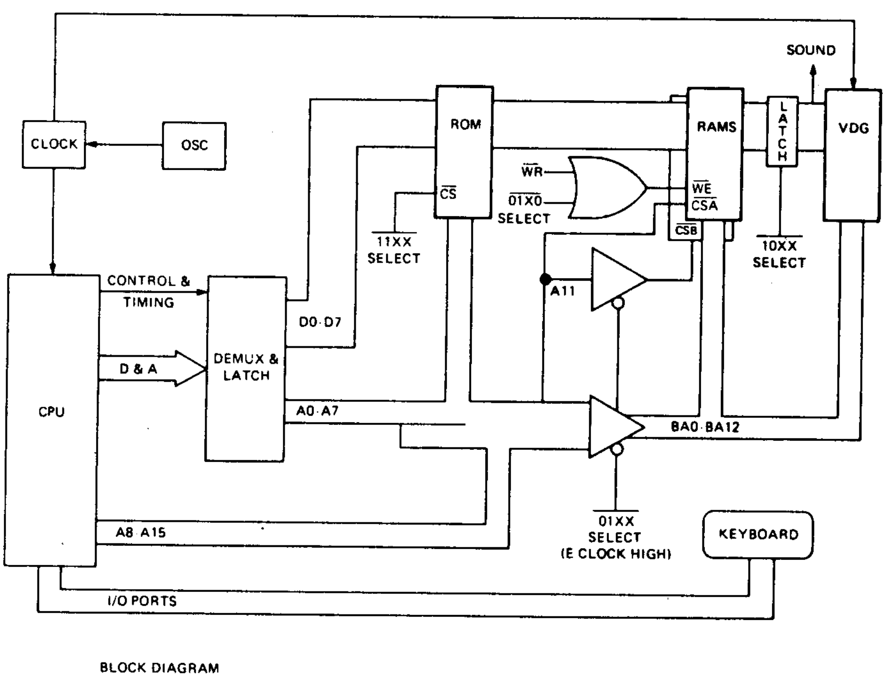


$\binom{A}{15}$	$\binom{A}{14}$	A 12	(E)	N C	U 13	N C	N C
B	A	G2	C2	2Y0	2Y1	2Y2	2Y3
0	0	0	1	L	H	H	H
0	1	0	1	H	L	H	H
1	0	0	1	H	H	L	H
1	1	0	1	H	H	H	L

$\binom{A}{15}$	$\binom{A}{14}$	LO	E must be HI (E)	N C	RAM CS	RAM U 8	ROM 20
B	A	G1	C1	1Y0	1Y1	1Y2	1Y3
0	0	0	1	L	H	H	H
0	1	0	1	H	L	H	H
1	0	0	1	H	H	L	H
1	1	0	1	H	H	H	L

Figure 2. Complete memory decoding, clocking, and data buffering of the MC-10. Truth tables are shown for all circumstances.

Figure 3. Block diagram of the MC-10. Keyboard directly attaches to the CPU, and VDG performs direct memory access (DMA) on the RAMs.



Continued from page 80

Incomplete addressing results in 16K of memory being reported for use by Basic.

So, inside my MC-10 is a piggybacked chip which includes address line 12 in the scheme. (The complete parts list is shown in Table 1.) It takes the 01XX decoding and refines it to 01X0 (X means unused). Memory is still phantomed, but only in two blocks: \$4000 to \$4FFF, and \$6000 to \$6FFF, both of which have binary most-significant-nybbles of 01X0. Memory is deselected (turned off) when the address changes to \$5000 or \$7000, since their most-significant-nybble is 01X1. The simple memory test finds no memory at \$5000, and reports the proper 4K value to Basic.

This incomplete decoding has its uses—to add 4K more of memory, for example. But first, an outline of the memory map is in order (See Table 2).

The place to put extra RAM is in that \$5000 to \$5FFF bank. Figure 3 is a block diagram of the MC-10 system. The CPU's lower address and data lines are combined, and must be demultiplexed via timing and control signals. Data lines feed ROMs and RAMs, and are latched to select the VDG mode and sound output. Address lines run directly to the ROM and

Please turn the page

also to a memory decoder, which provides the four memory block selects I have been describing. These select ROM (11XX), VDG (10XX) or RAM (01XX). Since the RAMs must be shared by the CPU and the VDG, a buffer is inserted on the address bus. In this way, the VDG can put its display address on the bus when the CPU is not using the memory. Since they share clock timing, this rudimentary direct memory access (DMA) process is simplified.

Understanding the DMA as well as the memory decoding is necessary to add 4K to the MC-10. Figure 4 is a detailed schematic of the address bus leading to RAM and VDG. In order to address the extra memory, address line A12 must be included in the decoding. It must also be buffered so the VDG line DA12 can access the area it needs without producing its own phantom addressing! The 12 address lines A0 — A11 use all 12 buffers of a pair of 74LS367 integrated circuits; there are no "spares" on board the MC-10. Task 1: Add a buffer for A12.

Next, the decoding of memory blocks must be improved for two reasons: to access two extra 2K RAMs, and to avoid (once again) phantom addressing that will cause the power-up memory test to be fooled. Task 2: Provide two additional chip selects for 2K RAMs.

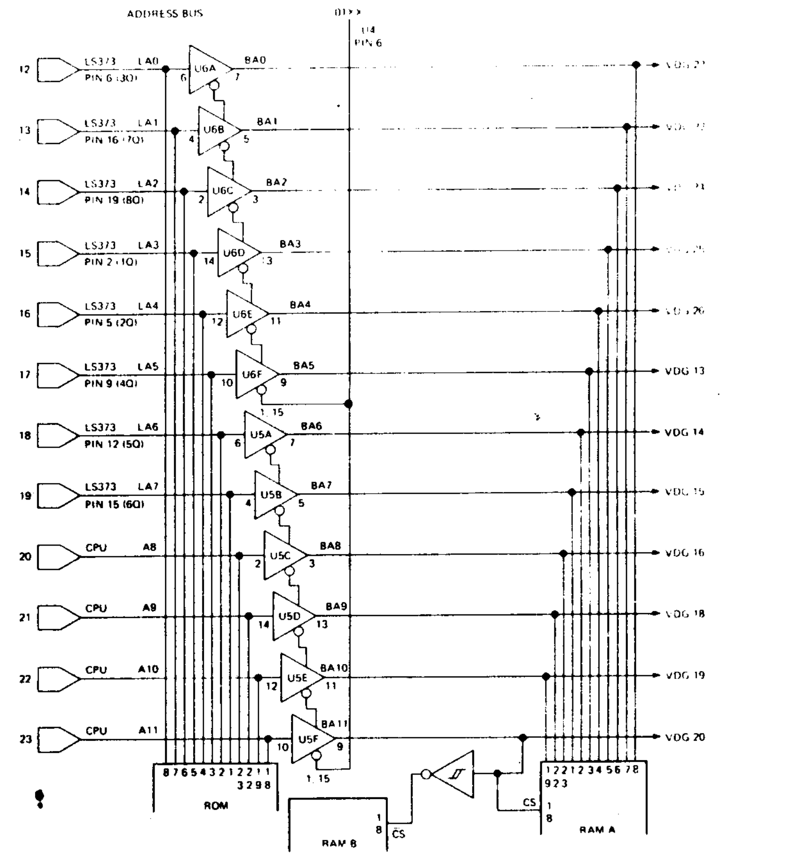


Figure 4. Buffered address bus of the MC-10 is shared by the CPU and the 6847 VDG, depending on the state of the E clock.

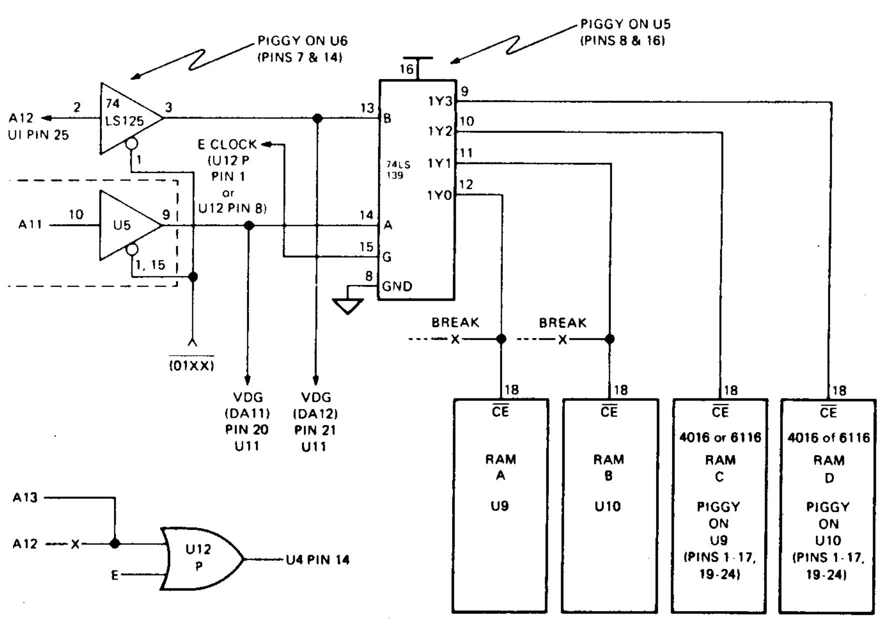


Figure 5. 8K memory upgrade schematic for the MC-10. Simple gate addressing is replaced by 74LS139 demultiplexer.

The result is the schematic in Figure 5. An additional buffer is provided for line A12 in the form of a 74LS125, and the memory selection is performed by a 74LS139 demultiplexer activated by the E clock, and decoding A11 and A12 into four 2K blocks. However, recall that there is an on-board modification (by the time you read this, the board may be revised—look carefully) to prevent access at $5000 to $5FFF, so that the memory test may be correctly performed. By moving this connection from A12 to A13, all the memory from $4000 to $5FFF can be accessed, while a new memory denial at $6000 to $7FFF is created.

Inside the MC-10

Disconnect the power and video cables. Four screws hold the case bottom on the computer; three are visible, and the last is underneath the inevitable warning label. The case snaps together at the sides with plastic tabs, so after the screws are removed, grasp it at the back and split it firmly. Do not pull the unit open: There is a keyboard cable inside. Once the unit is split, rock the keyboard forward, and you will see the insides, as shown in Photo 1.

Please turn the page

Continued from page 85

The keyboard cables come out first. Set the open computer in your lap, rest the keyboard forward against your body. Open the keyboard clamps by lifting them fully to the right. Hold the computer steady with one hand, and with the other grasp the whole width of the cable. Pull it straight up. Repeat this process with the other cable, and set the keyboard face down and well out of the way. Don't bend the cable ends or attempt to straighten the cables. The keyboardless unit will look like Photo 2.

There are three screws which hold down the circuit board, one at the back to the left of the video modulator case, one at the lower right, and one at the lower left, reached through a hole in the interference shield. Remove these; the computer will now lift out of the case. There may be several rubber supports glued into the case bottom.

Set the bottom aside, and turn the circuit board over. You will see a number of holes, a switch, and 11 rivets. Remove the rivets by pressing them out from above. There are a few left under the upper metal shield, so gently lift the lower flexible shield until the rivets come out. Be careful not to pull off the video switch when moving the lower shield.

There are eight solder points holding the upper shield in place. Heat these points from below with a 40-watt soldering iron, while lifting from above. Each point will give way with surprising ease. Work your way around; twice should do before the shield drops off. You will see the parts layout shown in Photo 3. Notice the piggybacked circuit on U12, and the three wires leading from it.

For the modification, you will need four integrated circuits: the 74LS125 and 74LS139 mentioned earlier, plus two 4016 memories. These masquerade under several numbers, including AM9128, D4016, etc. They are 2K x 8 static RAMs.

Bend all the pins of the 74LS139 up and parallel with its body except pins 8 and 16. Mount the 74LS139 on U5, with the notch pointing in the same direction, and solder power pins 8 and 16 to U5. Bend all the pins of the 74LS125 up and parallel with its body except pins 7 and 14. Bend these slightly so this 14-pin integrated circuit fits on U5, which is a 16-pin device. Mount it in the same direction as U5, and solder pin 7 to U5 pin 8, and pin 14 to U5 pin 16.

Now mount the two memories. Bend up only pin 18, and solder pins 1–17 and 19–24 to U10 and U9, respectively. Be very careful when soldering to the memories below—do not get them too hot, do not create solder bridges, and make sure all pins are soldered. Use a magnifying glass to verify all connections are clean and tight.

Now turn the board over. Locate the traces leading from pin 18 of each memory chip. Cut each trace through with a sharp tool; this removes the select lines from the memories. Return the board to the top side.

There are three wires leading from Radio Shack's corrective piggyback. Remove the wire leading to a plated-through hole at the lower edge of the board. Verify that this is the hole leading to the CPU (U1) pin 25. Move it to the CPU pin 24. Now route the following wires (this order is probably the easiest).

P means piggybacked IC:

1. U5 pin 1 to U6P pin 1.
2. U6P pin 2 to U1 pin 25.
3. U5P pin 15 to U12 pin 8.

Please turn to page 90

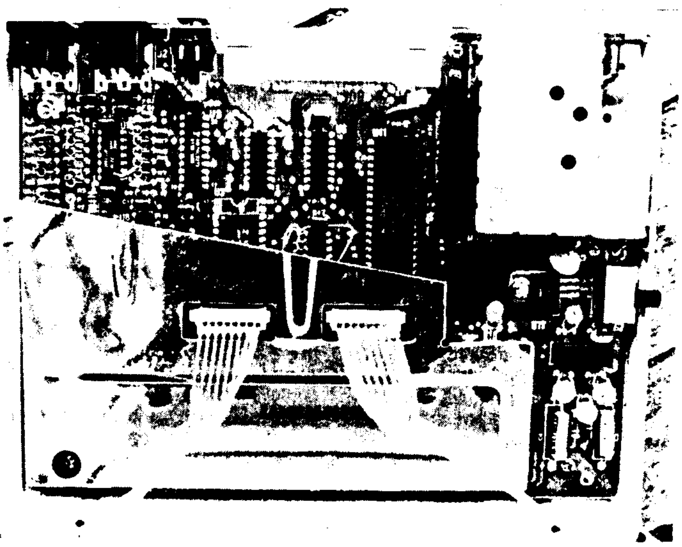


Photo 1. View inside the MC-10 as it is just opened. Clockwise from back: cassette/serial jacks, reset switch, expansion bus, RF modulator, on/off switch, power supply, RF interference shield, keyboard cables and connectors.

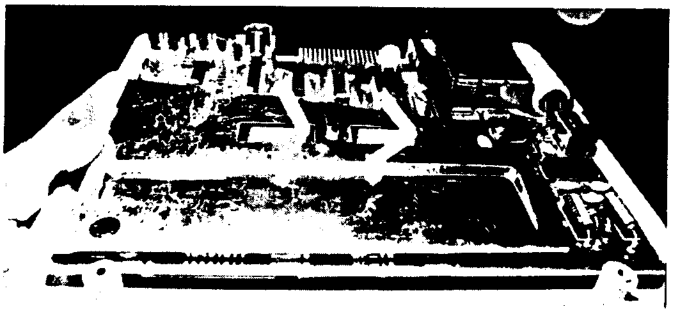


Photo 2. Keyboard cables lift out of snap connectors.

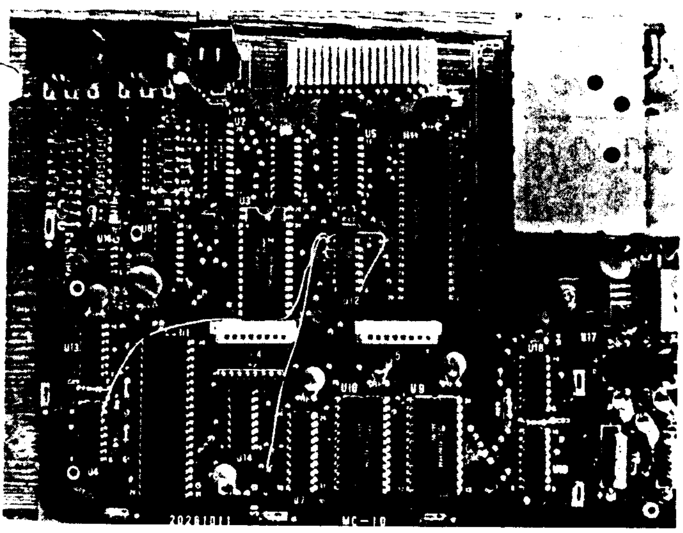


Photo 3. Underneath the shield, revealed 6803 CPU at front left; 8K Basic ROM in center; 6847 VDG to left of RF modulator; two D4016 RAMs up front. Note the Radio Shack piggyback in place.

Table 3. Table of Mode Control Lines (Inputs)

$\overline{A}$/G	$\overline{A}$/S	$\overline{INT}$/EXT	INV	GM2	GM1	GM0	ALPHA/GRAPHIC MODE SELECT
0	0	0	0	x	x	x	Internal Alphanumerics
0	0	0	1	x	x	x	Internal Alphanumerics Inverted
0	0	1	0	x	x	x	External Alphanumerics
0	0	1	1	x	x	x	External Alphanumerics Inverted
0	1	0	x	x	x	x	Semigraphics - 4
0	1	1	x	x	x	x	Semigraphics - 6
1	x	x	x	0	0	0	64 x 64 Color Graphics
1	x	x	x	0	0	1	128 x 64 Graphics
1	x	x	x	0	1	0	128 x 64 Color Graphics
1	x	x	x	0	1	1	128 x 96 Graphics
1	x	x	x	1	0	0	128 x 96 Color Graphics
1	x	x	x	1	0	1	128 x 192 Graphics
1	x	x	x	1	1	0	128 x 192 Color Graphics
1	x	x	x	1	1	1	256 x 192 Graphics

Table 4. VDG latch assignments through U8, latched by writing data to any address from $8000 to $BFFF.

D7	D6	D5	D4	D3	D2	D1	D0
Sound Output	Color Set	A*/G	GM0	GM1	GM2 & INT*/EXT	Not Used	Not Used

Text continued from page 86

4. U5P pin 14 to U12 pin 11.
5. U6P pin 3 to U5P pin 13.
6. U11 pin 20 to U12 pin 11.
7. U11 pin 21 to U5P pin 13.
8. U5P pin 12 to U9 pin 18.
9. U5P Pin 11 to U10 pin 18.
10. U5P pin 10 to U9P pin 18.
11. U5P pin 9 to U10P pin 18.

The modification is now complete. The resulting wiring is shown in Photo 4. Before testing, have a friend proofread the wiring with you. When you are sure it is correct, you are ready to test.

To test the computer, clear an area and set it down on a cloth. Insert the video and power cables and turn it on; your screen should clear red briefly (about 1/2 second), then return the start-up message on a green screen. If you see a flash of @ symbols before sign-on, or if there is no sign-on message, double check all your wiring.

Short of a bad memory chip or poor construction, there is little that can go wrong with this modification. The MC-10 is a simple machine, and its memory area is also stable, strong, and straightforward. It is hard to damage it.

The next test requires the keyboard. Remove power, and insert the keyboard cables in place; be sure not to bend the cable ends as you insert them. Restore power and PRINT MEM; you should get 7238. If you get 3142, something didn't take. If you get 11334 or 15430, don't get excited; that's not what you have. Check your wiring. If you get ?OM error, or if the machine locks up, make sure the 74LS139 pins are wired down. Finally, from command mode, enter this line: DIM A$(100,10). If you get an "OK" response, you win. Your upgrade is ready to work for you. If, however, the screen turns red and fills with identical odd characters, your wiring is bad in the piggybacking or shorted somewhere to one (or more) memory pin 18.

Oh yes: now put it all back together—the upper shield first, then the lower shield (all 11 rivets), the screws, the case. You're set; except for the label, it looks like it came off the shelf. PRINT MEM the next time you want to impress your users group!

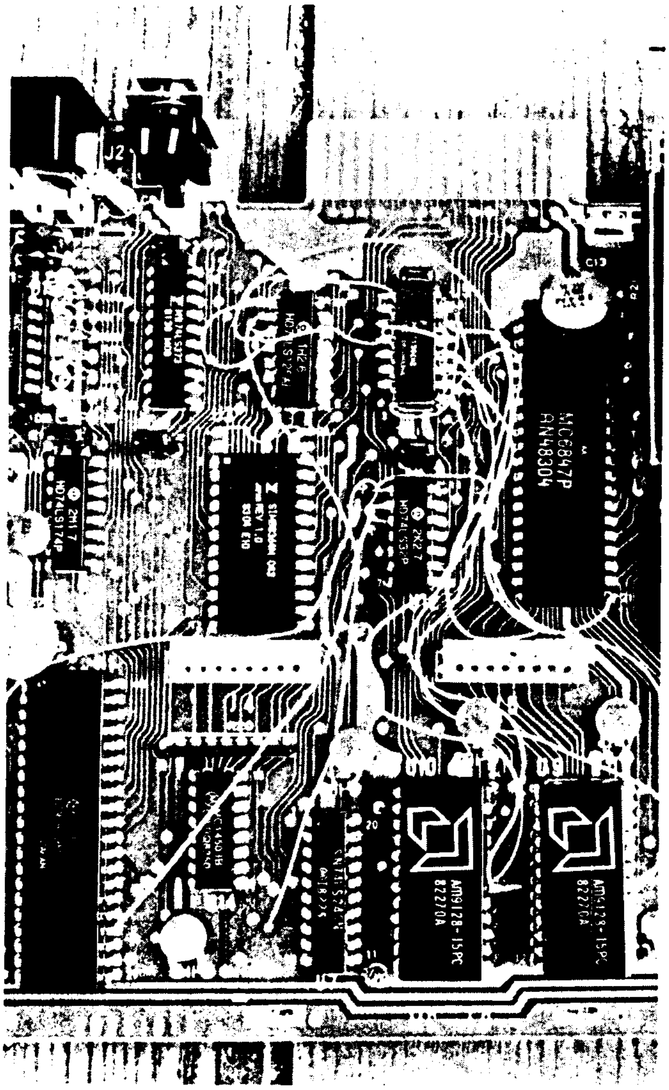


Photo 4. The 8K memory upgrade in place. Wiring is straight forward, although piggybacking two memory chips requires great care.

Parts List

Parts list for 4K upgrade:

U6P	74LS125	A12 buffer
U5P	74LS139	Memory selector
U9P	D4016	2Kx8 RAM
U10P	D4016	2Kx8 RAM

Complete set of four integrated circuits, wire and solder available for $25 from Green Mountain Micro, Roxbury, Vermont 05669.

Please turn the page

Continued from page 91

High-Resolution Graphics

Now that you have 8K of memory, you have access to all the high-resolution graphics modes. Just for a first try, do this: POKE 32768,255.

There. Excited? I was. Have a look at Figure 7; this is the schematic of the VDG. As noted earlier, the address bus is taken over by the VDG when it needs to access the display memory. The other item of interest in this schematic is the data latch, which not only selects the sound output, but also determines the VDG modes. These modes are summarized in Table 3.

GM2 and the INT*/EXT line are tied together, as are A*/G and D7, and INV and D6. This is an efficient connection scheme for the 256 data values, since only 64 characters are available inside the VDG:

D7	D6	Result
0	0	Normal alphanumerics
0	1	Reverse alphanumerics
1	0	Semigraphics–4
1	1	Semigraphics–4

VDG latch assignments through U8 are shown in Table 4. To go through the list of possibilities completely, try this:

```
10 INPUT"Speed";Q
20 FOR X = 0 TO 127 STEP 4
30 POKE 32768, X
40 FOR N = 1 TO Q : NEXT
50 NEXT
```

And finally, to try the sound output from Basic by directly manipulating the latch, do this:

```
10 X=32768:Y=128:Z=0
20 POKEX,Y:POKEX,Z:GOTO20
```

The MC-10 is a surprisingly versatile single-board computer. As a Basic computer, it looks like an interesting introductory machine. Since it can be expanded, and since features such as serial interface and reliable cassette storage are standard, I feel it can serve as a remarkable laboratory computer or process controller. This preview should get you started with some options and upgrades. Look in future "Custom Color" columns for regular tours through this excellent miniature personal computer.

■ ■ ■

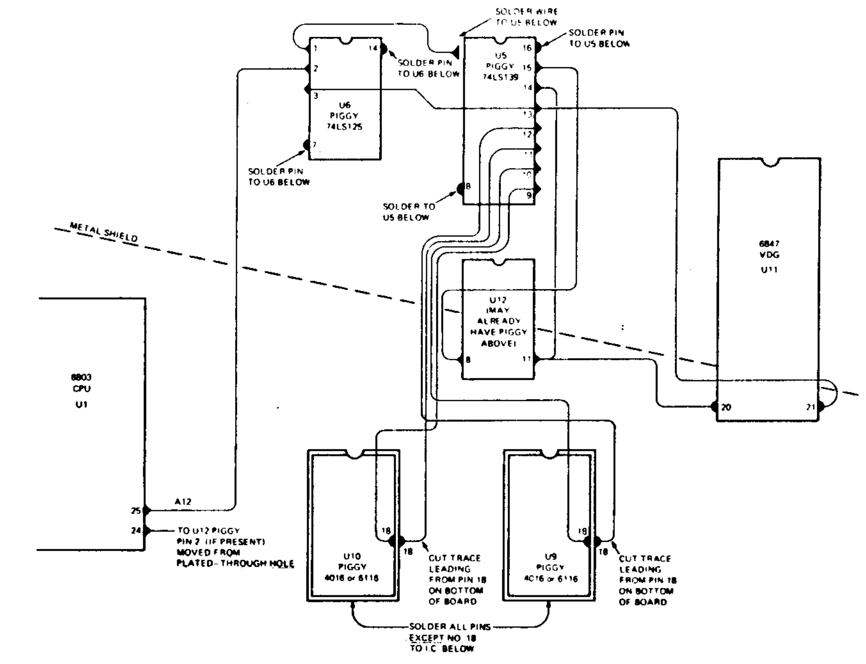


Figure 6. Wiring diagram of the 8K memory upgrade for the MC-10.

Figure 7. Access to the 6847 Video Display Generator, as well as to the sound output, is provided by a data latch address by any memory location from $8000 to $BFFF.