Hardware Modification TRS-80 Model I @Don Kelly

Replacing 4116 memory chips with 4164 memory chips.

Discussion:

The TRS-80 Model I uses a Z80 8bit processor capable of addressing up to 64K of main memory without memory management or bank switching. The Model I implements the lower 16K of memory as ROM containing BASIC and operating system primitives, and the memory mapping required for video, keyboard etc.

As built the Model I PCB (excluding the Expansion Interface) can house up to 16K of main memory. A 16K configuration consists of eight (8) 4116 memory chips at board locations Z13 to Z20 inclusive. When the EI is attached an additional two banks of 4116 memory chips can be added to bring the Model I to it's maximum of 48K of main memory.

There were a number of difficulties with Model I memory as implemented in the EI including CAS and/or RAS degradation, speed problems, reliability issues related to the length of the address and data busses and connector resistance along the interface cable from the Model I to the EI. At one point buffer chips were implemented in a special interconnect cable to help boost the TTL signal levels and compensate for the problem. I believe this was one way buffering.

As with other things in the Model I these problems were not universal, some units worked better than others, some had no problem at all.

This modification replaces the 4116 (16K) memory chips on the main Model I PCB with 4164 memory chips and has these advantages:

1 - With this modification Model I computers can have a full 48K of memory without the Expansion Interface.

2 - Memory buffering problems with the EI can be eliminated since there will be no need to install any memory chips in the EI.

3 - 4116 memory chips will become more and more rare, 4164 chips are more readily available.

4 - This modification will disable the memory's use of both the +12volt and -5volt power supplies. Since no other circuits use these supplies it becomes possible to operate the Model I with a single +5volt supply. This modification does not talk about how to implement a single 5volt supply.

There is also one disadvantage.

The extra load imposed on the +5volt supply by the 4164 chips may force use of an external 5V supply.

Though this modification assumes upgrading from a 16K Model I (4116 chips) the modification can be implemented on a 4K or 8K unit as well.

WARNING

If you intend to connect a modified Model I to an Expansion Interface you must **REMOVE ALL MEMORY CHIPS FROM THE EI.**

This is an extensive hardware modification to the TRS-80 Model I, if you are not 100% confident in your abilities to carry out delicate soldering activities please do not attempt this modification by yourself.

Differences in the 4116 and 4164

There are two main differences in the two chips:

1 - The 4116 uses Ground and three voltages +5V, -5V, and +12V, the 4164 uses only Ground and +5volts. Although the two chips do use the same pin for Ground (pin 16) they do not use the same pin for +5volts (the 4116 uses pin 9 and the 4164 uses pin 8).

2 - The 4116 uses 7 address lines while the 4164 uses 8. It is the use of the 8th address line that allows the 4164 to address 64K of memory instead of the 16K addressed by the 4116.

Though these sound like fairly simple differences they will require considerable work to accommodate on a circuit board like that used in the Model I.

The basic steps required to make this happen.

The last page shows the pin assignments for the 4116 and the 4164, use that page as a reference. Lets take a look at each of the 16 pins on the 4164, compare to the 4116 and note what has to happen.

Pin 1 - The 4164 does not use this pin, the 4116 uses it as it's -5volt connection. Since we will not require -5volts for the 4164 this pin needs to be disconnected and tied to ground. Tying unused pins to ground is just good practice with TTL technology.

Pin 2 - Both chips use this as their DATA-IN connection - no changes required.

Pin 3 - Both chips use this as their Write-Enable connection - no change required.

Pin 4 - Both chips use this pin as their ROW-ADDRESS-SELECT connection - no change required.

Pin 5 - Both chips use this pin as Address-0 - no change required.

Pin 6 - Both chips use this pin as Address -2 - no change required.

Pin 7 - Both chips use this pin as Address -1 - no change required.

Pin 8 - The 4164 uses this pin as +5 volts, the 4116 uses it as +12 volts, changes definitely required here.

Pin 9 - The 4164 uses this pin as Address-7 while the 4116 uses it as +5volts, changes definitely required.

Pin 10- Both chips use this pin as Address -5 - no change required.

Pin 11- Both chips use this pin as Address -4 - no change required.

Pin 12- Both chips use this pin as Address -3 - no change required.

Pin 13- Both chips use this pin as Address -6 - no change required.

Pin 14- Both chips use this pin as DATA_OUT - no change required.

Pin 15- Both chips use this pin as COLOM_ADDRESS_SELECT - no change required.

Pin 16- Both chips use this pin as Ground - no change required.

So from this analysis we can see that changes are required to only three Pin connections to convert from 4116 chips to 4164 chips. Lets look at each of these in more detail and then get into the step by step instructions.

Pin 1 - The 4164 does not use this pin, the 4116 uses it as it's -5volt connection. Since we will not require -5volts for the 4164 this pin needs to be disconnected and tied to ground. Tying unused pins to ground is just good practice with TTL technology.

This is conceptually the easiest pin to change. Typically when computer memory boards are constructed all similar pins of the memory chips are connected together, so all eight pin ones should be connected together in a string. All we really have to do is find where this string connects to the -5volt supply and cut the connection.

Pin 8 - The 4164 uses this pin as +5 volts, the 4116 uses it as +12 volts, changes definitely required here.

For pin 8 we have to find it's connection to the +12 volt supply and sever it and then connect the string of pin 8s to the 5 volt supply. Since 5volts is the "standard" TTL logic voltage this may represent many connections on a printed circuit card.

Pin 9 - The 4164 uses this pin as Address-7 while the 4116 uses it as +5volts, changes definitely required.

Like the other two pins these should already be strung together, we will have to confirm this, disconnect from the +5volt supply and connect to the processor's DATA-LINE-#7. Since we will be disconnecting from a power supply line we will need to look for filter and bypass capacitors on these pins and disconnect them as well.

Step By Step

Before you start you really should read through all of this to get an understanding of what is going to be required. As I said before this is not for the faint of heart, the good news is that I have already done the research for you.

I worked this modification out on a version "D" board and have implemented it on a version "G" board as well. If you have a different board version please review the steps below against your board before you start.

Materials

One low profile 14 pin chip socket. Solder (please visit Radio Shack and acquire the right kind of solder), a low wattage soldering iron with a small tip, some wire wrap wire (available at Radio Shack), an exacta knife (small one please) or perhaps a razor blade, good eyes and a steady hand. A copy of the Model I schematic and an Ohm meter or continuity tester would also be valuable assets.

FOR CLARITY ALL REFERENCES TO THE BOARD ASSUME THE EDGE WITH THE LARGE SILVER TRANSISTOR (2N6594) IS THE TOP OF THE BOARD (NORTH).

Implementing the changes for PIN - 1

The foil that joins pin-1 of all eight chips runs on the component side of the board and is quite a heavy trace. No changes are required to this trace. There are however four capacitors that need to be removed:

Remove the 8 4116 chips from their sockets and put them aside. Do not install the new chips at this point.

Step 1: Remove the following capacitors:

C -19 just above Z19
C -18 just above Z17
C -17 just above Z15
C -16 just above Z13

Step 2: The left leg of C - 16 connects through the board to the -5 volt line on the foil side of the board. Turn the board over while maintaining the top of the board away from you. Where Z 13 - pin 1 comes through you will see three heavy tracks on the board, cut the track on the right. For confirmation this is the track that travels to the right and connects to resistor R-19 and C-3. If you have an Ohm meter or continuity tester confirm the connection is broken.

Step 3: On the foil side of the board connect a short jumper wire from Z-13 pin 1 to Z-13 pin 16. This places a ground on pin 1 which is just good practice with TTL logic chips.

This completes what has to be done for pin-1, so now we move on to pin-8.

Implementing the changes for PIN - 8

Pin 8 - The 4164 uses this pin as +5 volts, the 4116 uses it as +12 volts, changes definitely required here.

Once again we find that the foil that joins pin-8 of all eight chips runs on the component side of the board and is a heavy trace easy to see.

Step 4: Z-19 Pin -8 on the foil side of the board has the connection to +12 volts. On the foil side of the board cut the foil connected to Z-19 pin 8.

Step 5: ***Connect pin 8 to +5 volts as part of step 6.***

Implementing the changes for PIN - 9

This is the pin requiring all of the work.

Step 6: Z20 pin-9. On the component side of the board cut the foil connecting to pin-9 just below pin-8. Reconnect the open foil (not the part going to pin-9) to pin-8, thus providing +5 volts to pin-8 of all the chips.

Step 7: Z19 pin-9. On the foil side of the board cut the foil connection to Z19 pin-9 by opening the short piece of foil between pin-9 and the vertical +5v foil. Be careful to avoid cutting the vertical foil.

Step 8: Z18 pin-9. On the foil side of the board cut the foil leading to Z18 pin-9 as close to pin-9 as possible.

Step 9: Z17 pin-9. On the foil side of the board cut the foil leading to Z17 pin-9 as close to pin-9 as possible.

Step 10: Z16 pin-9. On the foil side of the board cut the foil leading to Z16 pin-9 as close to pin-9 as possible.

Step 11: Z15 pin-9. On the foil side of the board cut the foil leading to Z15 pin-9 as close to pin-9 as possible.

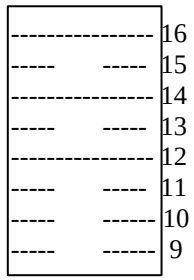
Step 12: Z14 pin-9. On the foil side of the board cut the foil leading to Z14 pin-9 between pin-9 and the lead of C-29.

Step 13: Z13 pin-9. On the foil side of the board cut the foil leading to Z13 pin-9 by severing the short foil leading from pin-9 to the adjacent vertical foil.

Step 14: Connect together pin-9 of each of the eight chip sockets using light wire, wire wrap wire is recommended. Solder with care and be sure you do not accidentally reconnect any of the +5volt connections with a solder bridge.

Implementing Address line 7 & Taking care of the memory jumper blocks.

Step 15: Establish the jumper block (components of X71) as required for the 16K memory implementation. That is:



Step 16: Implement Address line 7 by connecting a wire from Z17 pin-9 to Z51 pin-12.

Step 17: Implement RAM address decoding part 1 by connecting a wire from Z51 pin-13 to Address-15 at Z38 pin-9.

Step 18: Implement RAM address decoding part 2 by connecting a wire from Z51 pin 14 to Address-14 at Z38 pin-11.

This is the easiest way to implement the additional Address line encoding and allow all of RAM memory to be addressed. It also results in the 4164 chips being internally addressed in a non contiguous manner, this is of no consequence except that the wiring may appear illogical to the purists among us.

Step 19: Insert eight 4164 chips into the eight sockets at Z13 to Z20 inclusive be careful to orient the chips correctly. Integrated circuit chips have pin-1 identified by a small dot or notch on the chip. The mark on the 4164s you are installing should be toward the top of the board (pointing at the big silver transistor).

Now comes the system address decoding. This is presented as step by step instructions with an explanation of how I arrived at this at the end. To accomplish this part of the modification we require two gates, an inverter and a NOR gate. I used two gates to accomplish this simply because I didn't find a single gate that would do what is needed and I didn't wish to go to all the effort of implementing another chip onto the board. This works so what the hell.

I used Z9 pins 5&6 as my inverter and Z37 pins 8,9,10 as my NOR gate. On both the D and G boards that I worked on these gates were available, however various versions of the boards and other modifications may have used these gates. If this is the case on your board you will have to locate other gates to use and adjust these instructions accordingly.

I have done this modification twice once on a “D” revision board and once on a “G” board. On the “D” board Z37 pin-8 is connected to ground on the top of the board (component side) on the “G” board it was not. This procedure follows the careful way.

Preparing Z37 for use

Step 20: Remove Z37 from the board.

Step 21: Cut and remove the foil around Z37 pin-8 on the component side of the board. This foil may or may not be connected to ground (see note above).

Step 22: On the foil side of the board cut the foil joining Z37 pins 8 and 9.

Step 23: Confirm that no connection exists between Z37 pin 8 and Z37 pin 9 and that no connection exists between either of these pins and ground.

Step 24: Install a 14 Pin low profile chip socket at position Z37 and insert Z37 (74LS02) into the socket, check for bent pins.

Preparing Z9 for use

Step 25: On the foil side of the board cut the foil so that pin-5 is disconnected from the ground foil connecting pin-7 and pin-1. Be sure that the connection from pin-7 to pin-1 remains intact.

Final System address decoding changes

What is needed is to identify to the address decoding logic that any time address line 15 (A15) is high (one (1)) we are talking to RAM. The truth of this is explained later.

X3 may have been implemented as a block of 8 switches or as a block of jumpers which allow you to break certain jumpers and there by “program” the address decoder.

Step 26: On X3 open the switch or connection between the following pin combinations:

2 and 15
3 and 14
4 and 13
5 and 12

For clarity there should be NO CONNECTION between these sets if pins.

Step 27: On the foil side of the board connect a wire from Z9 pin-5 to X3 pins 2,3,4,5. For clarity pins 2,3,4,5 of X3 need to be connected together and then connected to Z9 pin-5.

Step 28: Connect a wire from Z9 pin-6 to Z37 pin-9.

Step 29: Connect a wire from Z38 pin-9 to Z37 pin-8.

Step 30: Connect a wire from Z37 pin-10 to X3 pins 12,13,14,15. X3 pins 12,13,14,15 should already be shorted together.

This completes wiring of the memory modification. Before turning on power please check all of your work, ensure that there are no solder shorts, that all connections are firm and strong and that all the wiring is correct.

When you turn on the Model I it should start normally. Enter the command :

PRINT MEM

expect the answer to be 48338.

If the answer is not 48338 recheck your work.

Some words on how the modification works, why it works and what else may be possible.

So first lets look at the address line connections to the 4164's. All the DRAM memory chips in use at this time use a form of multiplexed addressing. You may have noted that the chips (4164) have only 8 address lines which, if you know how to count in binary, can only address 256 memory locations (2 to the 8th).

The chips on the other hand have two address "qualifier" functions RAS and CAS (Row Address Select and Column Address Select) so each address line can be used twice effectively producing 16 address lines which means the chip can address 2 to the 16th locations. This is 65536 locations but is still known as 64K since 1K = 1024 locations.

Z35 and Z51 handle the issue of which set of addresses are presented to the 8 address lines on the memory chips. Each of these chips has two sets of inputs, pins 2,5,11,and 14 form one input set while pins 3,6,10 and 13 form the other. Pins 4,7,9 and 12 form the output and are connected to the memory chips. These chips act like a switch, when CAS is active one set of input lines connect to the output lines and when RAS is active the other input connects through. So this is how we get the multiplexing effect we talked about above to work.

Prior to the modification Z51 pin-14 and pin-13 were not used and on the output side pin-12 was not used. During the modification we connected address line A14 to pin-14 and address line A15 to pin-13 allowing these two additional address lines to participate in determining how many addresses would be active in the memory chips. In fact with this implementation all 64K addresses in the RAM chips can be addresses though as we will discuss shortly 16K are never used.

The Z80 CPU (Central Processing Unit) of the Model I has 16 address lines allowing it to address 64K memory locations. The main board of the Model I supports a maximum of 32K of memory (unless my modification is installed). Model I memory is "mapped" for several functions including Level I or Level II basic, video memory and the keyboard all in the lowest 16K of memory.

Address decoding is handled by Z21 and the jumper block at X3. All of this is clearly explained in the Model I Technical Reference Manual but I will give a brief tour just so the memory modification can be understood.

Z21 is wired as a three line decoder (decoding address lines A14,A13,A12) and has a logic map that looks like this:

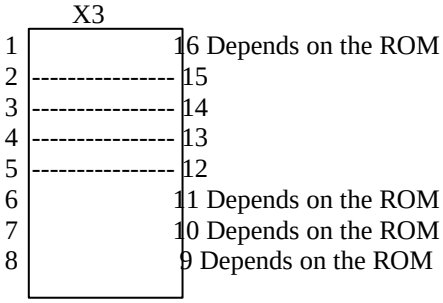
A14	A13	A12	Decoded Value	Pin Number
0	0	0	0	9
0	0	1	1	10
0	1	0	2	11
0	1	1	3	12
1	0	0	4	7
1	0	1	5	6
1	1	0	6	5
1	1	1	7	4

From this we can see that A14 divides the decode into two equal parts, the part where A14 is 0 and the part where A14 is 1. From the 64K memory capability of the Model I this represents two 16K memory segments, and in this case the lowest two 16K memory segments.

The other two address lines (A13 and A12) then serve to divide each of the two 16K segments into 8 4K segments. Interesting that the least amount of memory the Model I can handle is 4K isn't it?
We know from the Model I memory map that the ROM (s) and memory mapped locations are all contained in the lowest 16K of memory address space, the top 4 sets of addresses in the table above.

How this 16K of memory is decoded to allocate keyboard, video, and ROM is dependent on four of the X3 jumpers (16-1, 10-7, 9-8, and 11-6). Exactly how these four are set depends on the ROM used, (Level I, Level II and/or the types of ROM chips used)

For our purposes here we need to look only at the conditions when A14 is set to 1 (one), which is the second 16K of memory. From the table we can see that this is also subdivided by A13 and A12 into 4K sections, four of them. We also see that Z21 pins 7,6,5,4 connect to the X3 jumper block which then programs the Model I main board for 4K, 8K, 12K or 16K of memory. Lets be clear for 16K of memory X3 is set like this:



The Z21address decoder is enabled by two signals which come through Z73 pin-4,5,6. The combination of these two signals is what turns the decoder on so lets look at the logic of the Z73 gate and remember that Z21 is on when it's G1-G2 connections are low (zero).

A15	RAS*	Z21
0	0	0 Enabled
0	1	1 Disabled
1	0	1 Disabled
1	1	1 Disabled

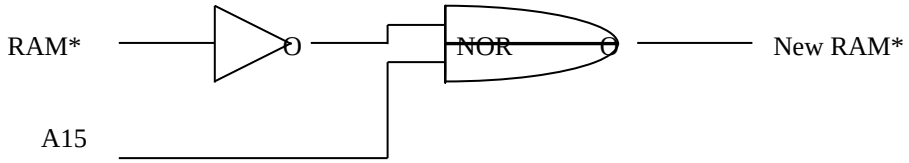
As the most significant address bit we know that A15 divides memory into two 32K pieces. From this chart we can see that any time A15 is a one memory on the Model I main board is disabled because the Z21 address decoder is disabled. This is exactly correct because the top 32K is supposed to be in the Expansion Interface (EI).

Knowing this it becomes clear that for the memory modification to work we need to tell the Model I main board that it's upper 32K of memory is local, not on the EI. So how can we do this.

Looking further at the address decoder logic we see that the X3 connections we talked about above produce a signal called RAM* when the CPU wants to talk to RAM as decoded by the Model I. We also know that A15 being a one (1) says the CPU wants to talk to the upper 32K of it's memory map and that in the Model I this is always RAM.

Logically all we have to do is be sure the RAM* signal is active when A15 is a one (1).

This is the logic required to do this:



What this tells the Model I is that all the addresses previously indicated by RAM* are still true but in addition any time A15 is high (one (1)) RAM* is also true. This is what we implemented when we did all the work on Z9 as our inverter and Z37 as our NOR gate.

The missing 16K

If you have been paying attention I am sure you realize that there is 16K of memory in the 4164 that never gets used because the Model I maps this address space to ROM, keyboard and video. If you are ambitious there is an opportunity here for some bank switching to make use of the missing 16K. This is nothing new computers like the Apple II already did this.

Here is a recent update.

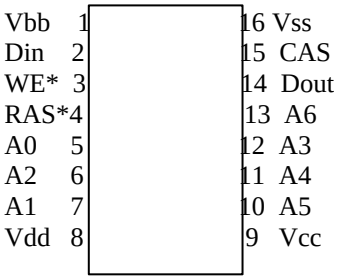
During this modification pin-1 (one) of the RAM memory chips (4164 s) was grounded simply as a best practice for TTL chips. With the advent of the 41256 chip this pin is implemented as A8, the ninth address line. The 41256 chip can be plugged directly into the sockets and will work just fine.

If you want bank switching the 41256 chip will have lots of unused memory for you to play with.

Hope all of this makes some sense. If there are issues or problems please contact me at don040@sympatico.ca

Don Kelly

4116



Vbb = -5V, Vss=GND, Vdd=+12V, Vcc=+5V

4164