4MB RAM UPGRADE FOR 520ST(M)

joska@online.no, 2015

INTRODUCTION

This document describes how to upgrade an Atari 520ST/STM to 4Mb using four 1Mb 30-pin SIMMs. This is **not** a plug'n play procedure – don't attempt this upgrade unless you're fairly familiar with a soldering iron. Pay attention and verify your work for every step. If you can read schematics and use a multimeter you can apply this upgrade to any ST.

Please note that when I started this upgrade I intended to use a 4Mb 72 pin SIMM. Unfortunately I could not get this to work reliably with any of the SIMMs I had, so I switched to 30 pin SIMMs. Some of the illustrations will be out of sequence and contain traces of the original SIMM upgrade because of this.

REQUIREMENTS

You will need some basic tools – a good soldering iron, pliers, screw driver for opening the case, the usual stuff. If you're capable of doing this upgrade you will already have all (and more!) the necessary tools.

Components:

- Four 1Mb 30-pin SIMMs. These can be of any type. Make, number of chips, parity/no parity etc doesn't matter.
- Wire. Wrapping wire, old IDE cables, whatever you have. Ribbon cable (IDE cables, floppy cables…) is ideal.
- Three 68 ohm resistors.
- One 33 ohm resistor.
- Optional: Stripboard.
- Optional: SIMM sockets or pin headers.

HOW IT WORKS

The ST – like virtually every other home computer since the 70's – use DRAM memory. DRAM access is typically multiplexed, which means that you get away with a lot less address lines than with direct access. The memory cells are arranged in a matrix, and the same address lines are used to address both the rows and the columns. To do this two types of additional signals are used – RAS (Row Address Strobe) and CAS (Column Address Strobe). This is all handled by the ST's MMU (memory management unit).

The MMU sits between the DRAM chips and the address bus, and decodes RAM addresses to DRAM column and row addresses. So when the CPU attempts to read an address in RAM, the MMU will transform this address into a row address and a column address. It will then put the row address on the DRAM address bus and strobe the RAS signal. Then it will put the column address and strobe the CAS signal. At this point the DRAM(s) will put the contents of the cell in the addressed row and column on the data bus, and the CPU can read it.

DRAM chips cells are typically 1, 4 or 8 bit wide, so it takes between 16 and two chips to fill the ST's 16 bit data bus. If each DRAM chip contains 256k one-bit cells, you need 16 of these for a total of 512Kb. This is the case in the 520ST.

In the ST the memory is arranged in two banks. Each bank in 16 bit wide, and can have either zero, 128Kb, 512Kb or 2Mb each. The ST will automatically detect the amount of memory in each bank. The 520 has 512Kb in the first bank and nothing in the second.

Each bank is again divided into two sub-banks, each 8 bits wide. To access the correct DRAM cell the MMU then needs a total of six signals – one RAS-signal for each of the two 16 bit main banks, and one CAS-signal for each of the four sub-banks.

SIMMs are just circuit boards with DRAM chips on them. The physical dimensions of the PCB and the layout of the signal connector is standardized, and contains all the signals found on the DRAMs. Connecting these to the ST is just a matter of routing the signals correctly. No other circuitry is required.

30 pin SIMMs have one RAS and one CAS line, eight data lines and up to 12 address lines (depending on size). So to upgrade an ST to four Mb you will need four 1Mb SIMMs. A single 4Mb SIMM is of no use, the ST can only access the first 1Mb of it.

SIMM		ST
1	VCC	5V
2	/CAS	CAS0L, CAS0H, CAS1L, CAS1H
3	DQ0	D0, D8
4	A0	A0
5	A1	A1
6	DQ1	D1, D9
7	A2	A2
8	A3	A3
9	GND	GND
10	DQ2	D2, D10
11	A4	A4
12	A5	A5
13	DQ3	D3, D11
14	A6	A6
15	A7	A7
16	DQ4	D4, D12
17	A8	A8
18	A9	A9
19	A10	n/c
20	DQ5	D5, D13
21	/WE	RW
22	GND	GND
23	DQ6	D6, D14
24	A11	n/c
25	DQ7	D7, D13
26	QP	n/c
27	/RAS	RAS0, RAS1
28	/CASP	n/c
29	DP	n/c
30	VCC	5V

The SIMMs are connected to the ST as shown in the table to the left. The blue rows are signals that are common for all four SIMMs, the yellow rows shows signals that are unique to each SIMM or pair of SIMMs.

There are two types of signals that are not common to all SIMMs – the data lines, which are common to pairs of SIMMs, and the RAS/CAS lines.

The data lines are connected to the ST's data bus like this:

	Bank	ST data bus
SIMM 1	Bank 0 low	D0-D7
SIMM 2	Bank 1 low	D0-D7
SIMM 3	Bank 0 high	D8-D15
SIMM 4	Bank 1 high	D8-D15

The RAS and CAS lines are connected to the ST's MMU like this:

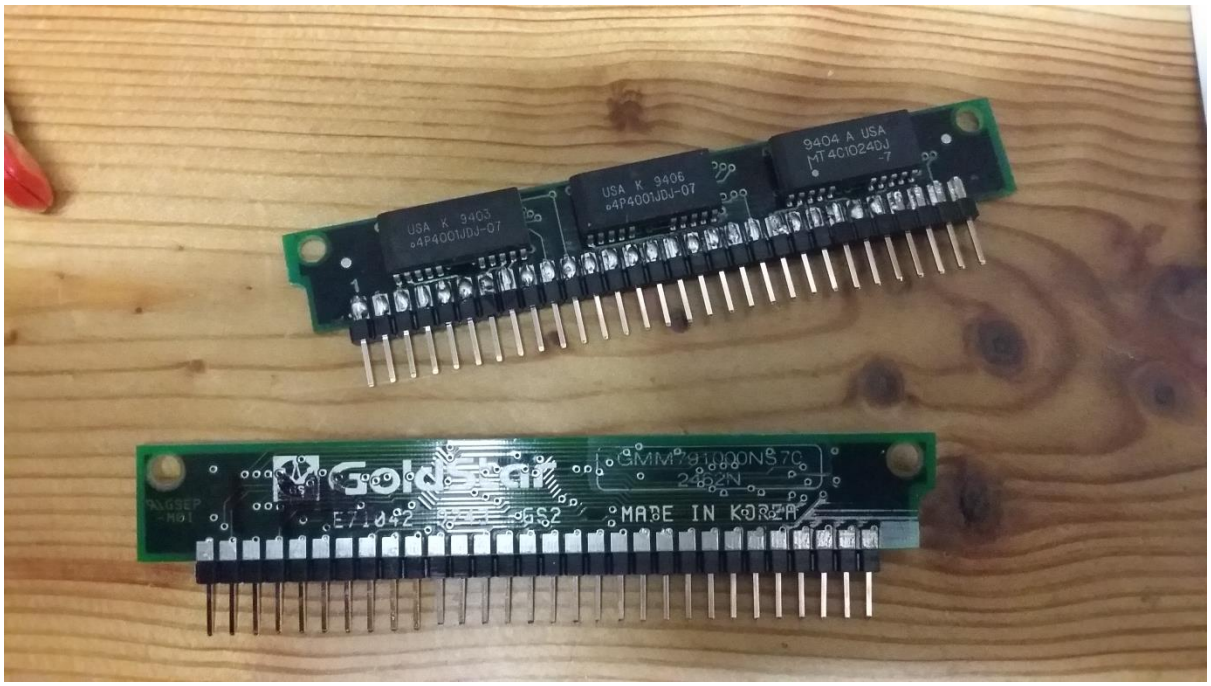
ST	SIMMs
RAS0	RAS on SIMM 1 and 3
CAS0L	CAS on SIMM 1
CAS0H	CAS on SIMM 3
RAS1	RAS on SIMM 2 and 4
CAS1L	CAS on SIMM 2
CAS1H	CAS on SIMM 4

PREPARE THE SIMMS

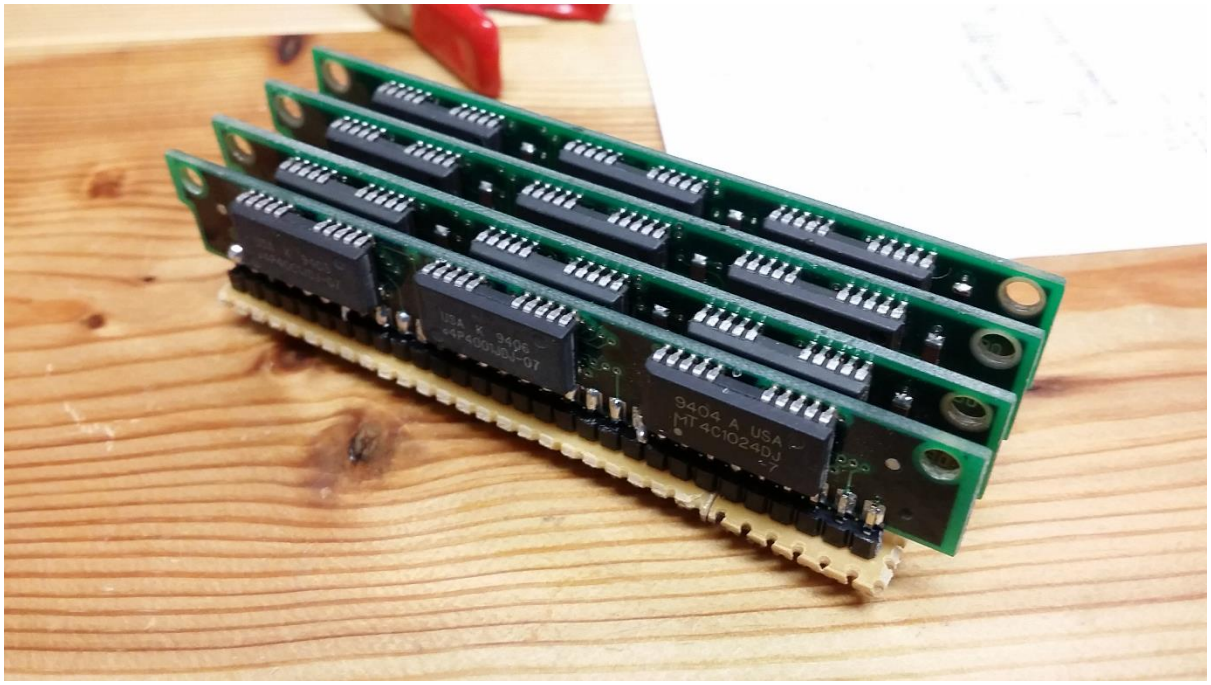
Most connections are common to all four SIMMs, so to make wiring tidier and easier it's recommended to mount the SIMMs (or SIMM sockets if you have them) to a piece of stripboard.

It's a very good idea to verify that the SIMMs are OK before you solder them in. I tested them in my STE.

If you don't have sockets – start by soldering pin headers to each of the four SIMMs.

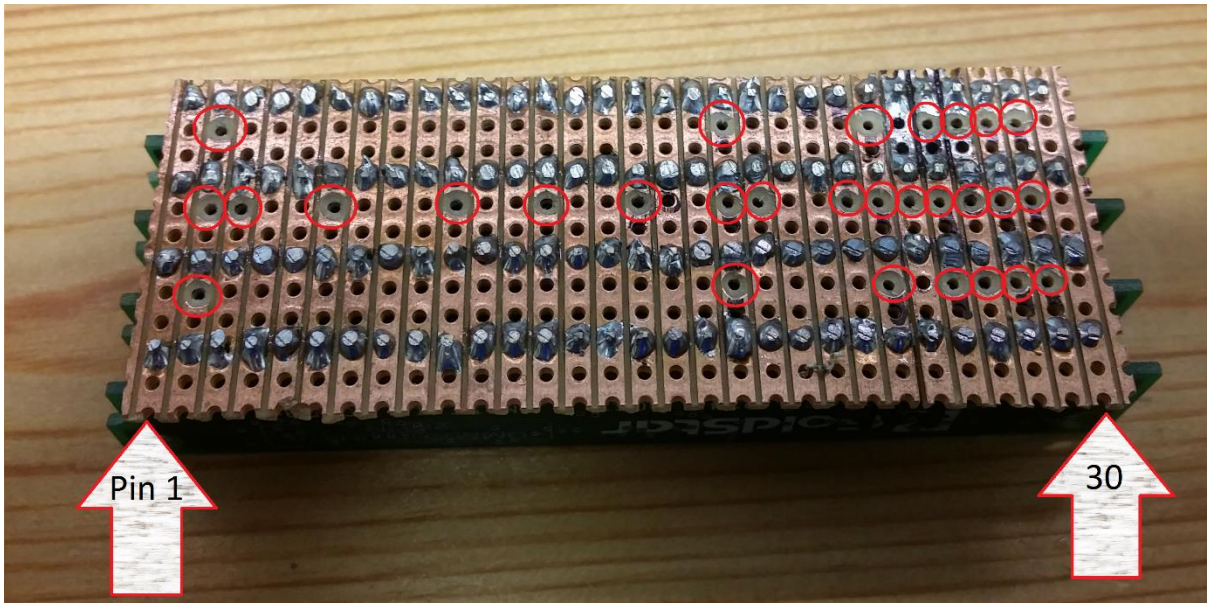


Then solder the four SIMMs – or SIMM sockets if you have them - to a piece of stripboard. All of them must face the same way.



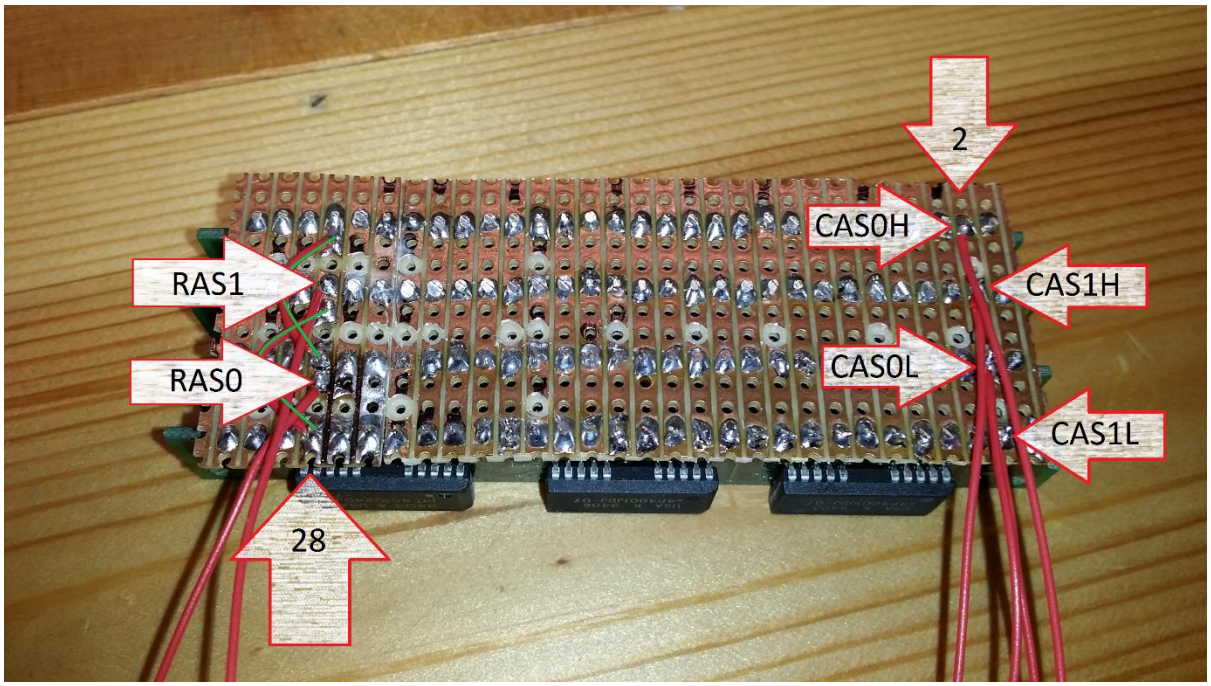
If you look closely you can see that I used two pieces of stripboard. Don't do that unless you have to piece together bits of stripboard from the scrap like I did.

Now you have to cut some of the traces. I use a 5mm drill bit for this. Make sure that the cut traces really are cut.



Pin 1 is the notched side of the SIMM.

Now add the various wires to the SIMM module.



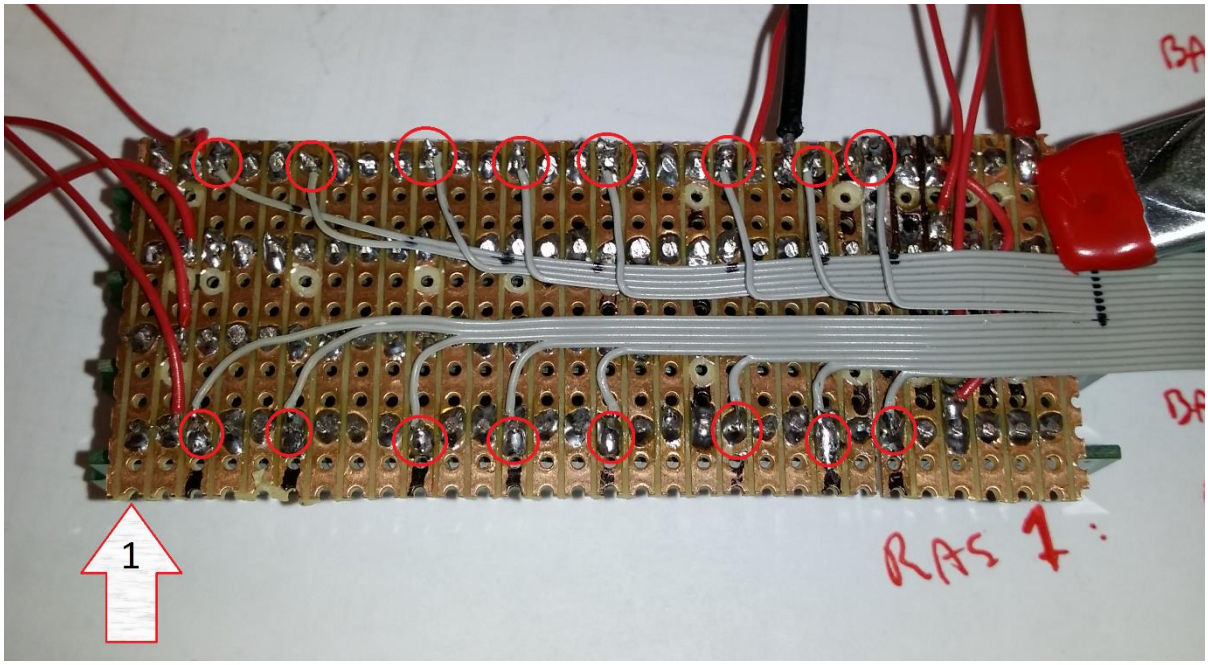
This image shows the RAS and CAS signals. Please note the two RAS jumper wires in row 28!

Cut six wires to about 20cm length, strip and tin them. Then solder them to the stripboard as shown here. It's a good idea to label the wires with a piece of paper tape or similar, that makes it much easier when you get to the point to connect the wires to the motherboard.

Please label the RAS and CAS wires according to this table:

Label	SIMMs
RAS0	RAS on SIMM 1 and 3
CAS0L	CAS on SIMM 1
CAS0H	CAS on SIMM 3
RAS1	RAS on SIMM 2 and 4
CAS1L	CAS on SIMM 2
CAS1H	CAS on SIMM 4

Now attach the 5V, ground and data wires.



For the data bus I've used 16 strands of an 80 pin IDE cable. You need around 25cm of this.

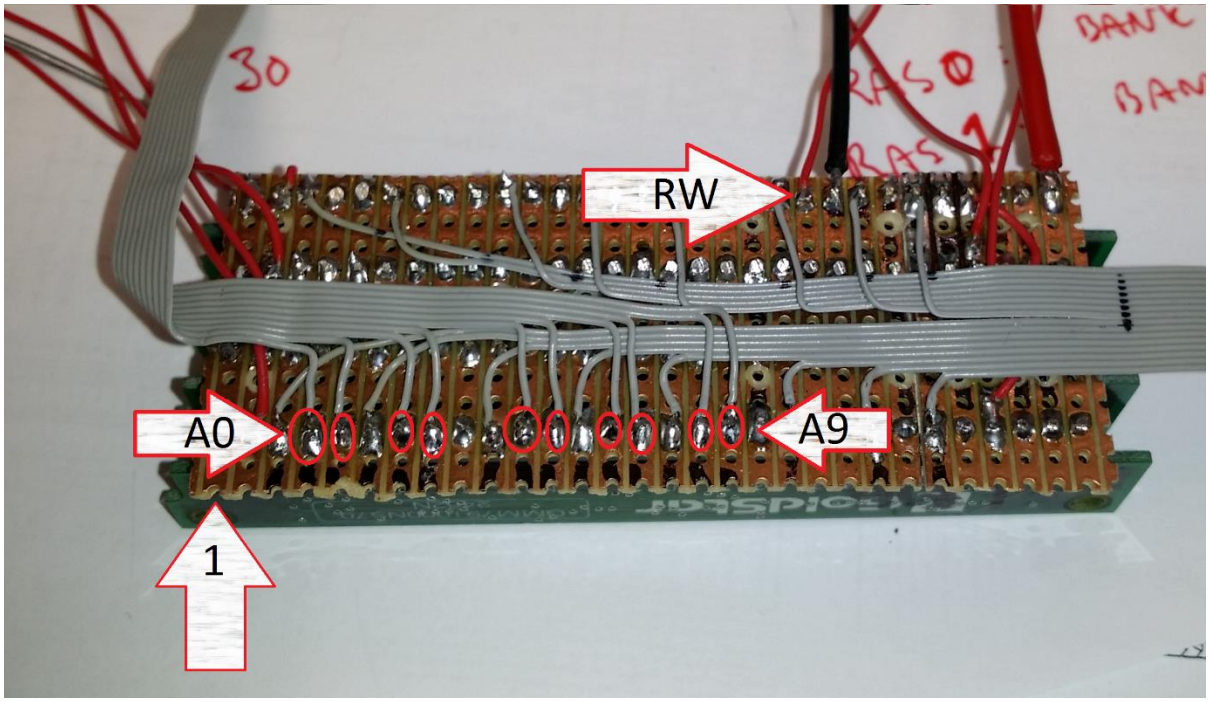
Top row, left to right: D0, D1, D2, D3, D4, D5, D6 and D7.

Bottom row, left to right: D8, D9, D10, D11, D12, D13, D14 and D15.

These will be connected directly to the ST's data bus. Make sure to label D0 on the other end of the cable to avoid connecting it "upside-down" to the motherboard. It's a very good idea to connect the wires in ascending order, this makes it much cleaner and easier to connect them to the ST's data bus.

5V is connected to either pin 1 or pin 30, ground to pin 9 or pin 22.

The next step is to connect the address wires. For this you need 10 wires, preferably from a flat cable like above. Cut the wires to around 25cm and solder them to the SIMMs as shown in this picture.



Now cut a piece of wire to around 25cm, strip and tin it and solder it to pin/row 21 of the SIMMs. This is the RW line, make sure to label the wire for later.

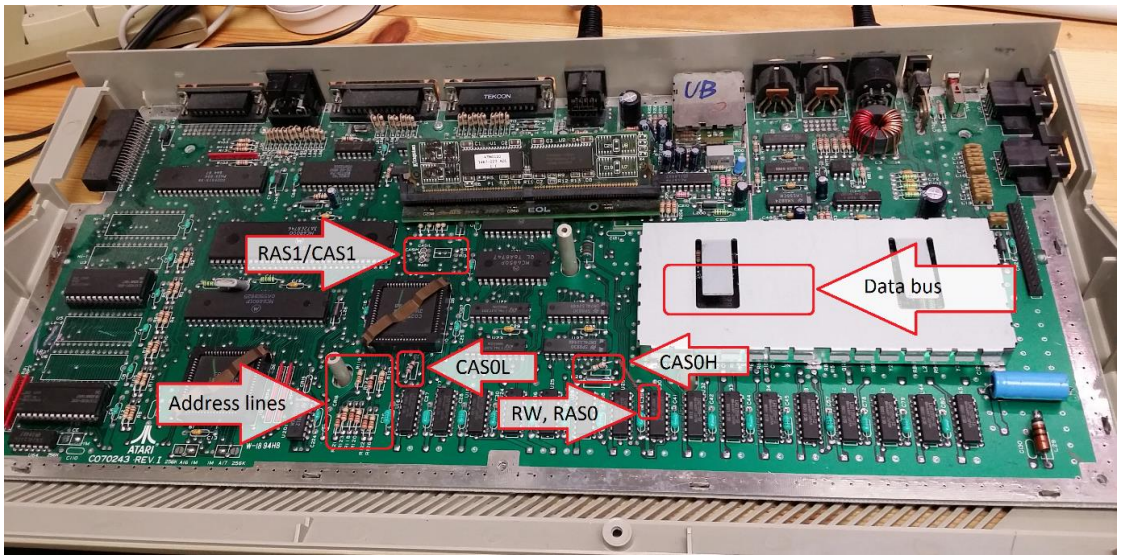
Double-check all connections, make sure that you've connected the wires correctly and that there's no short-circuits. Also, make sure that all wires are correctly labelled. You should have the following labelled wires:

- Six wires: RAS0, CAS0L, CAS0H, RAS1, CAS1L, CAS1H.
- One wire: RW or WE (same thing).
- 16 data wires (preferably a flat cable but separate wires are OK too) labelled D0 to D15.
- 10 address wires (again, preferably a flat cable) labelled A0 to A9.
- One 5V and one ground/gnd.

If everything checks out the SIMMs are now ready to be connected to the ST.

PREPARE THE ST

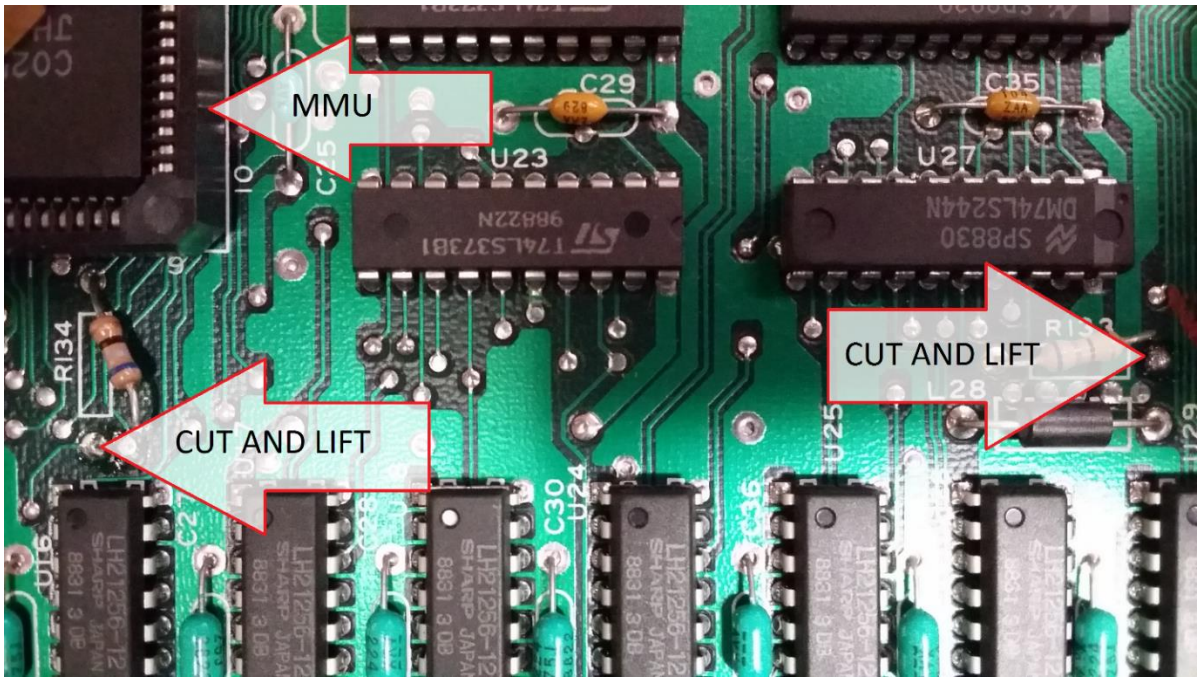
Start by removing the top case, keyboard and top metal shielding. You need full access to the motherboard. Leave it place in the lower case for now.



This is what you should see now. The location of the various signals is outlined, these are the areas we will work on. The SIMMs are located behind the keyboard, as they won't clear the shielding and keyboard otherwise.

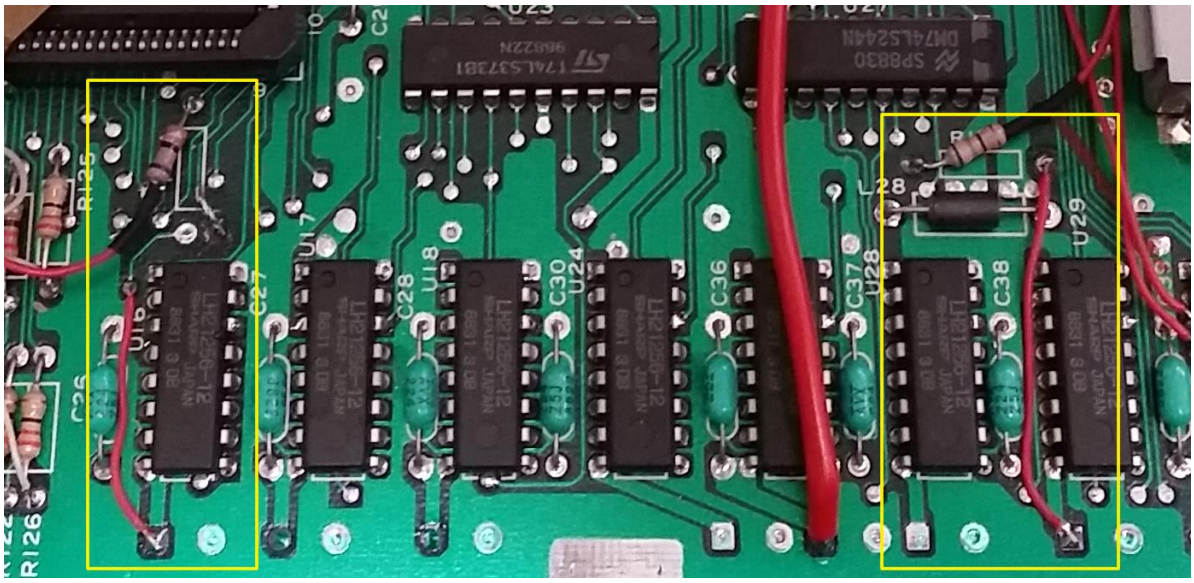
DISABLE THE ORIGINAL RAM

The first step is to disable the original RAM. Some tutorials suggests to remove the RAM chips – I recommend against that unless you're very familiar with this kind of work. Leaving the RAM chips in place is less work, less risky and it's easy to revert if you should want to do that in the future. Disabling the original RAM is easily done by separating the CAS0L and CAS0H from the RAM chips.



Cut one end of the resistors R134 and R133 as illustrated above. Lift the cut end from the PCB. We will connect the SIMMs' CAS-lines to the cut end of these resistors later on.

Now connect the solder pads on the "cut" end of the resistors to a 5V source. Pin 8 on a nearby RAM chip is a suitable source.



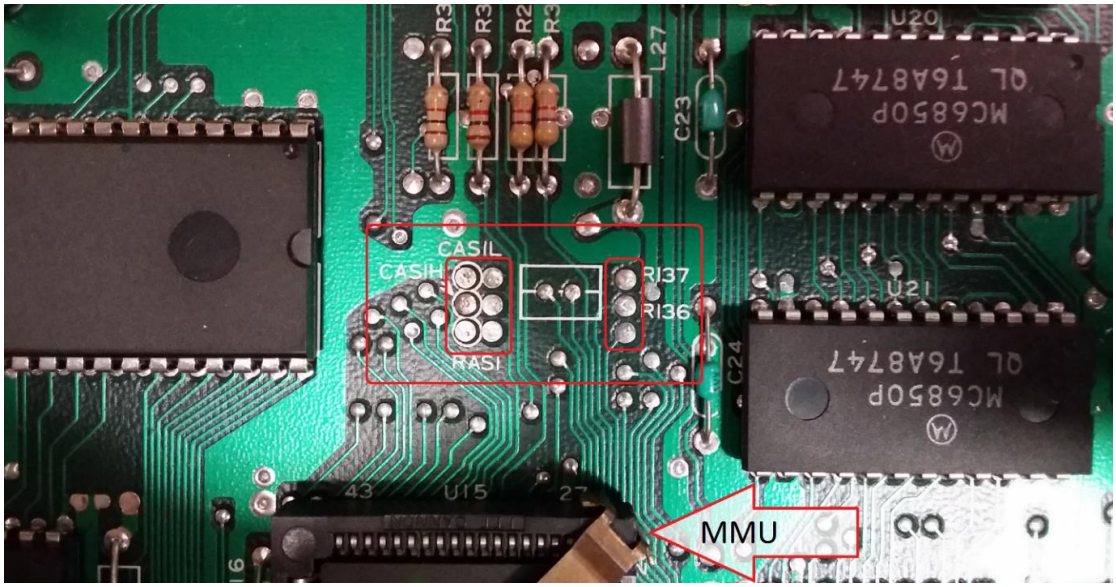
Please ignore the other wires. I forgot to take a picture of this step so this is from the final installation.

PREPARE RAS AND CAS SIGNALS FOR THE FIRST RAM BANK

RAM bank 0 is populated by the now disabled RAM chips. Carefully straighten the cut ends of resistors R133 and R134. These are the sources for CAS0L and CAS0H. RAS0 will be taken directly from pin 4 on (any) one of the original RAM chips.

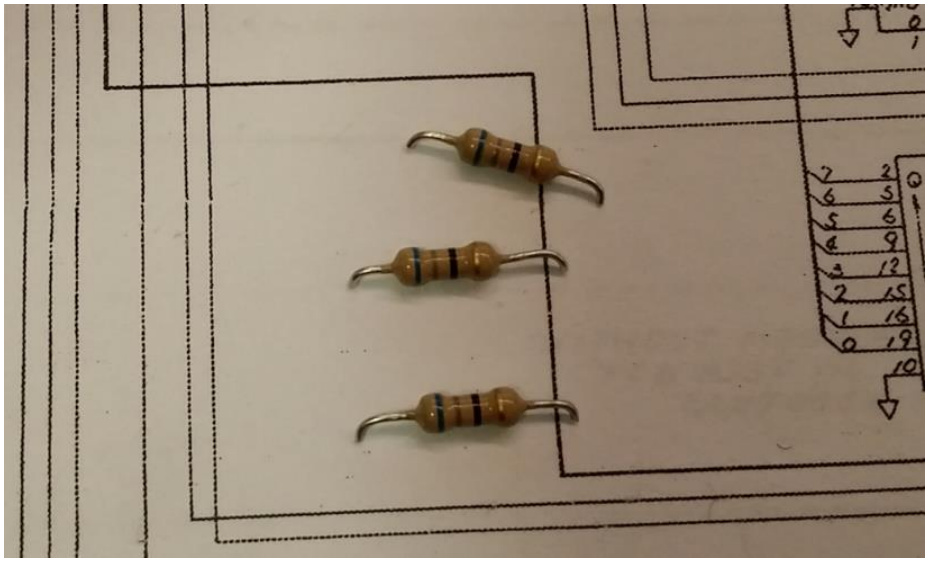
ENABLE THE RAS AND CAS SIGNALS FOR THE SECOND RAM BANK

The 520ST is only equipped with one bank of 512Kb memory from factory. Fortunately for us the Atari engineers thought of memory upgrades and made the RAS and CAS signals for the upper bank easily available on the PCB.

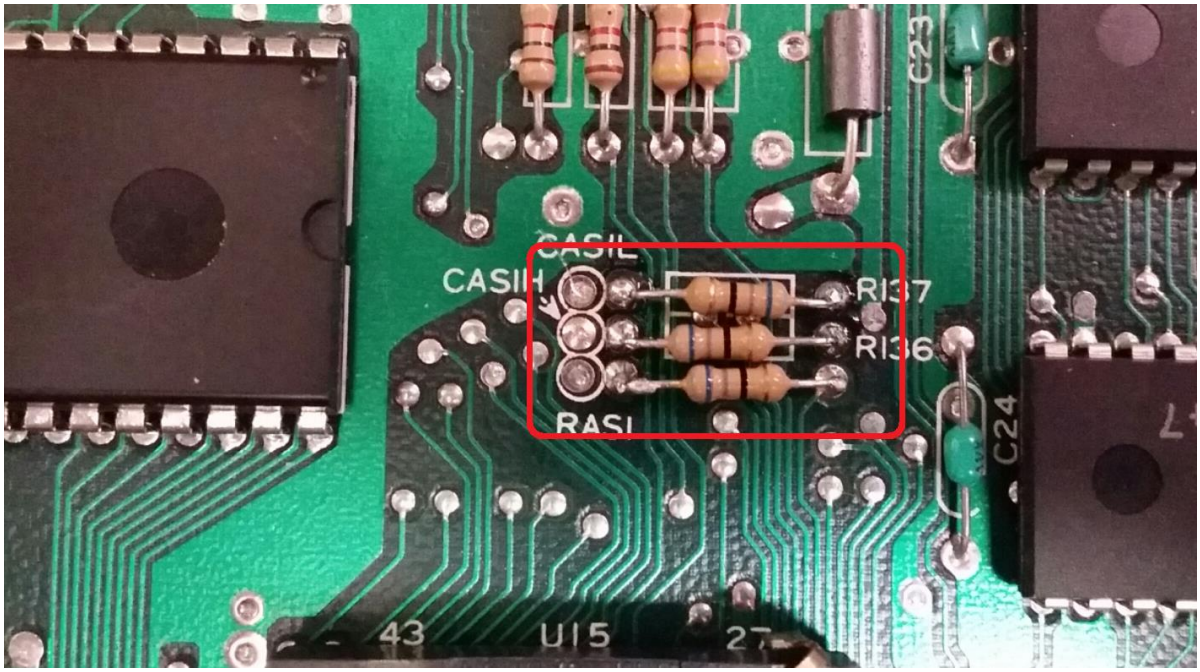


There are three solder pads above the MMU, marked “CAS1L”, “CAS1H” and “RAS1”. But the signals are not available there yet, but at the three solder pads to the right. You could pick up the signals directly from these, but instead we do it like the Atari engineers intended and fit three 68 ohm resistors to enable the labelled solder pads.

First we prepare the tree resistors by cutting and slightly bending the legs.



Now solder them to the solder pads as illustrated in the picture below. RAS1, CAS1L and CAS1H is now available at the labelled solder pads.



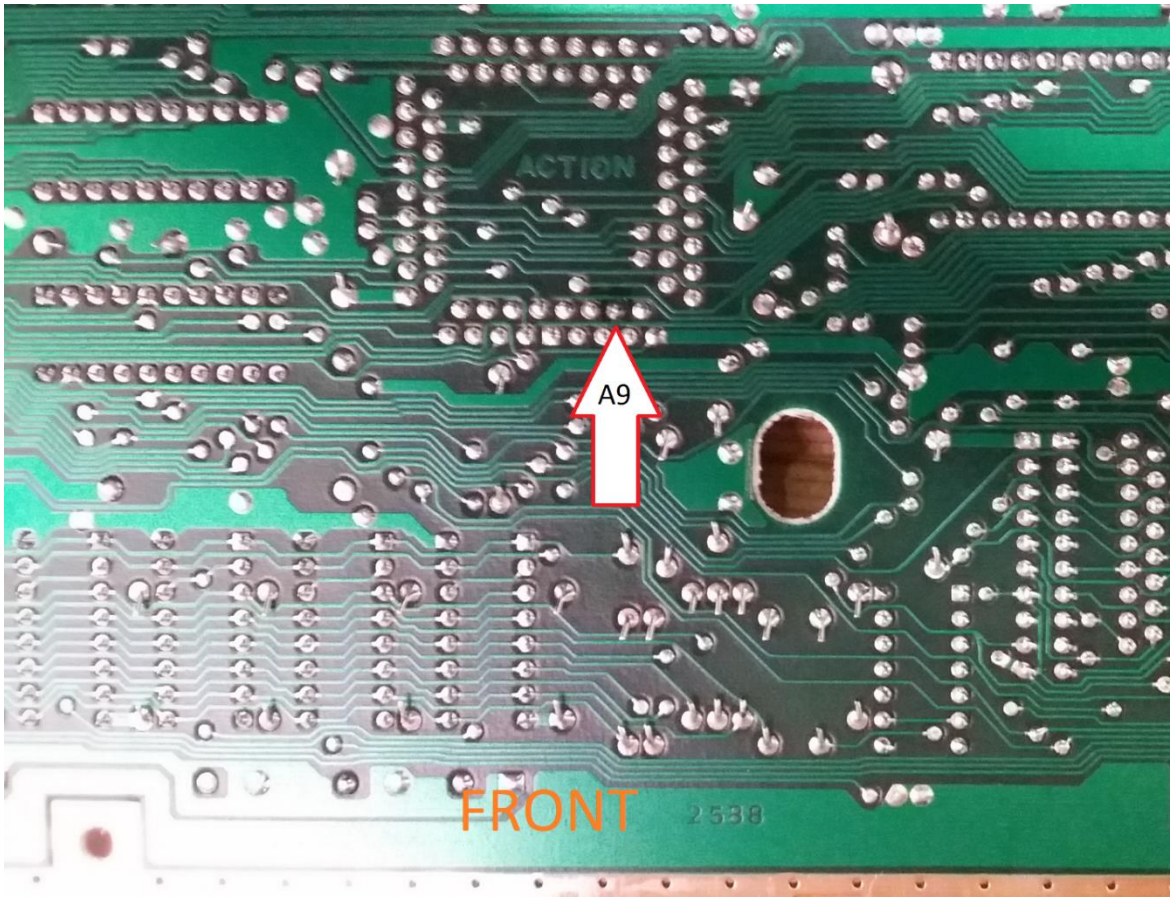
My eyesight is not what it used to be, as the poor soldering clearly shows…

RW/WE AND A9

RW/WE is picked up from pin 3 on any of the original RAM chips.

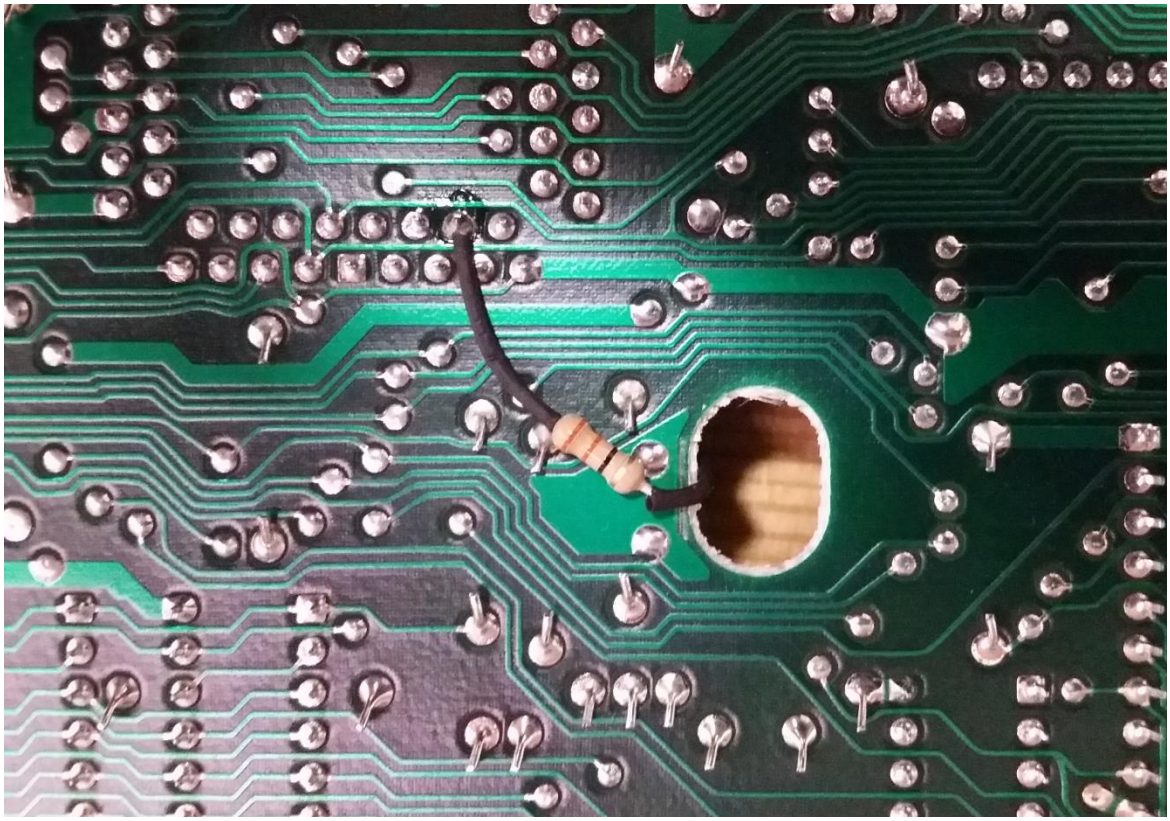
A9 is only available directly at the MMU. It's strange that Atari made the RAS and CAS signals for the upper bank easily available, but did not care about A9. Maybe they thought that nobody would ever want more than 1Mb…

To get access to A9 you must separate the motherboard from the bottom shielding and flip it upside down. Please note that the motherboard is secured to the case with three screws, remove those instead of using excessive force…

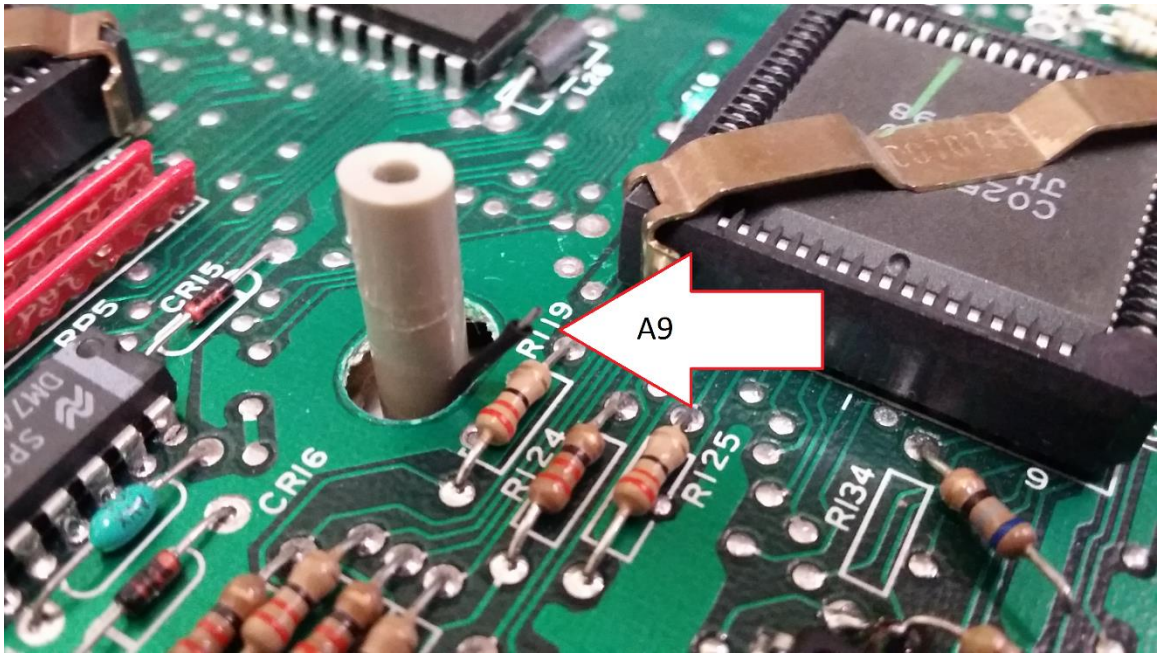


Locate the MMU soldering points. On my PCB this is marked "ACTION" for some unknown reason. A9 is available at the indicated solder pad.

Now prepare a 33 ohm resistor by covering the legs – apart for the last 1-2mm – with heat shrink tube. If you don't have heat shrink tube you can apply some tape to the motherboard to prevent short-circuits.



Solder one end to the A9 solder pad, and thread the other end through the nearby hole. This hole is for one of the case pillars.



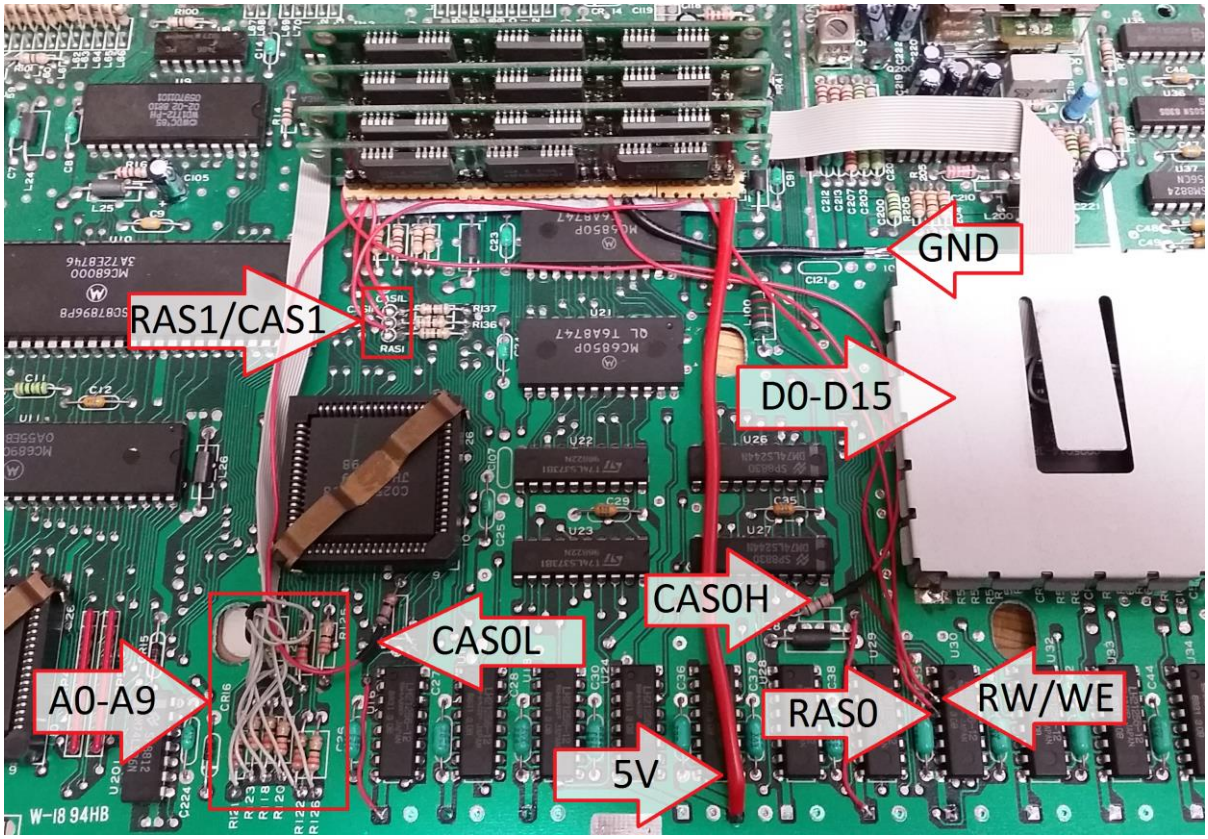
Flip the motherboard, fit the shielding and place it in the case again. The end of the resistor is now available near the MMU and is the source for the A9 signal.

The motherboard is now ready for the SIMMs.

FINAL STEP – CONNECT THE SIMMS TO THE MOTHERBOARD

This is the fun part, piecing it all together. If you've done a good job with the previous steps this will be very easy and a lot of fun.

First, start by finding a good spot for the SIMM module. In this picture you can see where I've located it:



It's relatively close to all the relevant parts in the ST. If you don't want to reinstall the upper case shielding you can locate the SIMMs further forward, as long as it doesn't conflict with the keyboard. You might also have lower profile SIMMs than me, or use angled SIMM connectors. In both cases you can choose location more freely.

When you've decided where to put the SIMMs, you can cut the various wires to length, and strip and tin the ends. Then connect them to the soldering points highlighted in the picture above.

RAS/CAS: RAS0, CAS0L and CAS0H are clearly marked in the picture. RAS1, CAS1L and CAS1H are labelled on the motherboard itself. You should have no problems here, as long as you labelled the wires correctly when preparing the SIMMs. Pay special attention to RAS0 – connect this to pin 4 on any of the original RAM chips.

RW/WE: Pin 3 on any of the original RAM chips. Pin 1 is marked with a dot, count counter clockwise from this.

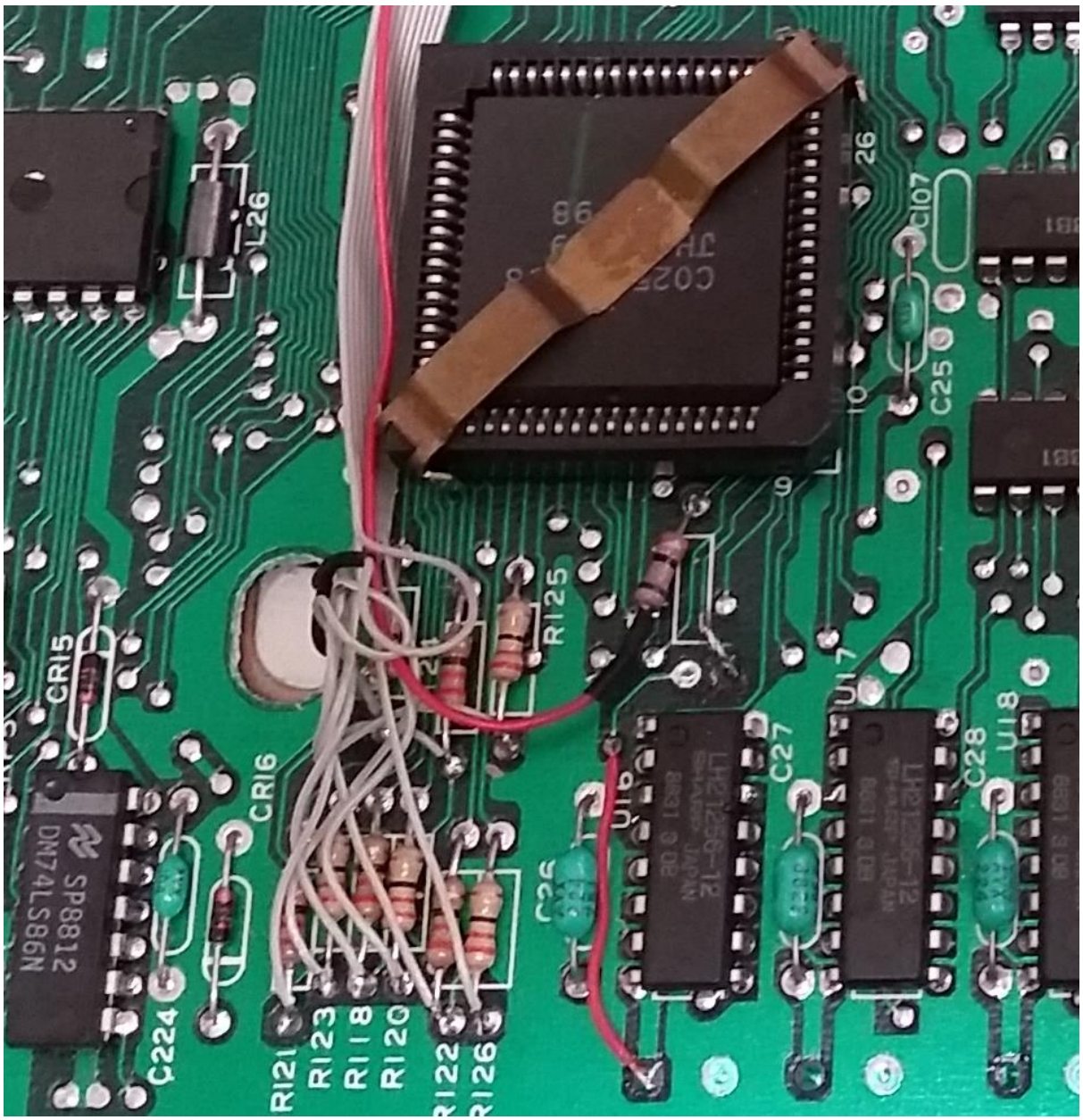
5V: I've chosen to connect this to the power supply for the original RAM chips. Don't just pick any random 5V source on the motherboard, the power to the various parts of the motherboard is distributed through a set of inductors and tapping into a random circuit might cause an overload.

Ground: Just pick a ground spot close to the SIMMs.

You're nearly there. Just 26 more wires to go!

ADDRESS LINES

The address lines are connected to a set of resistors close to the MMU, including A9 which is connected to the resistor you soldered to the underside of the motherboard.



The address lines in this picture are the grey wires. Please note the orientation – solder them to the end of the resistors which is furthest away from the MMU. Connect the address wires to these resistors according to the following table:

A0	R124	**A5**	R119
A1	R123	**A6**	R118
A2	R122	**A7**	R126
A3	R121	**A8**	R125
A4	R120	**A9**	MMU

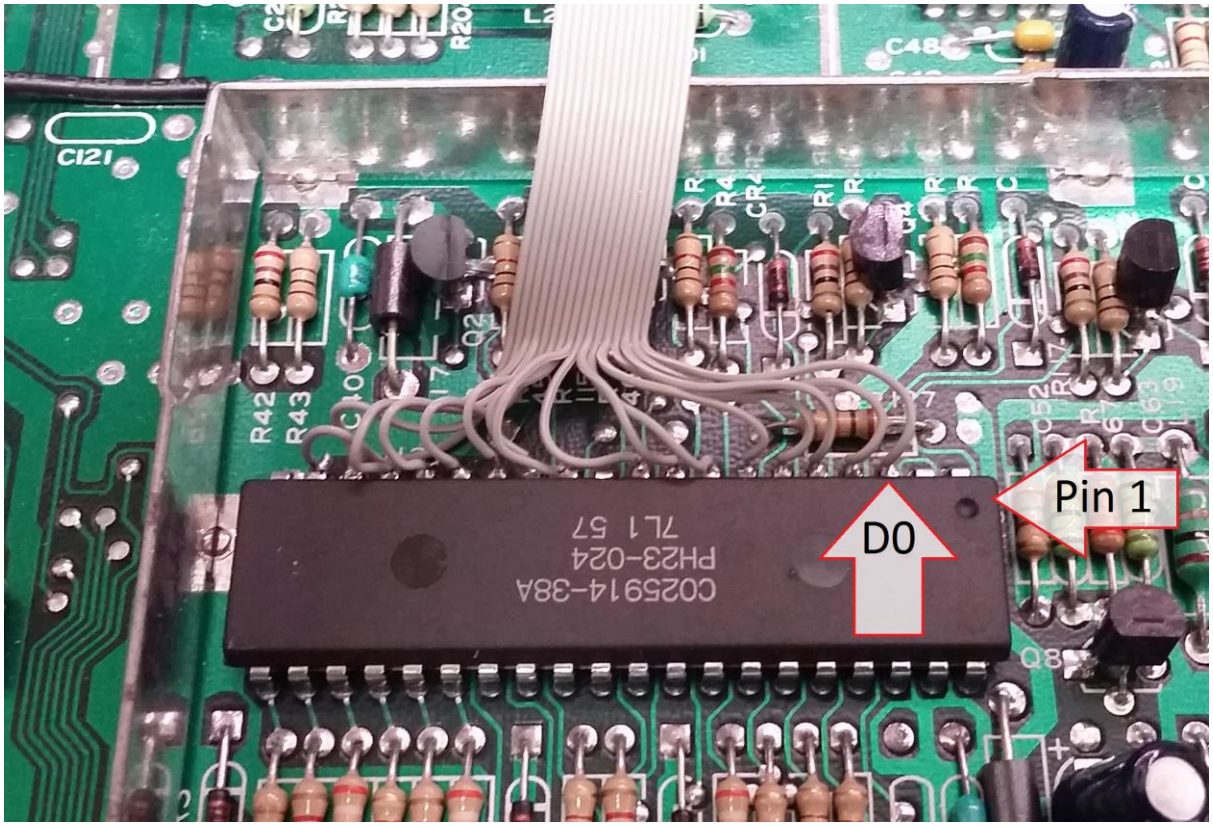
A9 is connected to the resistor you soldered to the underside of the motherboard.

DATA LINES

The 16 data wires from the SIMMs are connected directly to the ST's data bus. There are several ways to do this, I've chosen to solder them to the shifter. This is a DIP chip which is easy solder to. Another good spot would be the two data bus drivers U26 and U27.

You can also use the CPU's data lines, but that complicates any future upgrades like an IDE interface or alt-RAM card.

Now, pop the lid off the shifter box.



Route the data wires to the shifter, and cut, strip and tin them if you haven't done so already. Connect the wires to the shifter according to this table:

SIMM	Shifter	SIMM	Shifter
D0	3	**D8**	12
D1	4	**D9**	13
D2	5	**D10**	14
D3	6	**D11**	15
D4	7	**D12**	16
D5	8	**D13**	17
D6	9	**D14**	18
D7	10	**D15**	19

Pin 1 of the shifter is marked with a dot, count counter clockwise from this.

Double check all connections and put on the shifter box lid.

FINISHED!

That's it, everything is now hooked up. Double-check and triple check everything again. Make sure that there's no short-circuits, and that you haven't left any tools on the motherboard… Put some tape or paper between the SIMMs and the motherboard to prevent short-circuits. Hook up a monitor and the PSU and switch it on. It will take a little bit longer than usual for an image to appear on the monitor, this is normal with 4Mb RAM. The ST should then boot normally.

Don't despair if it doesn't work. Disconnect the PSU and check everything again. If you can't find anything wrong you can enable the original RAM and test again. To do this you must disconnect the RAS0/CAS0 lines from the SIMMs. Connect CAS0L and CAS0H from the SIMMs to ground somewhere. Disconnect the jumper wires from R133/R134 solder pads to 5V at the 5V end, and connect them to the resistors. Test the ST again. If it works but still only has 512Kb RAM, your RAM upgrade simply doesn't work for some reason. If all connections checks out OK, this can only mean defect SIMMs.