Alesis MMT8 16x Memory Expansion Modification (Black model MMT8's)

by Graham Meredith, 2006
gmeredith1@yahoo.com.au

Revised 15th January 2009

This modification expands the memory of the Alesis MMT8 to 16x the original memory capacity. It works just as if you had 16 separate MMT8's – each "bank" is an entire MMT8 memory bank.

The total cost of the expansion should be around $US45!!

Each bank is accessed by switching a 16 position rotary switch to choose the current memory. Each memory holds 99 songs, 99 parts, 8 tracks – just like the original MMT8 memory.

The down side of this expansion is that you can't use a part from, say, memory bank 1, and another part from memory bank 2, and put them into the same song. **Each memory is isolated from the other**. However, you can save and load the memory contents from each memory bank as a whole bank, as individual songs, or individual parts, just like doing memory dump on the original memory.

You can also save individual parts or songs to MIDI or tape, change memory position, and load them into the next memory. But you can't directly copy a part or song from one memory to another. You have to save them to tape or disk and then load them into the next memory.

Equipment

Soldering iron, 15W (no greater – can damage the circuit tracks by excess heat), solder
Antistatic wrist band and cable
Wire cutters

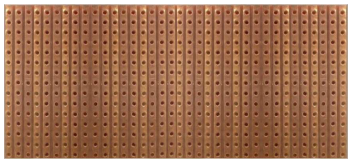
Components required

RAM chips - x 2 needed:

BSI SRAM memory chip BS62LV4006PC-70 SRAM
(512k x 8 bit) – About US$19 each – 2 needed

http://www.jameco.com/webapp/wcs/stores/servlet/ProductDisplay?langId=-1&storeId=10001&catalogId=10001&productId=157358&

Other bits:



Veroboard stripboard

Heat shrink insulation tubing - small diameter 2-3mm, black and red

Double sided mounting tape

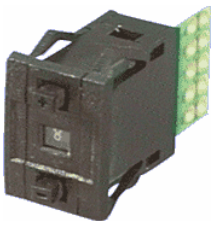
Hookup wires - insulated, of different colours, such as ribbon cable

Resistors - 10k, ¼ watt – **x 8** needed

Switches: 4-bit Hexadecimal Rotary encoder switch: about US$6.00 --- **x 1** needed

http://www.jameco.com/webapp/wcs/stores/servlet/ProductDisplay?langId=-1&storeId=10001&catalogId=10001&productId=581301&

Or this better, neat pushbutton one instead, with a number display:



https://www.jameco.com/webapp/wcs/stores/servlet/ProductDisplay?langId=-1&storeId=10001&catalogId=10001&productId=577117&

Preparation:

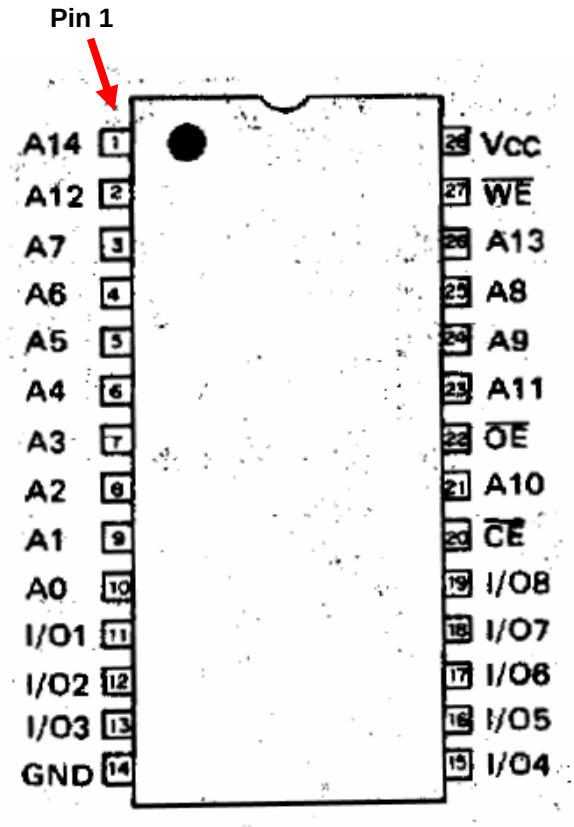
Put on the **antistatic wrist band and cable**; connect the cable somewhere to earth, such as a water tap on the kitchen bench. This will prevent static discharge building up and damaging your chips.

Study the pin out diagrams for each chip below. Also, examine the actual chips when you get them. Make sure you can identify which is **pin 1** on the chip (the top of the chip has a small **notch** in one end of the chip; pin 1 is the first pin on the left of this notch).

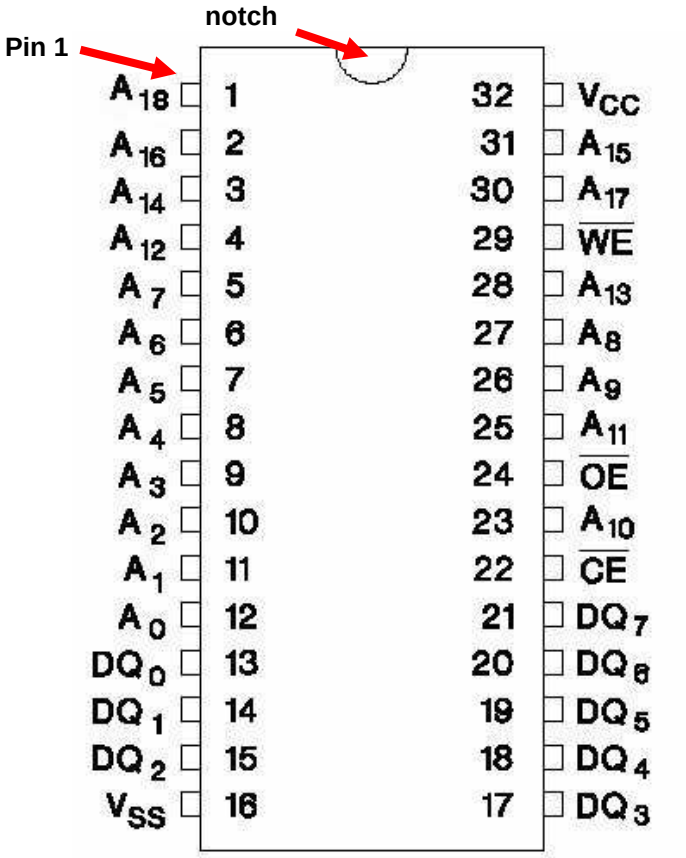
The original 28-pin 32kx8 SRAM and the new 32-pin 512x8k 4Mbit SRAM have the address pins and data pins in the same location. The differences are:

The **4Mbit** chip has **32 pins – it is a larger physical chip**, and pins 1, 2, 30 and 31on the top are the 4 **highest address** lines.

Original MMT8 RAM chips CXK58257P-12L



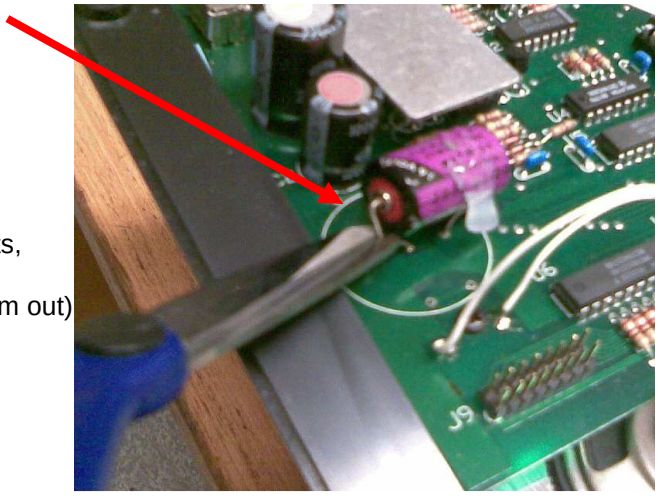
New SRAM chips BS62LV4006PC-70



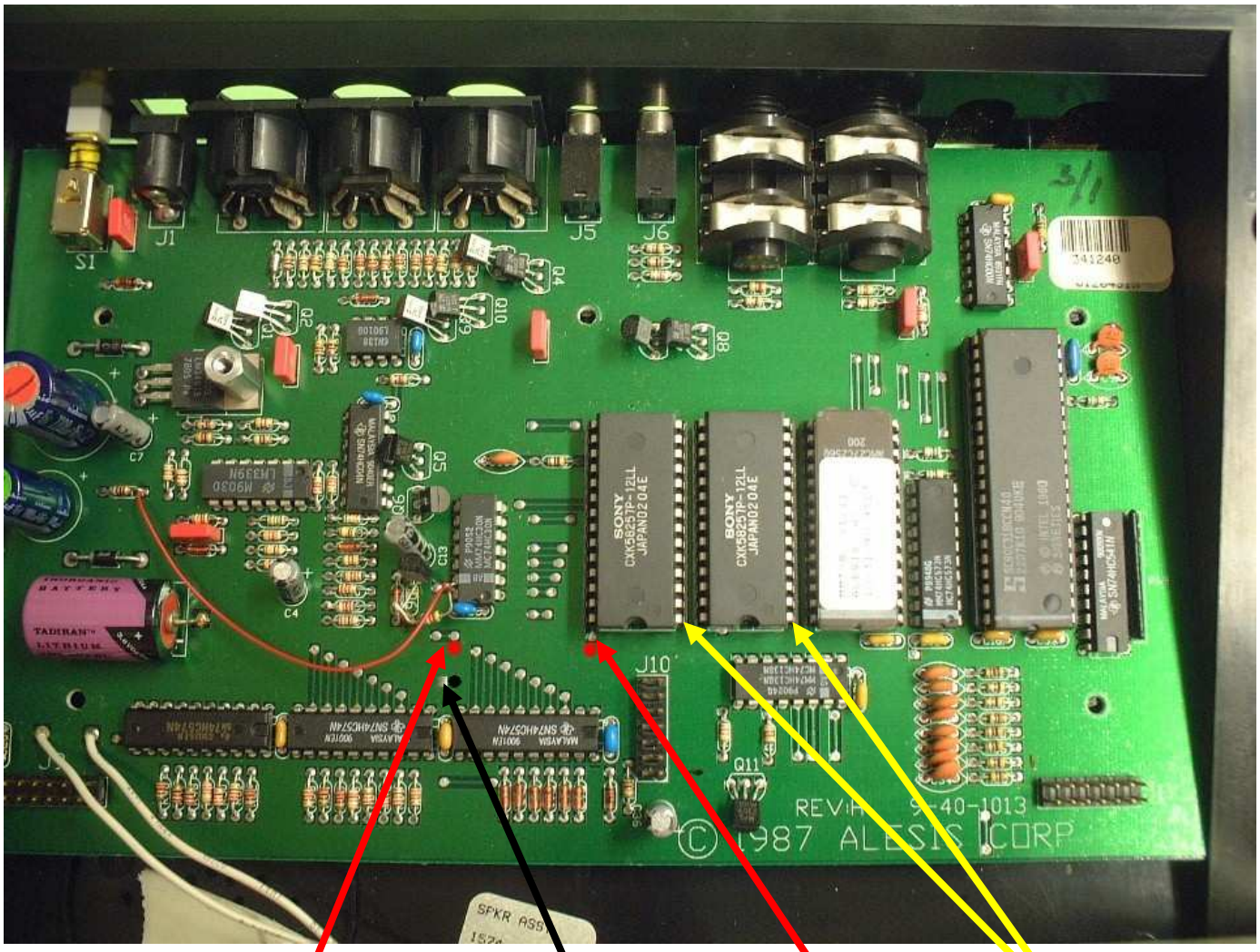
1) Disconnect the **memory battery** by cutting the **+ battery wire** that is soldered to the board. You can connect it later simply by soldering the cut wire back together. Make sure your valuable sequences are saved to disk or computer as you will lose the memory when this is done.

2) Locate the 2 **RAM chips** on the MMT8 circuit board and put a **mark on the circuit board** with a felt tip pen where the location of **pin 1** is for each chip, so you can remember which way they go in (check the pin diagram of the old chip above).

3) **Remove** the old **RAM** chips (they should already in sockets, just carefully pull them out, and leave the sockets in place. You may need a small flat headed screwdriver to prize them out)



The power supply points to the memory chips: Black MMT8's



4) **Study the circuit board photo above**. This is the **Black MMT8 circuit board** (the **grey MMT8** is different, so download the specific guide for it if you have a **grey MMT8**).

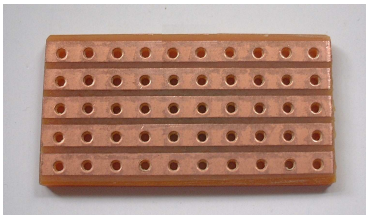
You will see a red arrow pointing to a **solder blob** on the board next to the memory chips. This is a convenient **+5V power supply point** for the new memory chips. Mark it with a red marker. In the new RAM chips, pin 32 is the **+5V power supply pin (Vcc),** which this point will connect to when we install the chips.

The other solder blob point marked with a **red arrow** is the **encoder switch power supply point**. We will mention this later.

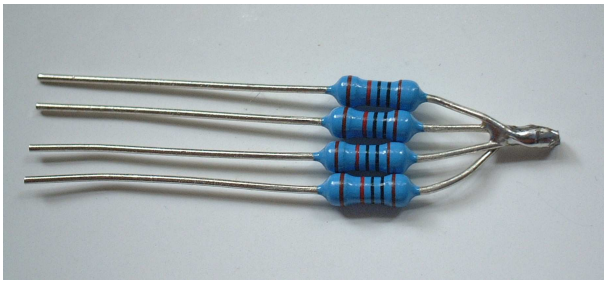
The **ground point** for the expansion is marked with the **black arrow.** It is simply a patch on the board that you scrape away the green insulation lacquer to expose the copper surface, and then solder to that point.

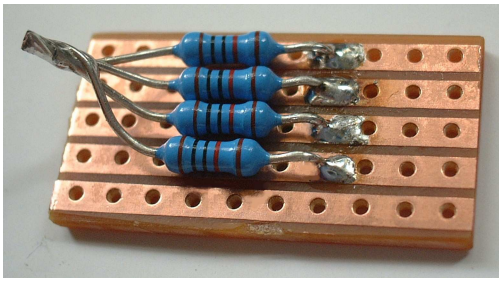
Making the Encoder board

1) Cut a piece of **veroboard strip** to the size shown below:



2) Take 4x **10k resistors** and solder one end of them together. Solder the other loose ends to the veroboard in the position on the board as shown below:



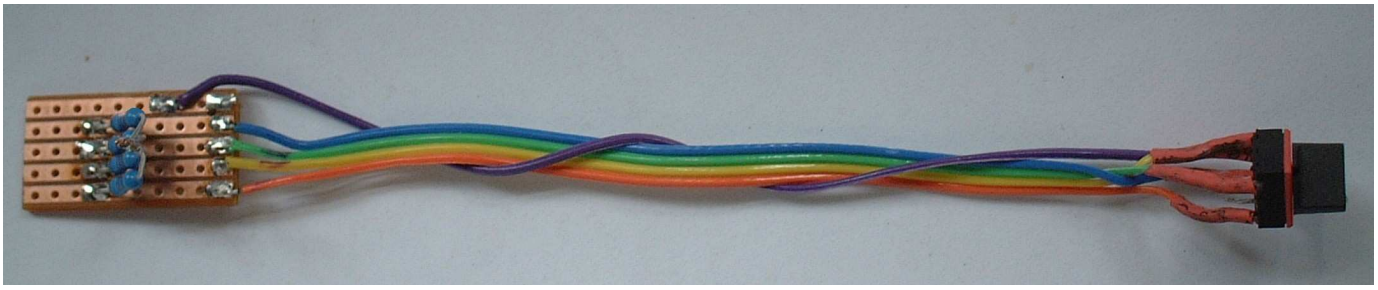


3) Take a 9" length of **rainbow cable** and peel out a section of the colours **blue**-**green**-**yellow**-**orange**. Solder one end of these wires to the veroboard. Solder the other ends to the rotary encoder terminals. It will have its terminals marked with the numbers **1**, **2**, **4** and **8**, and also 2 middle ones marked "**C**".

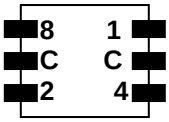
The **blue** wire should go to the terminal marked "**1**".
The **green** wire should go to the terminal marked "**2**".
The **yellow** wire should go to the terminal marked "**4**".
The **orange** wire should go to the terminal marked "**8**".

Solder a **purple** wire to one of the encoder terminals marked "**C**" (it doesn't matter which one).

Use **heat shrink insulation tubing** to cover all the bare soldered terminals of the encoder.



4-bit Rotary Encoder (underside view)



Wiring up the memory chips

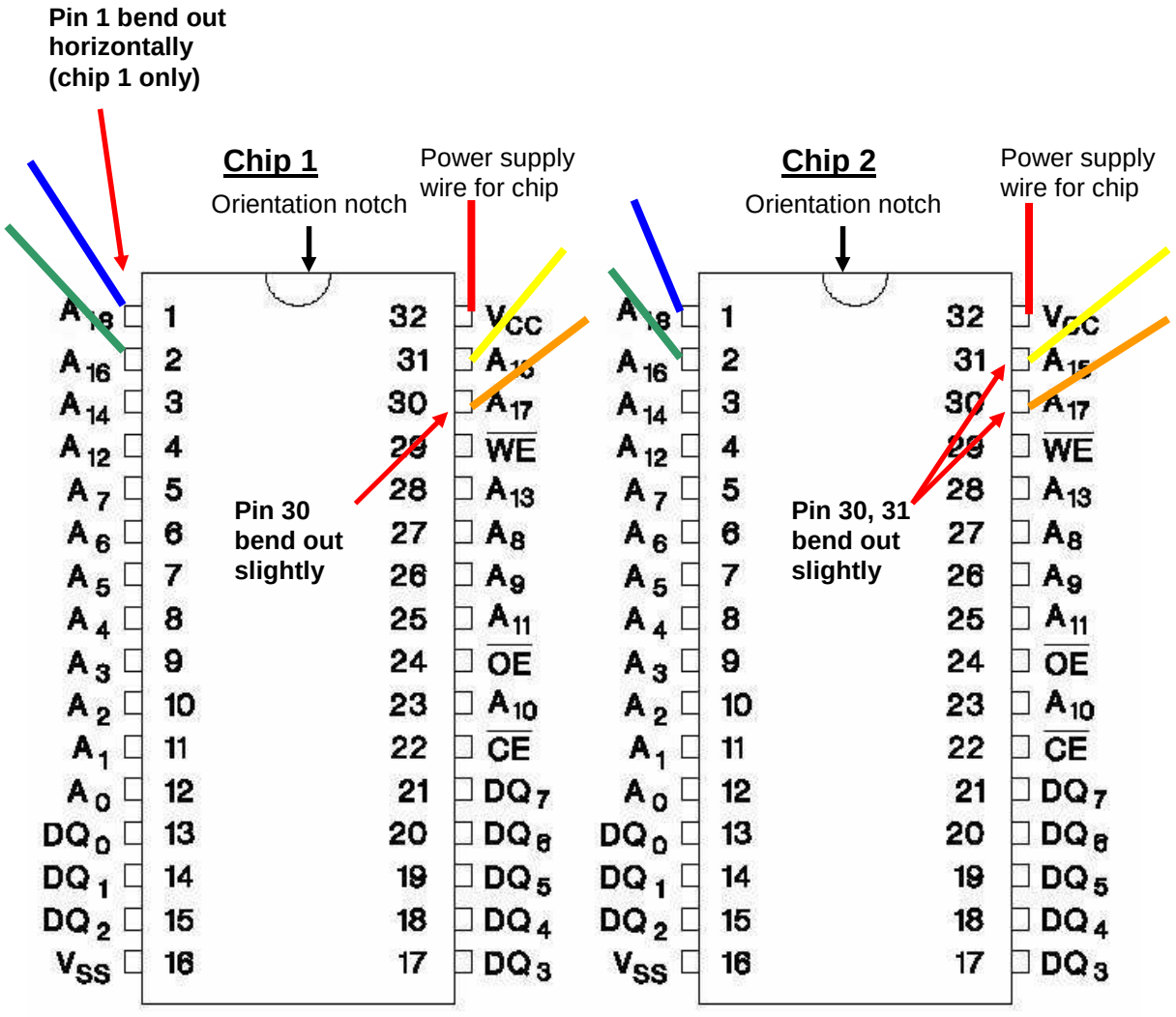
The **top end** of the chip is marked by a small **notch** in the end of the chip.

1) Bend the rows of chip pins slightly **inward** to get them to fit in the socket holes. This is best achieved by placing the chip on a table on its side and rolling the chip over slightly while pressing down on it. Do this to both sides of the chip.

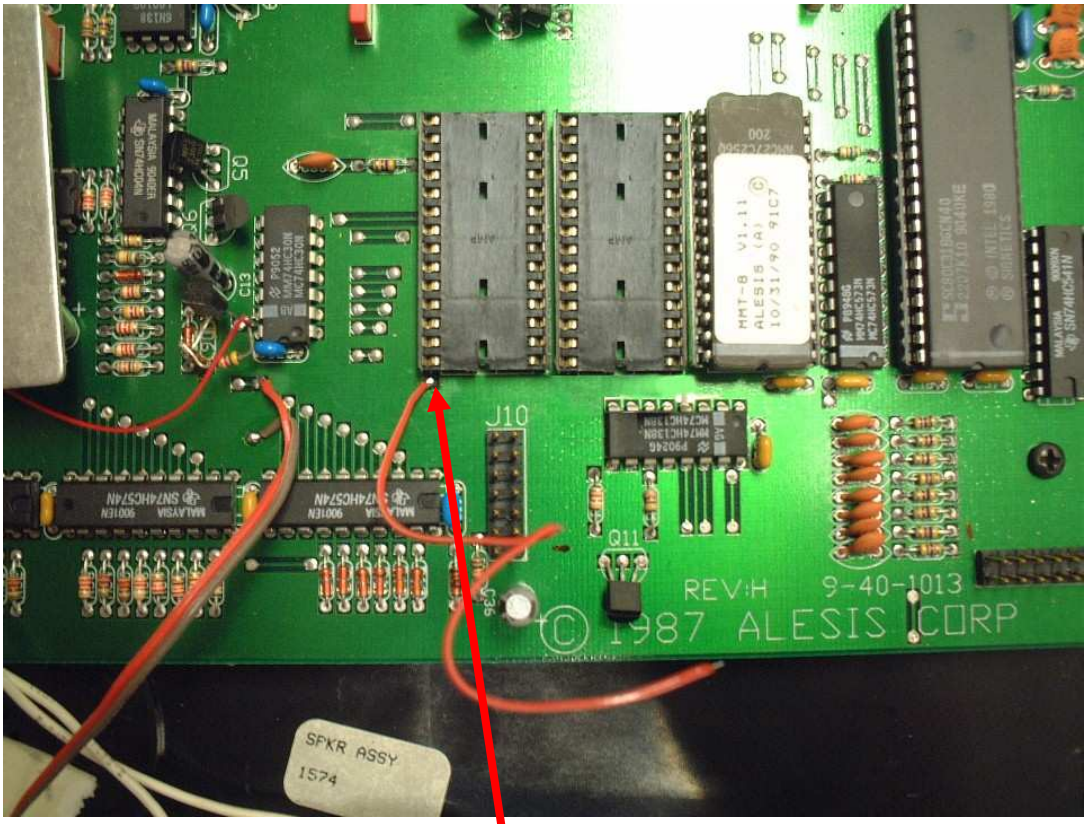
2) Bend out **pin 30** slightly at an angle from vertical on **both chips**. When we insert the chips in the board sockets, we want this pin to **NOT** go into the socket hole, but instead be on the **outside** of the socket.

3) Bend out **pin 31** slightly on **chip 2**, as directly below it on the circuit board is the **+5V supply point** to the chips. We don't want the pin to touch this point.

4) Bend out **pin 1** of **chip 1** completely **horizontally**. This is because on the board, there is a **small chip** directly under where this pin will go. If it is not bent out flat, the pin will hit the top of the chip and prevent the memory chip from seating in its socket.

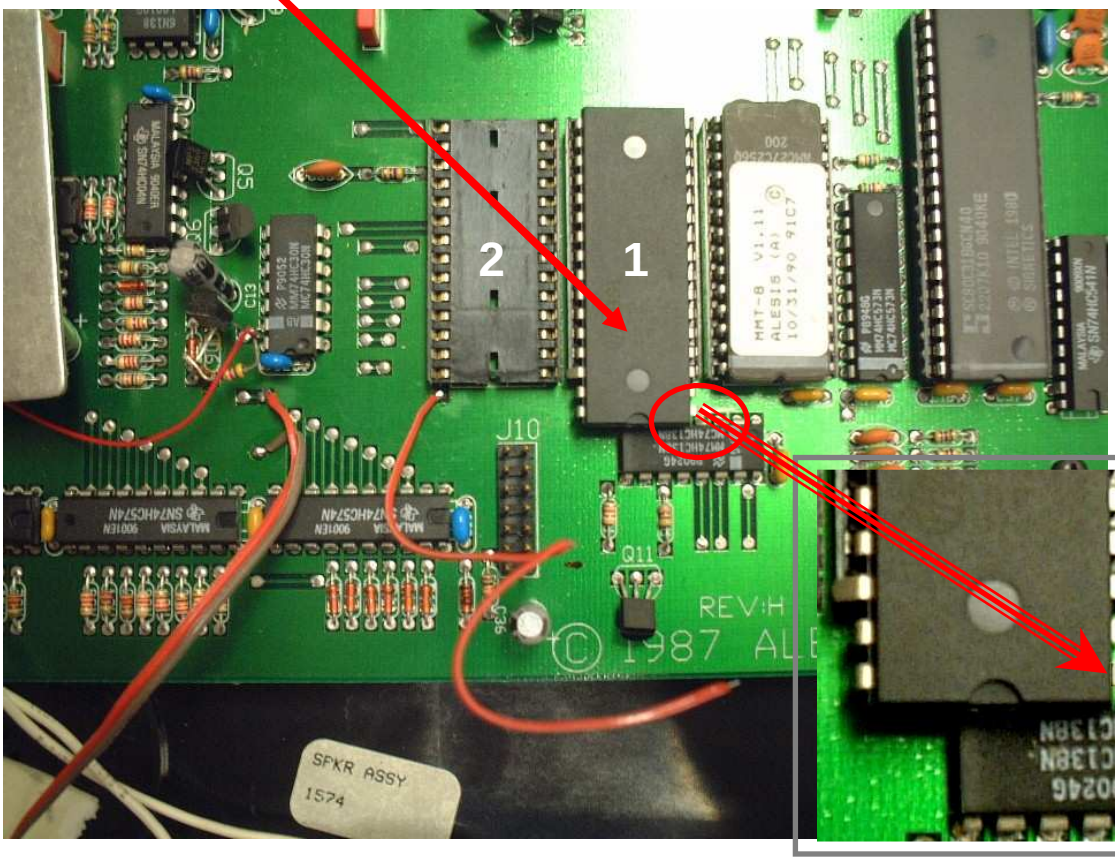


Installing the chips



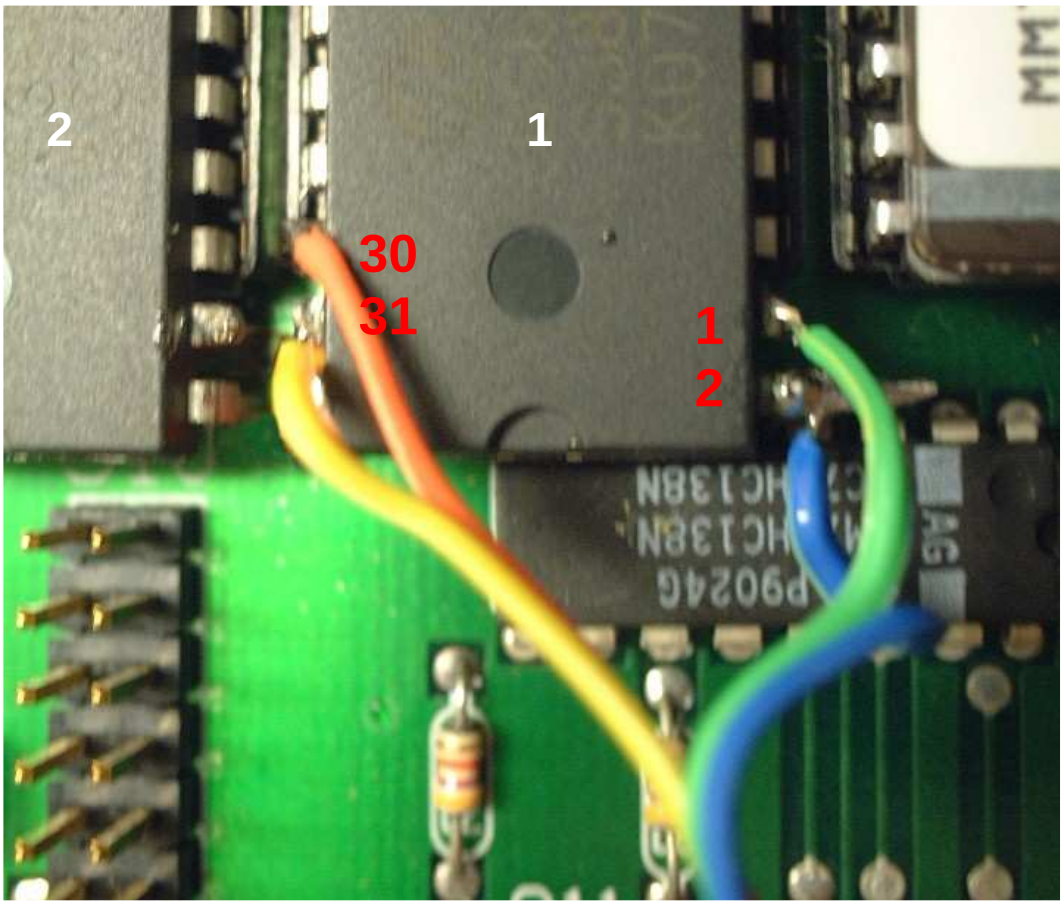
1) Solder a 5” length of **red wire** to the solder blob on the board as shown above. This is the **power supply** wire to the chips.

2) Rest the new **chip 1** on the **right hand socket** to check for alignment of the chip pins. **Pins 1, 2, 31 and 32** should **overhang** the socket.



This chip 1 is the one that has its **pin 1 bent out** completely horizontally

1) **Remove chip 1** from the socket. Solder 3" lengths of **blue, green, yellow,** and **orange** wires to the **top of the shoulders** of chip 1's **pins 1, 2, 30 and 31**. Rest the chip again on top of the **right hand socket** as shown below, with pins **1, 2, 31 and 32** overhanging the socket.



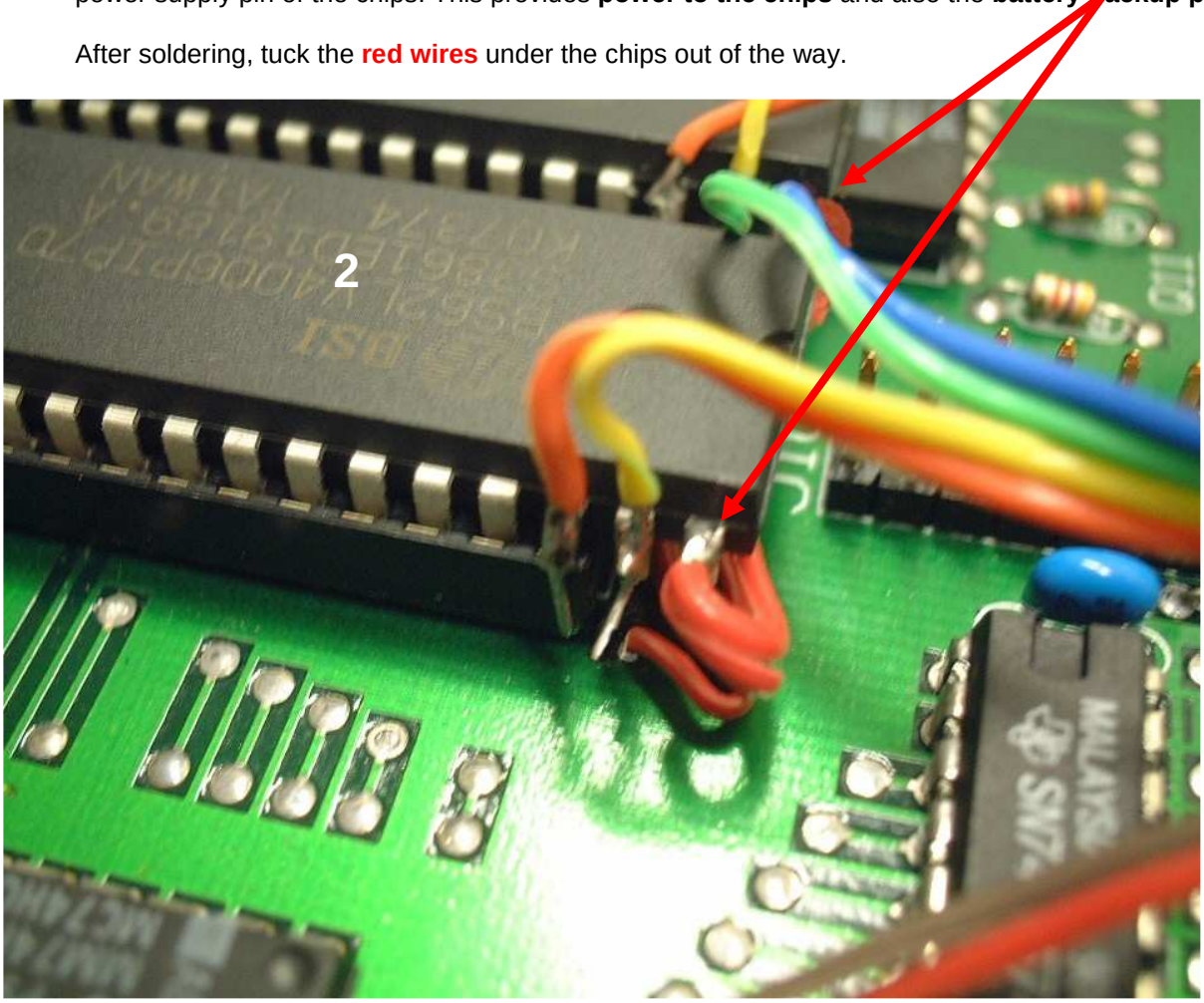
If everything looks ok, and **pin 30** is sitting outside the hole in the socket, **push down firmly** on the chip all over to push it into its socket. You should feel the chip move down in the socket. Check that the chip has properly contacted the **socket top** and that there are no significant gaps all around, and that **pin 30** has gone **outside** the socket. Also check that **pin 1** is not touching the **little chip** on the board. If it is, gently bend the pin up slightly until it is clear of the little chip.

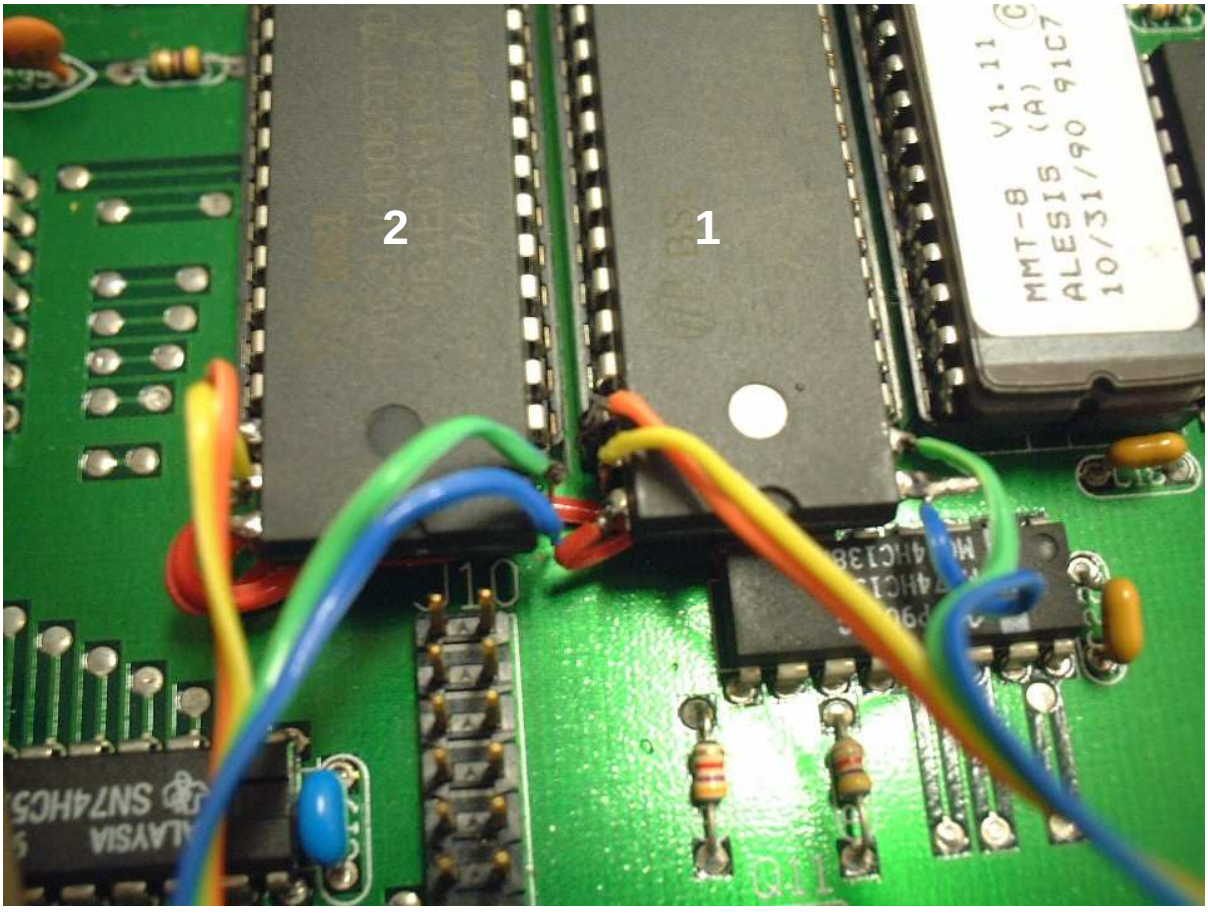
2) Repeat this procedure with the **other chip,** soldering the same coloured wires to the same pins, except that you **don't** have to have **pin 1** bent out flat to clear any chips on the board. Make sure that **pin 31** does not touch the solder blob on the board with the **red wire** attached to it – bend it out a little if it is close to it. And again, make sure **pin 30** is sitting **outside** the socket hole. Push **chip 2** down in the socket as per **chip 1.**

3) Examine the chips in their sockets. Make sure that there are **no chip pins touching** each chip's neighboring chip (especially **pin 30**)

4) Solder the **red power wire** that is soldered to the solder blob on the board to **pin 32** of **both chips**. This is the power supply pin of the chips. This provides **power to the chips** and also the **battery backup power** to them.

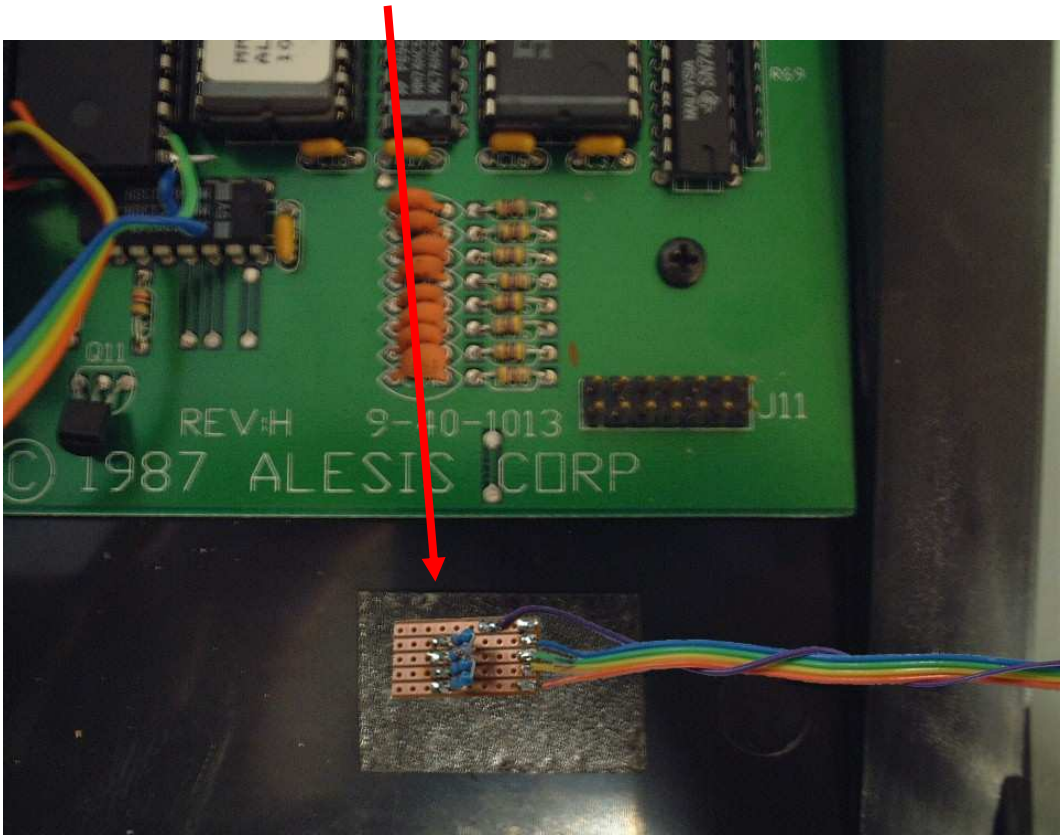
After soldering, tuck the **red wires** under the chips out of the way.



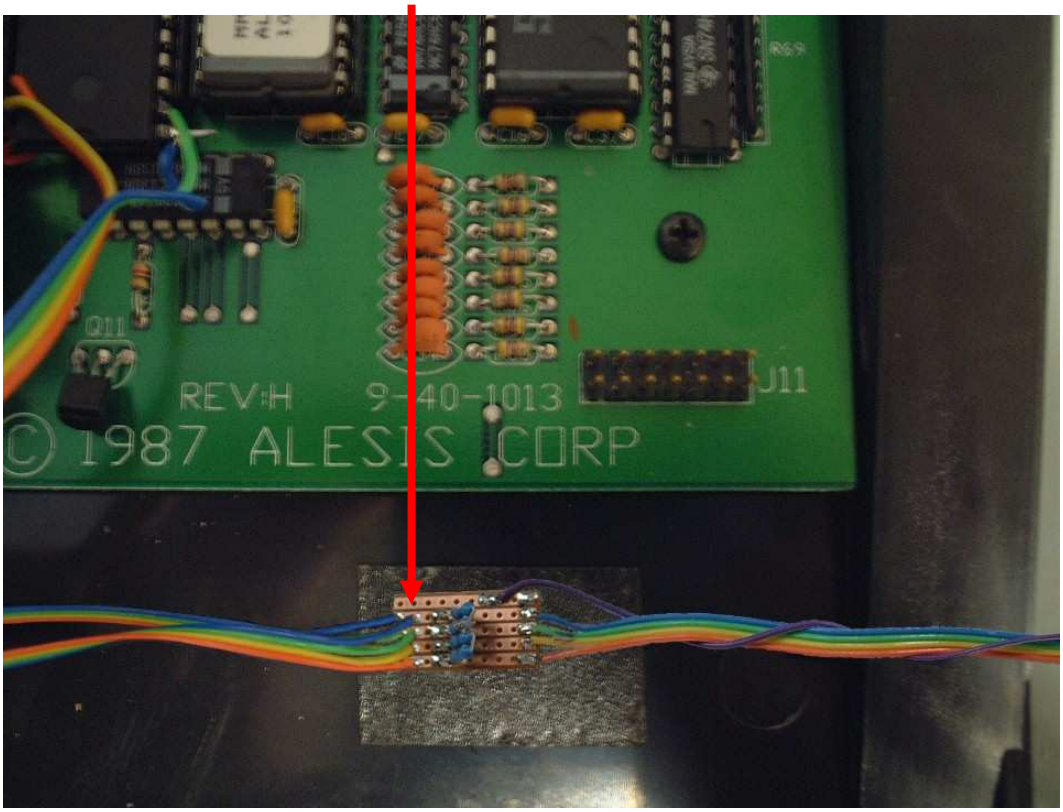


Mounting the encoder board

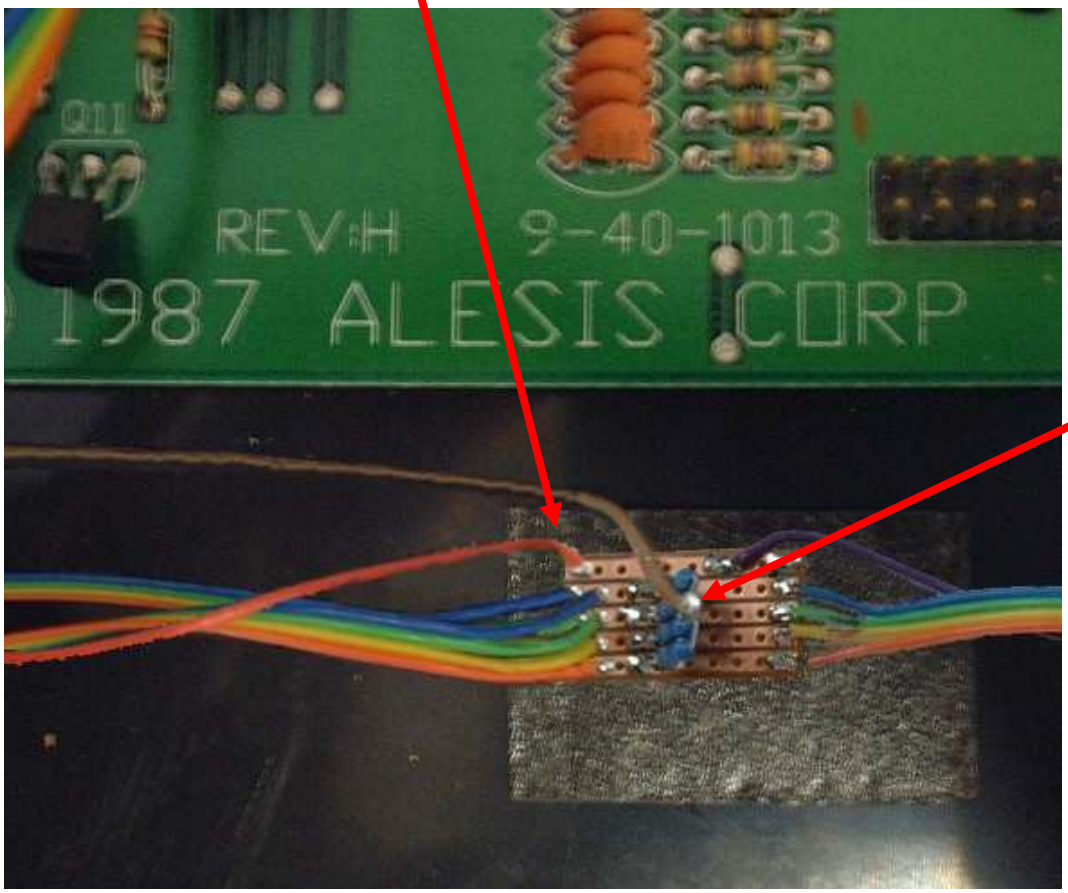
1) Place a small strip of **double sided sticky tape** somewhere convenient in the casing to mount the **encoder board.** Stick the encoder board to it.



2) Now solder each pair of same **coloured wires** from the chip pins to the **encoder board**. Solder them in the colour order as shown. The **same colour wire** from the chips should be on the **same copper strip** that the coloured wires from the encoder are on.



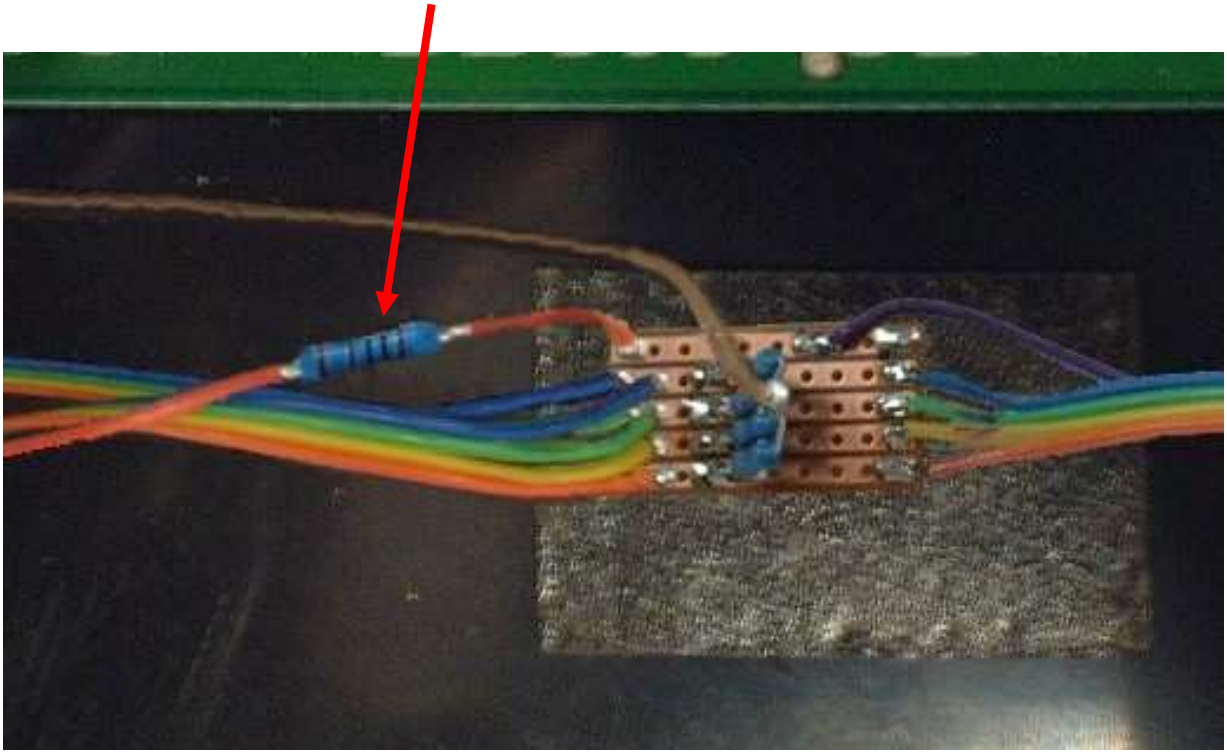
Solder a **red wire** to the **top copper track**.
This is the same copper track that the **purple encoder wire** is soldered on to.



Solder a **brown wire** to the end of the **resistor cluster** as shown.

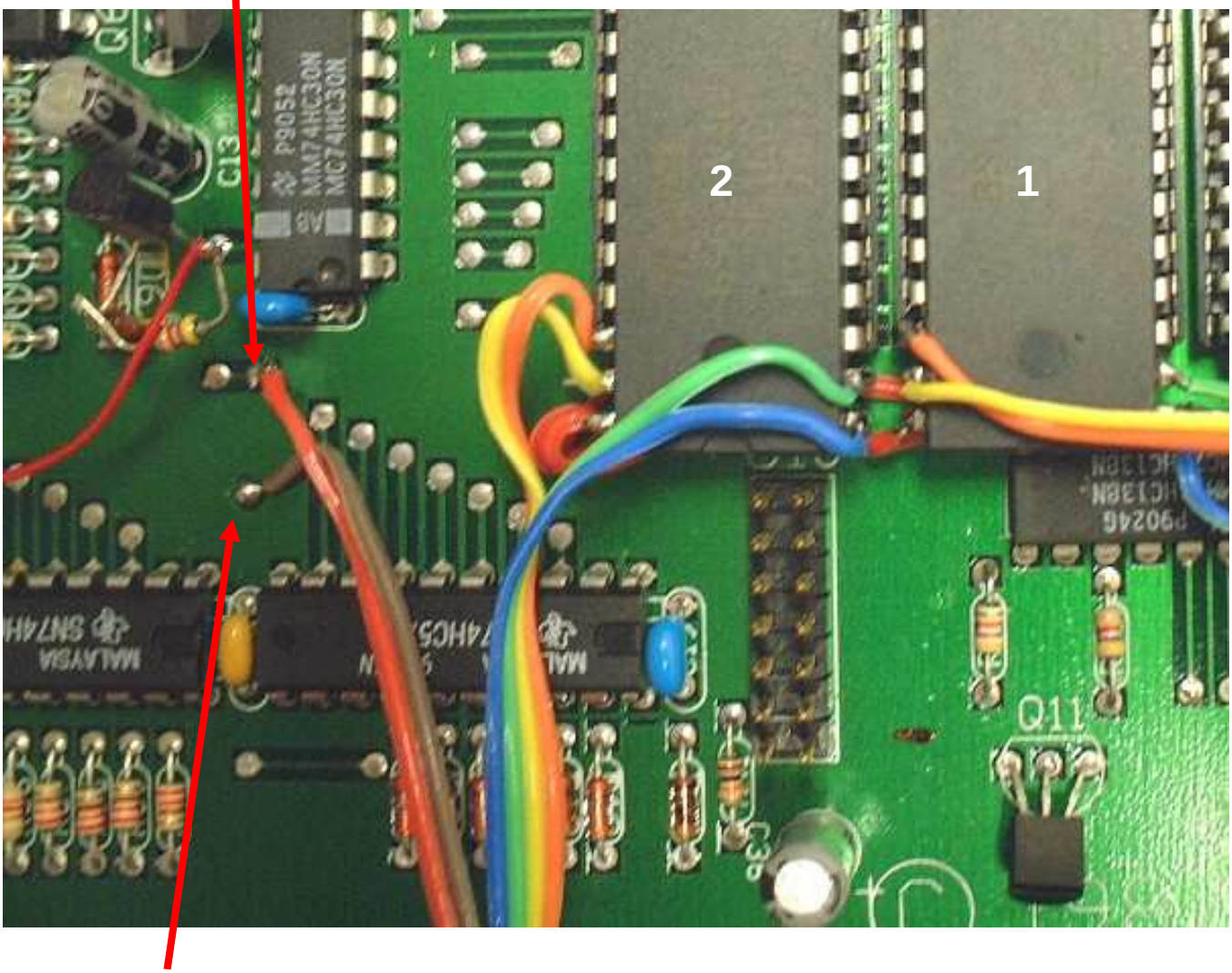
Insulate it with some heat shrink tubing.

Solder a **1k resistor** into the **red wire** as shown. **Insulate** the resistor connections with heat shrink tubing or insulation tape.



The Encoder power supply

Solder the **red wire** from the **encoder board** to this solder blob on the board here.
This is the board's **+5V power supply** point.



Solder the **brown wire** from the encoder board here. This is the board's **ground point**.

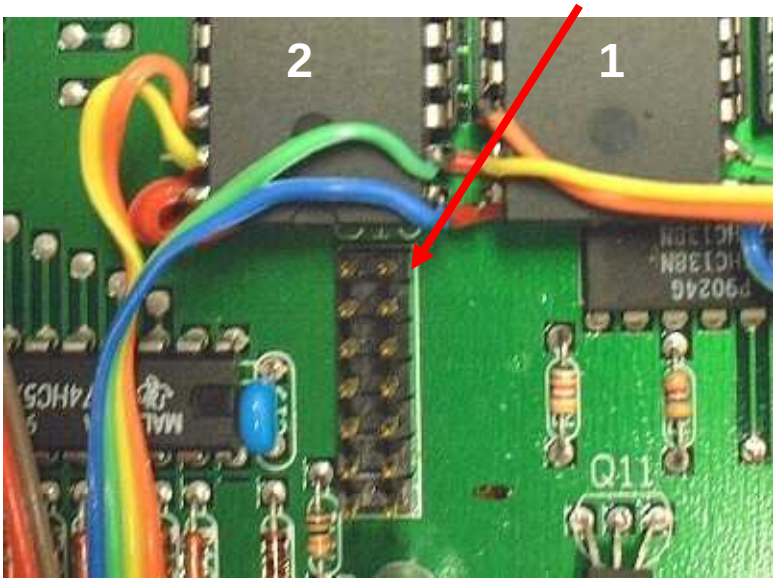
It is simply a small area on the board that you scrape the green insulation away from to expose the copper. You then solder the **brown wire** to it.

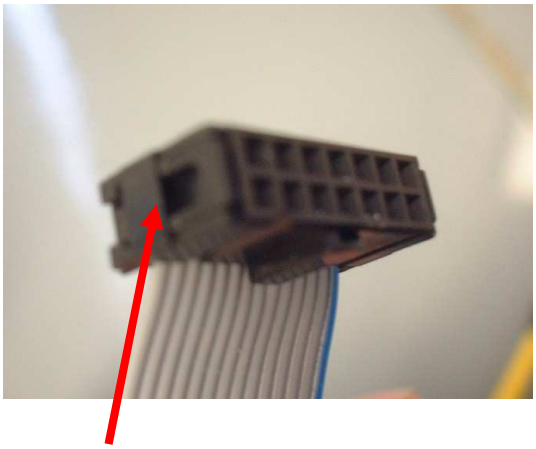
Congratulations! You've finished it!!

Testing the memory expansion and formatting (initializing) the memory

1) **Reconnect your memory battery**. Solder the cut leg back together.

2) Loosely put the casing back on after **reconnecting** the top casing **ribbon connector cables and plugs** to the connectors on the circuit board. One of the **connector plugs** is right next to **memory chip 2**. You will find that you won't be able to get the plug back on, because the chip is in the way:





Take the **connector**, and using **rough grade sandpaper** or a **file**, begin to **file** down the side of the connector that is hitting the chip, until the plug just slips down beside the chip on its connector pins and connects properly. Push the connector down securely.

3) **Dangle the rotary encoder** out of the casing for testing. When you are sure everything is working, you can tidy it up by mounting the encoder somewhere convenient on the front panel, with the wires going through a small hole in the casing somewhere.
Alternatively, you can swap it later for the neat **pushbutton wheel type** encoder mentioned in the parts list. If you use that one, you just cut a small rectangle hole in the casing and it pushes through from the front and mounts neatly and flush with the surface.

4) **Turn the rotary encoder to position "0".** This is the **first** memory position.

Connect the power supply to the MMT8 and do an **initialization** (formatting) of the machine to clear the memory – press and hold the **ERASE, PAGE UP, and PAGE DOWN** buttons when turning on the **power** while **holding** these buttons down for 3 seconds. This will initialize and erase the memory and ensure there is no random data in it.

Plug the MMT8 into a MIDI synth, and try recording a **part**. Record a few parts make a few simple, quick songs.

5) **Switch the machine off, and then back on again**.
It should still have remembered your test recordings. If not, check that you have reconnected your memory battery properly, and that the solder connection is good. Also, check that you have connected the red power wire on the circuit board to **both chip's pin 32.**

6) Switch the machine off, and turn the encoder to **position "1",** which is the **2nd** memory position. Initialize the memory, repeat the procedure, record some tracks. Switch it off. Repeat for the other memory positions.

When you get to encoder position **"15"**, this is the final **16th** memory. If you now keep turning the encoder (it can turn continuously around and around) to position "0", it starts it from scratch again at memory 1, then 2, 3, 4 etc.

Once your machine is working well, you can record away. Mount the encoder somewhere on the casing and put the machine back together. As mentioned, If you get the better **pushbutton encoder** mentioned in the parts list, all you have to do is cut a small rectangle hole in the casing and it fits in it neatly and securely.

The memories in use

The general best practice is to switch the machine off before changing the memory position.

If you **change the memory position** and then hit play **without powering down**, the machine will freak out and crash.

However, I have found that you **CAN** change memory positions without powering down **IF** you don't hit play, but instead, scroll up to the next part which has something in it, and then scroll back to the original one it was on. Now if you hit play, it will play ok.

If you change the memory position and there is **absolutely nothing** in the memory bank you change to and you **haven't initialized** that memory, it will freeze. But if each empty memory **has been initialized** first, as in the testing procedure, then this won't happen at all, and you'll be ok.

So, in summary, if you want to **change memory positions without powering down**, do the **initialize** steps for each memory position **once** to ensure each memory has been formatted, and it will be ok.

Disclaimer

No responsibility will be taken by me for any damage done as direct or indirect result of this modification. Do it at your own risk. Protect your equipment from static by wearing an earthed antistatic wrist band when handling the chips or circuit board.

Graham Meredith 2006
Revised 15th January 2009