MB-6582 Base PCB Construction Guide

Please read the entire construction guide before buying parts or starting to solder anything.

Note that Base PCB revision 1 (the red ones bought directly from me) has a few missing tracks. See the bottom of this page.

I have tried very hard to document and explain the process that I followed while constructing my original MB-6582 and how I suggest people construct their MB-6582. I assume people constructing an MB-6582 base PCB have a basic level of electronics knowledge and some basic soldering skills. You do not need to know exactly how each and every part works, but you must know the very basics of electronics, that electrical current flows along the tracks on the PCB, that all points along that track should measure the same voltage, etc.

So some parts of this guide are specifically to help the newbie that might have done a few simple electronics kits, but might not have soldered these kinds of components before, or know exactly what type of component to buy. So I'll tell you where you could go wrong, and how not to go wrong. I'll tell you the best order to solder things together and how to test things along the way. If you want more information about how it works, you can refer to the MIDIbox module documentation (i.e. the Core, DIN, DOUT and SID modules).

If at any time you do not understand what to do, or the instructions are confusing, post your questions on the MIDIbox forum and I or someone else will help.

PSU Options

There are three different ways of supplying power to the base PCB:

- **PSU Option A:** C64 power supply, generating only 9v supply (no 12v) ... exactly like the prototype, it converts the 9v AC coming from the C64 power supply. This option only allows use of SIDs that need 9v (8580 or 6582), the 6581 needs 12v. One 9v voltage regulator is used to supply 9v to all the SIDs.
- **PSU Option B:** C64 power supply, generating 9v and 12v supply ... almost exactly like the "C64 PSU Optimized" circuit, which adds the 9v DC to the 5v DC to create 14v DC, enough to get the 12v required for 6581 SIDs. Instead of a voltage regulator on each SID module, there are just two (one 9v and one 12v).
- **PSU Option C:** Uses 5v regulation circuit to provide option A or B when not using C64 PSU (which has its own 5V regulation). *I have added this option at the last minute to the PCB as a potential workaround for people unable to get a C64 PSU. People could generate 5v from a single unregulated 9v AC supply (9v AC adapter) or supply separate unregulated 9v AC and unregulated 5v DC supplies (9v and 5v AC adapters), using the same power socket and switch. Please do not assume this idea will work well, it might introduce noise, hum and heat!*
- **PSU Option D:** External regulated power supply circuit, supplying +12v,+9v,+5v and ground ... for those people wanting to use their own power supply, because they don't want to use the C64 PSU.

You must decide at the start which of these PSU options you wish to use, as this affects which components you solder to the PCB.

If you wish to use the PacTec PT-10 case (like the original MB-6582), then you should use a Commodore 64 power supply (i.e. the thing which plugs into a Commodore 64), and you can choose between option A or B.

Option A is preferred if you only have 8580 or 6582 SIDs and never want to use the older, noisier, buggier 6581 SID. Option A will only produce a 9v supply for the SIDs, and the 6581 SID needs 12v.

Option B is for a mixed SID setup; if you have 6581 SIDs, you need 12v and this option will produce 9v and 12v.

Option A and B use the same parts except option B requires two extra voltage regulators, part V2 (7809) and part V3 (7812).

Option C is for people who cannot use a C64 power supply. Simply explained, it is an extra 5v voltage regulator and capacitors (V4, C13, C14) that can be used instead of the 5v voltage regulator inside a C64 power supply. *Please always consider option C as a potential workaround for people without C64 power supplies. All I have done on the PCB is break the normal 5v supply lines so you can reuse the power port and power switch, and provide some space on the PCB for a 5v voltage regulator. I am just giving people some extra options should they get really stuck buying a C64 power supply.*

PSU Option A

Solder a bridge at J71 between pin 2 and 3. (This connects V1 common pin to ground).

Solder a bridge at J72 between pin 2 and 3. (This connects V1 output pin to 9v supply lines).

Solder a bridge at J73 between pin 1 and 2. (This connects 5v from power socket/switch to 5v supply lines).

Solder V1, B1, C1, C2, C3, C4, C11, C12.

Do not solder V2, V3, V4, C13, C14, J74

PSU Option B

Solder a bridge at J71 between pin 1 and 2. (This connects V1 common pin to 5v, so it outputs 14v relative to ground).

Solder a bridge at J72 between pin 1 and 2. (This connects V1 output pin to 14v supply lines, to go into V2 and V3).

Solder a bridge at J73 between pin 1 and 2. (This connects 5v from power socket/switch to 5v supply lines).

Solder V1, B1, V2, V3, C1, C2, C3, C4, C11, C12.

Do not solder V4, C13, C14, J74

V1 cannot share the same heatsink with V2 and V3, as the heatsink mounts are connected to the common pin and V1 common pin is connected to 5V, V2 and V3 common pins are connected to ground

PSU Option C

In options A and B, a 9v AC and a *regulated* 5v DC power supply come through the power socket and switch, both generated by the same transformer in the C64 power supply. If people don't have this power supply, they could connect a 9v AC supply and an additional *unregulated* 5v DC supply through the power socket. In this case, J73 is used to divert it to J74 and through the 5v regulator and then back to J73 and the 5v supply lines. *In theory,* this will allow option A or B to work, and can generate 9v and 12v supplies. Alternately, people could connect a single 9v AC power supply, then connect the regulated 9v DC supply at J4:pin3 to J74:pin2, then connect the 5v DC output at J74:pin1 to J4:pin2. *In theory,* this will allow option A to work, but option B would not be possible because the 5v is generated from the 9v supply, it can't be *added to* the 9v to make the 14v required in option B.

People choosing this solution can work out for themselves what extra wiring, parts and solder bridges are needed, or they can ask me for help.

PSU Option D

Connect power from your external power supply at J4. If you are only using 8580/6582 SIDs, then 12v is not required.

If you are using PSU Option D, then you probably know enough about electronics to need no further help, but I will specify the following anyway:

Do not solder V1, V2, V3, V4, B1, C1, C2, C3, C4, C11, C12, C13, C14, J1, J74, S1

Do not solder bridges at J71, J72, J73

If you want, you can reuse C3 and C4 as extra smoothing capacitors on the 5v supply lines, in which case bridge J73.

Socket/Switch Options

If you are constructing a base PCB to go with the PacTec PT-10 case like the original MB-6582 (or want to use the PCB mounted sockets and switch with a different case), then you will need to solder the PCB mounted sockets and switch at the top of the PCB. So you will need to solder S1, J1, J12, J13, J21, J22, J23, J24. Although fairly obvious, I'll explicitly tell you… **Do not solder J1A, J12A, J13A, J21B, J22B, J23B, J24B.**

If you are choosing not to use the PCB mounted sockets and switch on the base PCB (e.g. panel mounted or separate PCB mounted) then you will need to connect those components to the

base PCB with wires. In this situation, pads for SIL headers are provided. Solder headers to J12A, J13A, J21B, J22B, J23B, J24B. Solder header to J1A if using PSU Option A or B. Always use headers and connectors, this will let you adjust wire lengths after attaching the base PCB to the case, and is a more professional look anyway.

If you are choosing not to use the PCB mounted sockets and switch on the base PCB and if you are using PSU Option A or B, then you need to wire a DPDT switch *between the pins of your 7-pin DIN socket and J1A* and for convenience, refer to the "C64 PSU Optimized" circuit as a guide. If in doubt, use your multimeter to test the voltages coming out of your 7-pin DIN socket. Remember to always use headers and connectors, do not solder wires directly to the PCB. You will also need to bridge the pads of the PCB mounted switch S1, so power supplied at J1A goes through the pads where the switch would be mounted normally, therefore solder bridge between pins 1 and 2 of S1 and between pins 4 and 5 of S1 to simulate an “always on” switch in S1.

If you are choosing not to use the PCB mounted sockets and switch on the base PCB and if you are using PSU Option C, then you will be following the same steps for PSU Option A or B, and supplying either 9v AC, or 9v AC and 5v DC, to J1A via a switch, therefore solder bridge between pins 1 and 2 of S1 and between pins 4 and 5 of S1 to simulate an “always on” switch in S1.

If you are choosing not to use the PCB mounted sockets and switch on the base PCB and if you are using PSU Option D, then you need no further help, you can ignore J1A, and thus have no need to bridge pins of S1. Supply 12V/9V/5V/ground at J4 and make sure your switch will cut power to 12V/9V/5V pins (i.e. it is not good to be only turning off the 5V supply).

Soldering

Whether you're a newbie or veteran at soldering, you should still have a look at soldering guides and other information about soldering, so you can learn how to do it well, or confirm you've been doing it well all along.

I recently came across these guides, which are really well presented:

- http://www.curiousinventor.com/guides/How_To_Solder
- http://www.sparkfun.com/commerce/present.php?p=BEE-6-SolderingBasics

If you are new to soldering the biggest advice I can give is buy the thinnest solder you can, get a temperature controlled soldering iron with a fine tip, and get a desoldering pump (the cheap kind, not a motor-driven thing). Even if you can't afford a fancy soldering iron, get the desoldering pump to fix your soldering mistakes (i.e. too much solder on a pad!).

Soldering requires three hands; one to hold the part, one to hold the soldering iron and one to hold the solder and push it into the joint. Unfortunately, people only have two. There are a few techniques which I use to compensate for the missing third hand.

Tacking

Tacking is what I call temporarily attaching the component to the PCB so you can then use two hands to solder the other joints while the component is in place. The process is as follows:

- Put the component in the PCB and hold it in place with one finger. Do not trim the leads!
- Pick up a blob of solder with the soldering iron.
- Push blob of solder into one joint of the component on the bottom side.

This will leave a messy, unfinished joint, just a little blob on the side of the lead sticking it to the pad, it might even be weak enough to break if you move the component too much, but it is enough to hold the part in place while the PCB is upside down. Then it is time to finish soldering this component, as follows:

- Solder *the other* lead properly by putting the tip of the iron right in the joint, so it touches both the pad and the lead.
- Insert solder from the opposite side, when the pad and lead have heated enough, it will just start melting as you insert more solder.
- Stop inserting solder when there's enough solder to cover the pad. Note that solder will get sucked into the hole, keep inserting until a little cone of solder forms, and the entire pad is covered with solder.
- Repeat these steps to solder the other leads of that component.
- Do not forget to properly solder the joint you “tacked”. Melt the solder from the tack and insert more solder.
- Trim all the leads as close to the joint as possible. Cut just the lead, do not cut into the solder. You want just the cones of solder on the PCB.

This technique goes against the advice given in other guides, that you should **never** carry a blob of solder to the joint. You could just as easily bend the leads a little to hold the part in place (at least for parts with bendable leads). However, if the tacked joint is fully reheated with the iron and replaced with a proper solder joint, I think there's nothing wrong with this technique, and it's not much different to tinning wire (adding solder to the end of wire) before it is soldered to a terminal.

More advice

- It is a good idea to tack a component to the PCB and then turn the PCB over and check if it is placed straight and also placed flat onto the PCB.
- Don't touch the component close to the lead being soldered. Ideally, tacking is a quick process and the part won't get hot enough.
- Solder mask stops solder from sticking where it shouldn't be. You can practically smear a big blob of solder over a pad and it will stick to just the pad.
- Tacking joints on opposite sides/corners of IC sockets will hold the socket better than just one tacked joint.
- Pads connected to the ground plane (or to wide tracks) take longer to solder because the heat dissipates away from the joint easily. They are therefore much harder to tack, so choose a different joint to tack (i.e. choose the pad which has a ring around it, not the pad which has no ring).

Desoldering

Avoid at all costs the need to desolder a component. If you must desolder a part, it is probably safest to cut it off on the top side, leaving enough to pull the lead out *from the top side* while the iron is on the joint on the bottom side. This will avoid ripping the pad off the PCB. Then after, suck up the solder on the bottom side. The reason for this is that you can easily heat up all the solder in the joint, if you try to pull out the part after sucking up solder, then there's less solder, poorer conductivity, it's not all fluid, etc.

Soldering Order

Please read carefully any advice about specific components before you start soldering.

For the base PCB, I recommend the following order of soldering (essentially shortest-to-tallest):

- All resistors that are mounted flat **(except R40 to R55, solder these at time of control surface construction!)**
- All diodes (there are 5 on the base PCB).
- Capacitors that go inside an IC socket (each SID has one, each PIC has one). These should be mounted flat, put a 90 degree bend in the leads before mounting.
- All IC sockets. Depending on the manufacturer, you might need to trim some plastic off them to make room for a resistor.
- All headers.
- All other components, in sections, with testing after each section is complete.

Doing it in this order will allow you to lie the PCB upside-down and flat on your work area and hold the components close to the PCB while you solder. You can then solder all those components with the PCB upside-down, not needing to turn it over to put new components in. A lot of time spent soldering is inserting a component, flipping the PCB, picking up/putting down the soldering iron, etc.

To stop components all falling out while you turn the PCB upside-down, you can bend the leads a little, but once the PCB is flat on the work area, it is best to bend the leads back to vertical again, so your solder joint forms a nice cone. It is also a good idea for resistors etc. to trim all the leads to 5mm after the PCB is upside-down, so there is room to put the soldering iron on the joints.

If you are lucky to have two base PCBs, you can use one of them to hold a lot of components (especially headers) in place while you flip the PCB over and then solder a lot in one go. In this case, I choose to solder headers first.

After you have done these components, you can then solder the other components, but I advise completing the rest in sections and then testing each section separately, to make trouble-shooting easier:

- Sockets/switch at top of PCB.
- PSU section
- Core sections
- SID sections

Even if you only have one or two SIDs, I highly recommend you solder all four Core and all four SID sections and test them all. You might get new SIDs in the future and it would be a lot easier to have the other sections already soldered and ready to insert SIDs.

Headers

For the newbies: SIL (single-row inline) headers have one row of pins. DIL (double-row inline) headers have two rows. They are those strips of gold pins held together with black plastic, that should be mounted with the plastic part and longer pins on top.

This is how I cut up the SIL and DIL header strips that came with SmashTV's parts kit:

- DIL header: J15_COREx (4 of 8×2), J11 (1 of 4×2), Spare 4×2
- DIL header: J6_COREx (4 of 5×2), J8_COREx/J9_COREx (4 of 5×2)
- SIL header: J5_COREx (4 of 10×1)
- SIL header: J4_COREx (4 of 4×1), J3_SIDx (4 of 3×1), J23_SIDx (4 of 3×1)
- SIL header: J1_SIDx (4 of 3×1), J2_SIDx (4 of 3×1), J70 (1 of 3×1), J2 (1 of 2×1), J3 (1 of 2×1), J25 (1 of 2×1), J4 (1 of 4×1), Spare 3×1
- SIL header: J11_COREx (4 of 2×1), Spare 32×1

If you want, fill J8 and J9 (beneath J9_CORE4) with two spare SIL headers, or from your own stock of DIL headers.

Most headers are optional. I put all of them in for completeness, to assist in testing and troubleshooting, and in case I ever need them for future expansions.

The absolute bare minimum headers you should put in are:

- J11 (for switching which Core is connected to MIDI Out)
- J15_CORE1 (for master Core’s LCD)
- J3_SIDx, J23_SIDx (for connecting feedback pots, or at least grounding the SID audio inputs)

Then the list of useful headers for a “default” MB-6582 setup (no expansion port usage):

- J70 (for passive mixed output, to connect to headphone jack)
- J2 (for power LED)
- J25 (~9V-11V DC, for connecting to fan)
- J3 (5V DC, for connecting to fan, if 9V-11V makes the fan too noisy)
- J1_SIDx, J2_SIDx (for voltage switching between 9V and 12V per SID, see PSU Option B and D)

Then for the advanced users, perhaps wanting to connect things via the expansion port:

- J6_COREx, J7_COREx (for analog outputs via AOUT modules)
- J5_COREx (for analog inputs)

Then there's headers for things like extending the DIN/DOUT chain, or connecting things via I2C, or redundant LCD ports. If you know enough to want them, you don't need me explaining where they

are.

I've only identified which headers are optional for completeness of documentation. **Just put them all in, except ones ending in A or B (these are alternatives to a socket)**

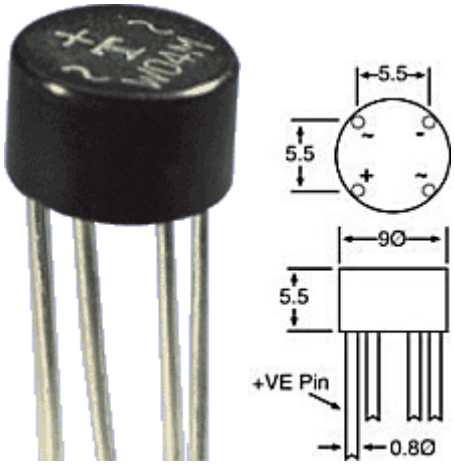
Advice About Specific Components

Please pay close attention to this advice.

Bridge Rectifier B1

The bridge rectifier is a four lead component, two leads connect to AC current (marked with a wavy line "~" symbol) and the other two leads are positive (+) and negative (-), which in this case will output 9V DC and ground. The bridge rectifier that I use, and which the PCB expects, has the positive (+) and negative (-) leads in diagonally opposite corners. Please pay attention and make sure your bridge rectifier has the same lead markings in the same positions as the PCB, and that you put it in the right way.

The bridge rectifier I use is generally identified as "W04 400V 1.2 Amp" and looks like this:



Resistors R30 to R39 (resistor networks for DIN modules)

These resistor networks can be 5-pin or 6-pin (i.e. 4 resistors with common pin, or 5 resistors with common pin). The common pin is marked with a dot and should go in the hole with the square pad (the pin with the part name above/below it). There is a box surrounding 5 pins, it is OK to put in a 6 pin resistor network with one pin going into the hole outside this box, i.e. **DO NOT CUT THE LEAD OFF THE 6th PIN**

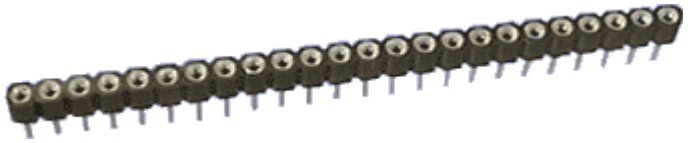
Resistors R40 to R55 (resistors between DOUT and control surface LEDs)

Solder resistors R40 to R55 at time of control surface construction. This is because these resistors need to match the brightness of the LEDs you choose to use in your control surface. People who are joining bulk orders for control surface components (like LEDs) are advised to wait until they have the LEDs in their possession and can test the actual brightness of the LED with a range of resistor values (like 220 Ohms, 1k, 2k, 3k). A lot of people will get a bulk order of superbright or ultrabright LEDs from eBay which are in the 1000 to 10000 mcd range, and using 220 Ohms will put far too much current through the LEDs and they will be way too bright. Unless you want to use your control surface as a LED flashlight, you will need to increase the resistor values to decrease the current through the LEDs and thus decrease the brightness. So rather than solder 220 Ohm resistors and discover you'll need to desolder those resistors and potentially damage the pads and tracks (if it can happen to me, it can happen to you!), it is much better to experiment with one LED and some other resistor values. I discovered the hard way that the 220 Ohm resistors were way too bright with 4000 mcd blue LEDs, even when each LED was multiplexed in the matrix and only lit 1/8th the time. I eventually settled with 3k. You might find 220 Ohms works well with common 3mm LEDs of 100 mcd brightness, or 1k works well with 500 to 1000 mcd LEDs. **For the ultimate in reconfigurability, use machine pin socket strips in the pads where the R40 to R55 resistors go, so you can easily experiment with different resistor values after connecting the control surface.**

Capacitors C1_SIDx, C2_SIDx, C21_SIDx, C22_SIDx (SID filter capacitors)

These are the filter capacitors for the SID. For SID 6581 they should be 470pF. For SID 8580 or SID 6582 they should be 22nF. Since it is likely you might change these capacitors to suit the type of SID, or just for different filter results, you might decide to copy what I do and **not solder these capacitors to the PCB** but instead solder machined pin socket strips to these pads. I use a strip of six pins, and cut the tips of the 2nd and 5th pins so it fits in the PCB where there are no holes. Then I can insert and remove capacitors without soldering and desoldering. Machined pin sockets have gold contacts and are the same or better than the contacts between the SID and the SID's IC socket, so there is so little difference to soldering them in directly that there's no reason not to use sockets. If you are having trouble finding machined pin sockets in strips, you can cut a machined pin IC socket in half, or even cut a cheap dual-wipe IC socket in half.

This is what a machined pin socket strip looks like:

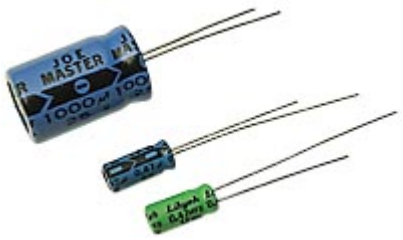


Electrolytic Capacitors

Polarized capacitors (typically electrolytic capacitors) are the cylinder shaped capacitors with values in the microfarad range (i.e. 1µF, 10µF, 100µF, 2200µF). They have positive (+) and negative (-)

leads. The hole/pad for the positive (+) lead is marked on the PCB, but typically electrolytics only have the negative (-) lead marked. So put that lead in the hole that is **not marked with a +**

Here are some electrolytic capacitors with the negative (-) lead marked with a strip of arrows and the "-" symbol:



Diodes

Diodes have an anode and a cathode. Most diodes have the cathode marked with a line. Diodes on the PCB are marked with the standard symbol for a diode, the arrow points to the hole/pad where the cathode goes. So you put the diode with the cathode mark matching the point of the arrow. **Unfortunately the diodes inside each PIC socket (D1_CORE1, D1_CORE2, D1_CORE3, D1_CORE4) were placed over a via (hole) so the symbol on the PCB is obscured. These diode symbols point down, the horizontal white line that is visible is the cathode end, put the cathode in the bottom hole.**

ICs

All ICs on the base PCB should use IC sockets and **should not be soldered directly to the PCB.** ICs have a little notch at the "top" end or a little dot next to pin 1 (the "top left" corner). IC sockets also have a little notch at one end. The outline where IC sockets should go on the PCB also has a little notch. Align the notch of the IC socket with the notch on the PCB. When you are ready to insert ICs (after you have tested voltages on every IC socket), align the notch (or little dot) with the notch of the IC socket.

Completing the Base PCB in Stages

Since this PCB combines multiple MIDIbox modules, if something should not work first time, it is much harder to troubleshoot (debug) because you cannot disconnect the modules. So it is perhaps best to construct the base PCB in stages and test after each stage.

If you do not understand basic electronics concepts like "testing pins are connected" or "test for break in continuity" or "testing voltage of pin X relative to pin Y" or do not know how to use a multimeter to make it beep to test connected pins or test voltages, then this project is a little too advanced for your current level of knowledge. You will need to learn more about electronics and ask questions on the forum. People on the forum are happy to

help and teach, but you must also be willing to learn.

As mentioned before, soldering in a shortest-to-tallest order lets you do soldering in batches, so you should still use this soldering order also. So to be more specific about the order to solder, here's a more comprehensive list:

- All resistors that are mounted flat **(except R40 to R55)**
- All diodes (there are 5 on the base PCB).
- Capacitors that go inside an IC socket. These should be mounted flat, put a 90 degree bend in the leads before mounting.
- All IC sockets. Depending on the manufacturer, you might need to trim some plastic off them to make room for a resistor.
- All headers.

Now after soldering all those, you can start soldering together the PSU section. Go back to the start of this guide and follow the soldering instructions for the PSU option you want to use. Then follow the soldering instructions for the power socket and power switch for that PSU option.

- Test that the power rails and ground are not connected (shorted). You can test this at J4.
- Test that each of the following IC socket's pins are correctly connected to the 5V rail:

SIDs: test pin 25 (fourth from top right); COREs: test pin 11 (eleventh from top left); 74HC595/74HC165s: test pin 16 (first from top right) Optocoupler 6N138: test pin 8 (first from top right)

- Test that each of the following IC socket's pins are correctly connected to ground:

SIDs: test pin 14 (bottom left) COREs: test pin 12 (twelfth from top left) 74HC595/74HC165s: test pin 8 (bottom left) Optocoupler 6N138: test pin 5 (bottom right)

You should now be ready to plug in your power supply and turn on the power switch. It is important that you get your multimeter ready to test the voltages at J4, and test them quickly after turning on the switch, so if you don't see the right voltages, then you can turn it off quickly and avoid damaging components. If you've put the bridge rectifier and capacitors in the correct way and tested for short circuits across the power rails, then you should not have a problem.

Before connecting power to the power socket:

- Check you've completely followed the soldering instructions for the PSU option of your choice, especially soldering the bridge at J71, J72 and J73 (if instructed to do so).
- Check the orientation of J1A (if you are using a panel-mount socket to connect a C64 PSU). It is a good idea to test the voltages coming out of the C64 PSU first, to make sure you really have 5V and ground connected to the right pins! Ideally you should use a *polarized header and connector* for J1A to avoid swapping 5V and ground and destroying components!

Then connect the power, turn on the switch and:

- Check DC voltage between 5V and ground at J3.
- If using PSU Option A or B, check voltage at J25 (should be above 9V, around 11V depending on load).
- Check output of V1 by checking voltage across C12 (bottom pin should be 9V relative to top pin).
- Check 9V pin of J1_SID1, J2_SID1 relative to ground (at ground pin of J3).

- If using PSU Option B, check 12V pin of J1_SID1, J2_SID1 relative to ground (at ground pin of J3).
- **Insert jumper into J1_SID1, J2_SID1 depending on type of SID - 9V for 8580/6582, 12V for 6581**
- Check voltage between pin 28 of SID socket (top right corner) relative to ground.

If these voltage checks are passed, then you have finished the PSU section and can continue. If these voltage checks fail, disconnect power immediately. You will have to work out what is wrong by testing connnections. Start with testing pins on the C64 PSU, then checking for any breaks in continuity between the power socket and the rest of the circuit, i.e. you can test if the switch is working and pins 1&2 and 4&5 are connected when the switch is on, etc.

Assemble the SID modules.

Before inserting SIDs and uploading MB-SID V2, you might want to test the audio buffer circuits of the SID modules, and then get the SID to play a 1khz test tone.

Upload the testtone app: http://www.ucapps.de/mios/mbsid_testtone_v3b.zip

The PIC is generating a 1khz square wave going to pin 8, which is normally used during writing new data to the SID. With the SID out of the socket, you can bridge SID socket pin 8 with 27 with a bit of wire or a resistor (anything less than 1K should work I think), and make this 1khz square wave go through the audio buffer and out the audio outputs and into a mixer/amp. Thus you can test the audio buffer without a SID. Then, when this works, you can remove the wire/resistor, power off, insert the SID and power on. Now the SID should play the same 1khz tone, this time because the PIC is sending the SID data and it is internally generating the 1khz tone and outputting it through pin 27.

For people experienced enough to fully populate the PCB without any more help, I'll quickly note some extra info here:

The J3_SIDx and J23_SIDx headers on my MB-6582 are for connecting the audio input/output to dual-gang 500k pots for feedback into the SID and creating cool sounds (distortion or resonance boost or self-oscillating filter!). You could instead choose to connect audio input jacks for filtering external audio (pref. with a switched jack so that the input is grounded if there's no plug in it). If you do neither, you really should use a jumper to connect the audio input (left pin) to ground (middle pin) or you will get lots of noise in your SID output.

J11 at the bottom of the board is used to switch which PIC's serial output is connected to the MIDI Out. You use this during uploading of MIOS and the MB-SID V2 firmware to each PIC. Place a jumper vertically across two pins to switch between the PICs 1,2,3,4 (1 is the master, on the right). Leave the jumper in position 1 when you've finished. All PICs receive the same MIDI In, which is OK during uploads and the MB-SID V2 firmware also requires them all connected to MIDI In. I've labeled this 1,2,3,4 and the Cores/PICs are identified 1,2,3,4 but the device IDs you burn into the PIC are 0,1,2,3.

Don't forget R80. This resistor is required for the CAN Bus to function. If you populated all the Cores and they aren't working, then you probably forgot R80.

Finished?

Maybe not. Go read the notes at the end of this page:
http://www.midibox.org/dokuwiki/wilba_mb_6582

Base PCB Revision 1 Errors

http://www.midibox.org/forum/index.php?topic=9781

http://www.midibox.org/forum/index.php?topic=9775

From:
https://wiki.midibox.org/ - **MIDIbox**

Permanent link:
https://wiki.midibox.org/doku.php?id=wilba_mb_6582_base_pcb_construction_guide&rev=1255980780

Last update: **2009/10/19 19:33**