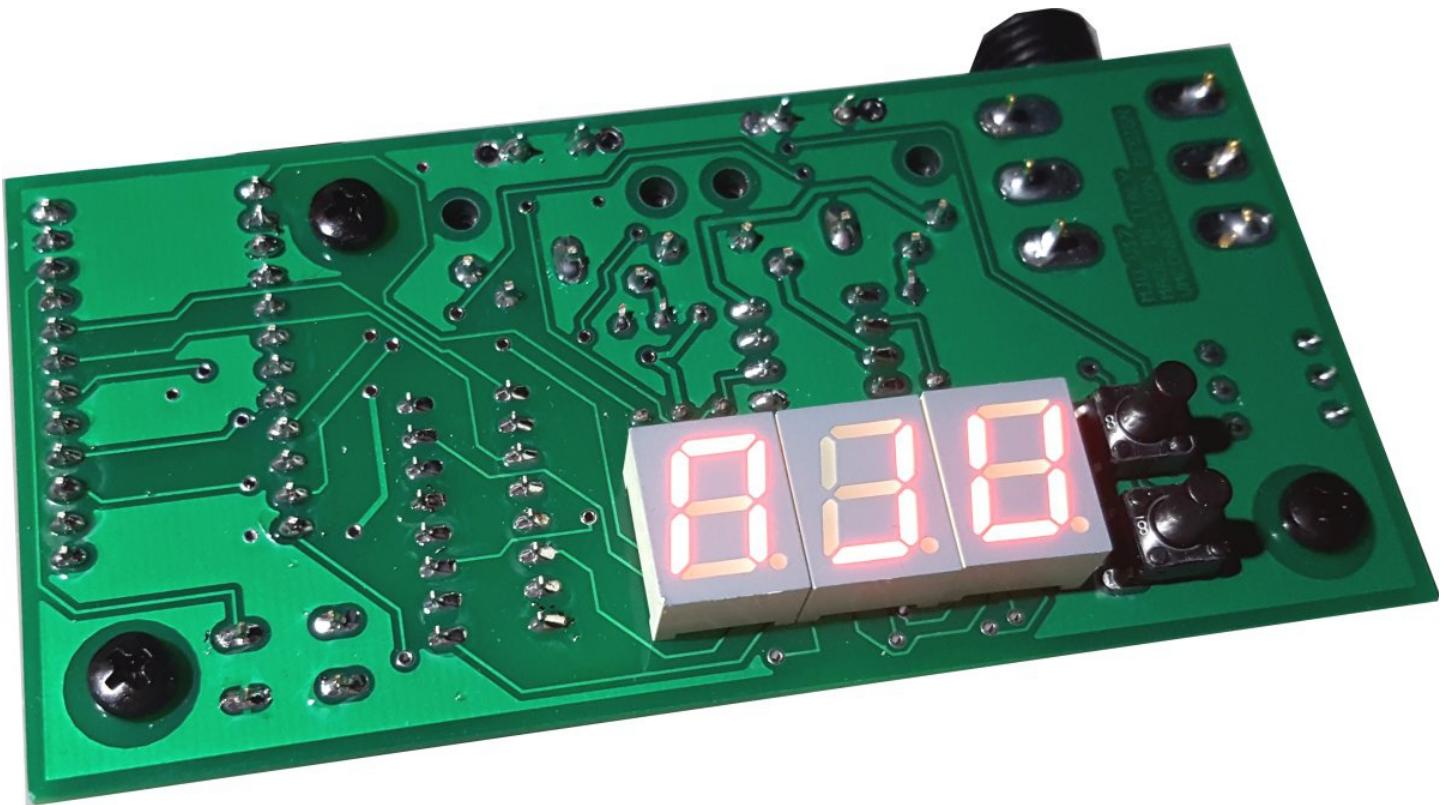


MJU – Midi Jack USB TRS Accessory & Event Processor



An open-source Do-it-yourself
project based on Arduino.

MOUNTING INSTRUCTIONS

Welcome to the mounting instruction sheet for the GMLAB MJU Drawbar Controller kit. To assemble this kit correctly and set it up to work properly, a certain skill with electronics and computers is required, plus some tools and a little bit of patience and attention.

REQUIRED TOOLS

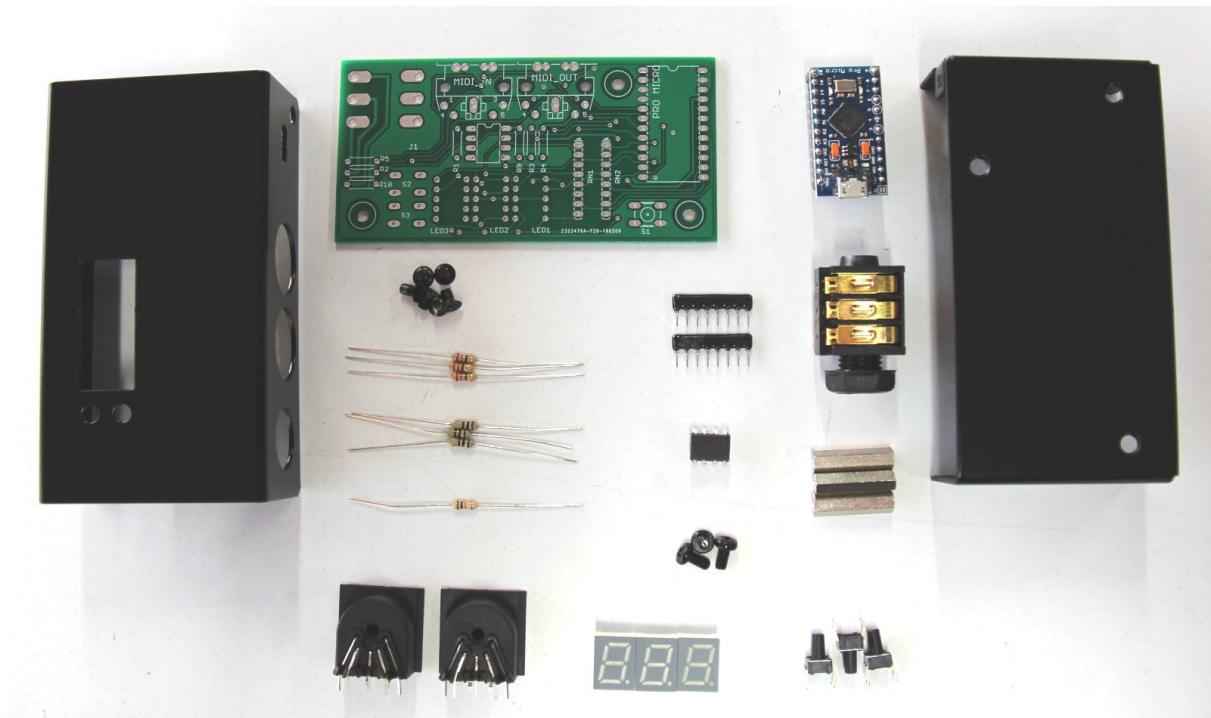
1. Soldering iron, preferably a temperature-controlled 60W iron with a 1,5 ~ 2,5 mm wide tip;
2. Solder, preferably good quality 0,8 ~ 1 mm diameter;
3. Good quality cutters;
4. Phillips screwdriver;
5. A computer with Arduino IDE installed.

PREPARATION

Prepare a clean and tidy surface, with just the required tools handy and make sure you have discharged your body from electrostatic charge by touching some metal object that makes contact with the floor. Optionally, wear an ESD wristband.

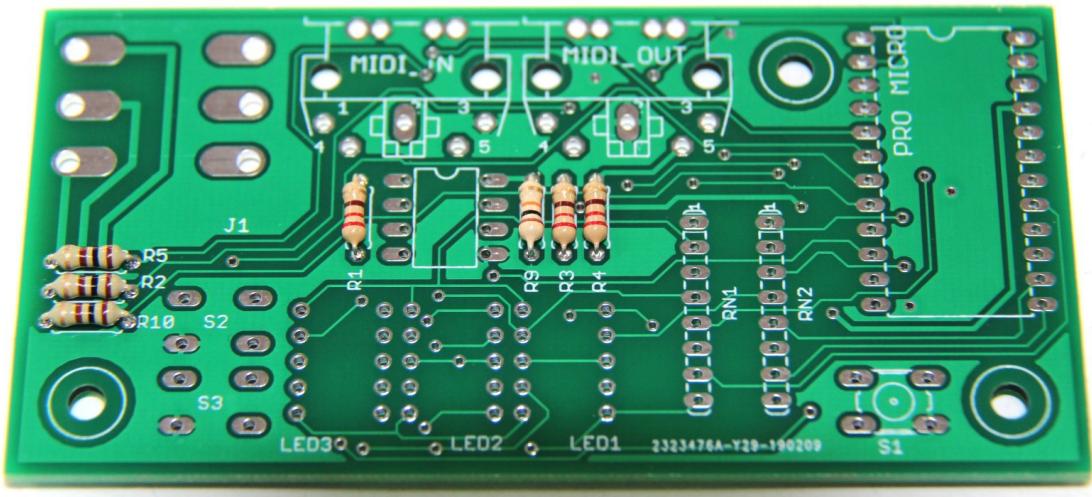
WHAT'S IN THE KIT

- 1x PCB MJU main board
- 1x Arduino Pro Micro (Leonardo clone)
- 1x Metal bottom piece
- 1x Metal top cover
- 3x 220 ohm 1/4W resistors
- 3x 100 ohm 1/4W resistors
- 1x 10K ohm 1/4W resistor
- 2x 220x4 Ohm resistor networks
- 1x 6N137 DIP-8 Photo-coupler
- 3x 7-segment LED Digit
- 3x Tactile switches
- 1x TRS 3.5mm Jack connector
- 2x MAB DIN5 connectors
- 1x USB - MicroUSB cable
 - All required screws and spacers
-

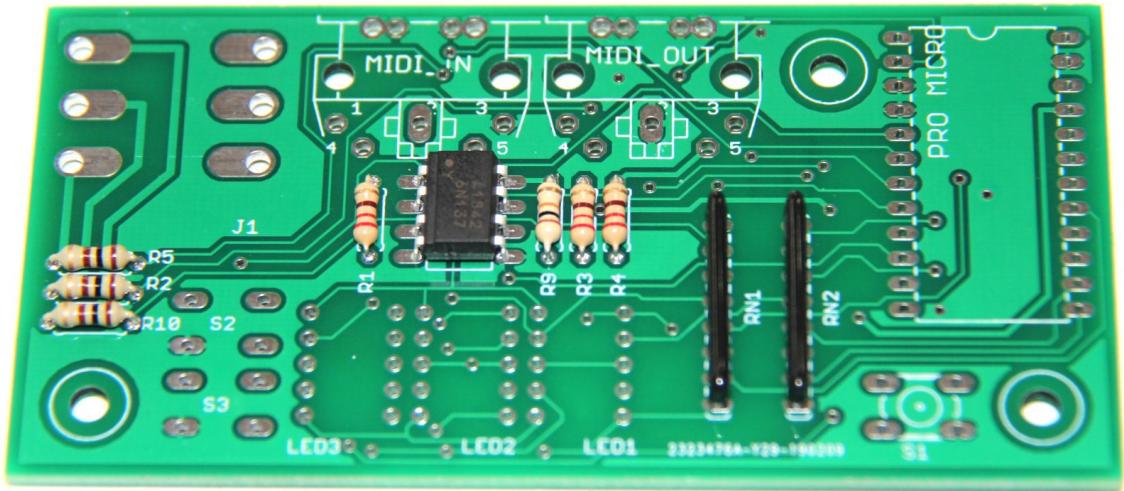


STEP 1: Let's start from the passive components, from the lowest till the tallest. Take the resistors, bend the terminals 90 degrees and add them to the Main PCB in the correct positions (check the schematics and the component numbering). Do the same with the resistor networks.

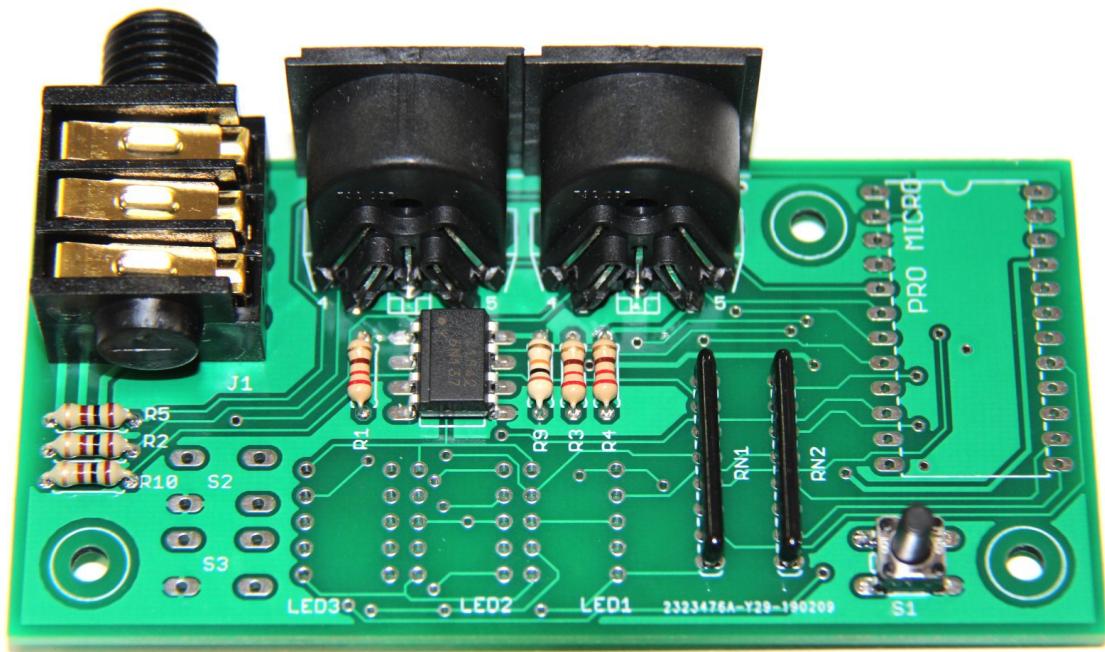
Note: the resistor networks are all parallel, don't worry about the direction, even though they have a mark on the first pin.



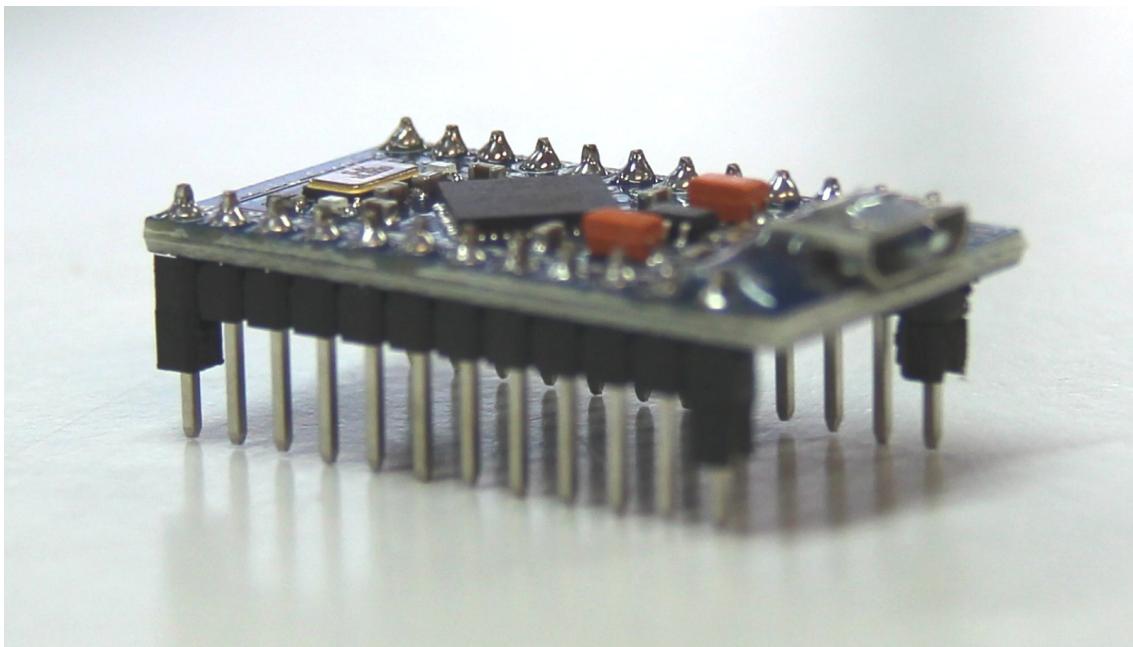
STEP 2: Add the Integrated Circuit 6N137. Pay attention not to keep the solder pin for too long, and that all pins have been soldered correctly. Do not solder two or more pins together, each pin is soldered separately. Also pay attention to the direction. Drawings on the PCB have a notch on a side indicating where pin 1 goes. The same notch is generally present on the IC itself, or you'll also find a small dot or a vertical line on the left side, where pin 1 is.



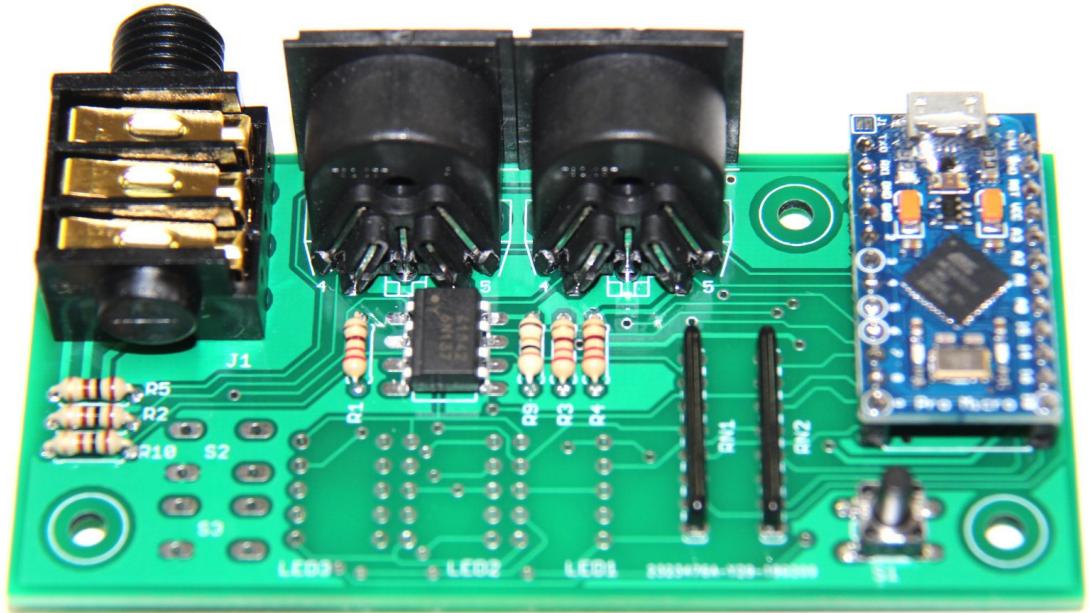
STEP 3: Mount the connectors and one of the three tactile switch. The other two tactile switches will be mounted on the other side of the board. Refer to the image below.



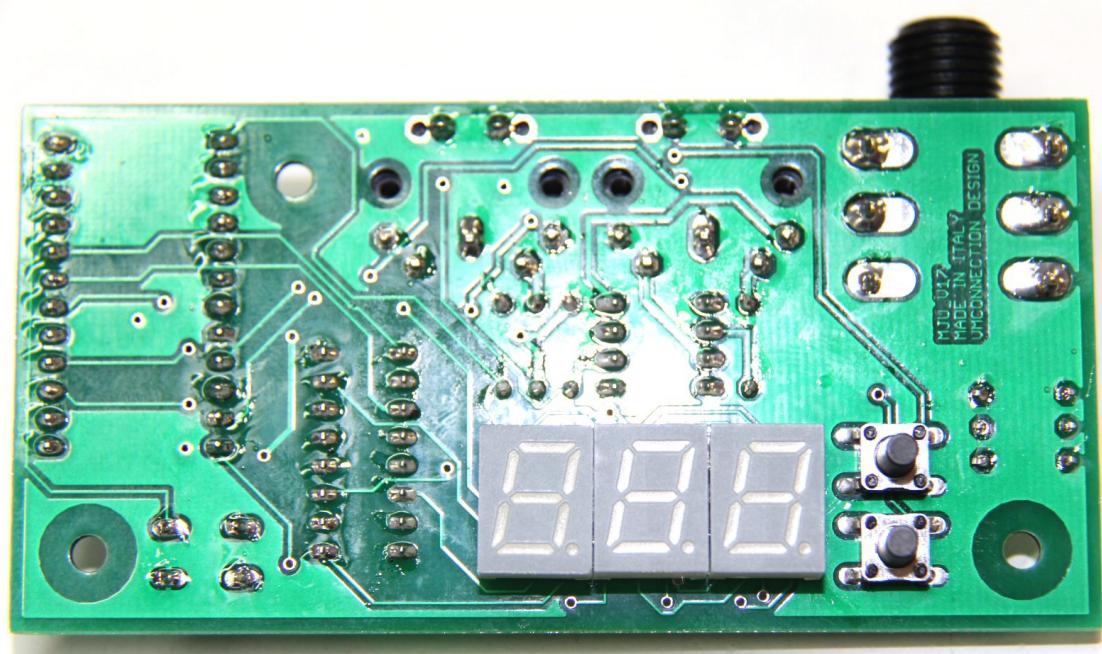
STEP 4: Add the the CPU board. Please note that the CPU comes pre-programmed with the default Arduino sketch and comes with all SIL terminals soldered. Also, you can notice from the picture below that there are four small plastic spacers mounted at the four edge pins, these are needed to keep the board at the required height so that the micro-USB socket can reach the hole in the enclosure.



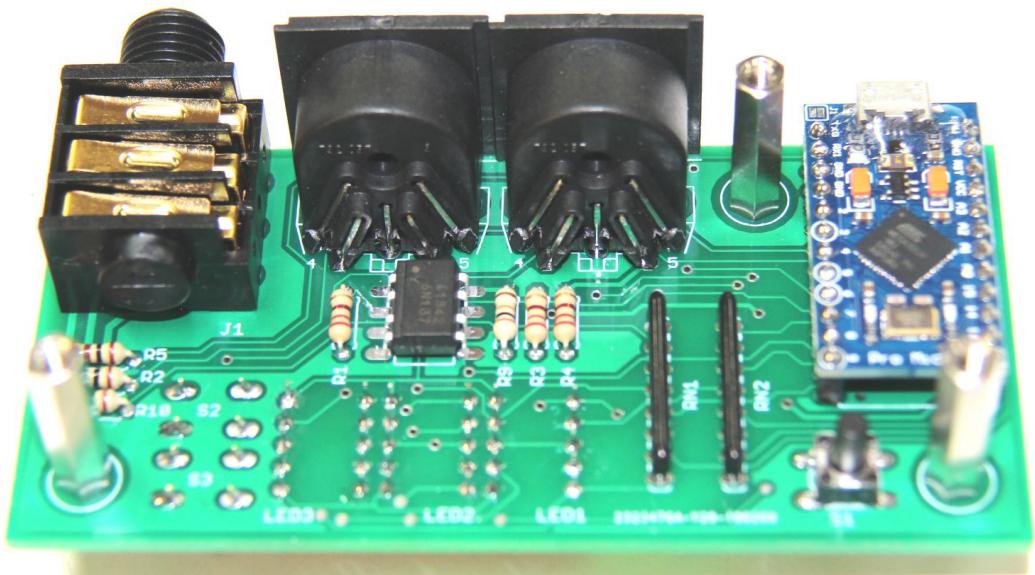
This board contains active SMT electronics and is subject to ESD and overheating, make sure that your soldering iron doesn't exceed 350 °C (~660°F) and don't keep the soldering pin on the solder pads for too long.



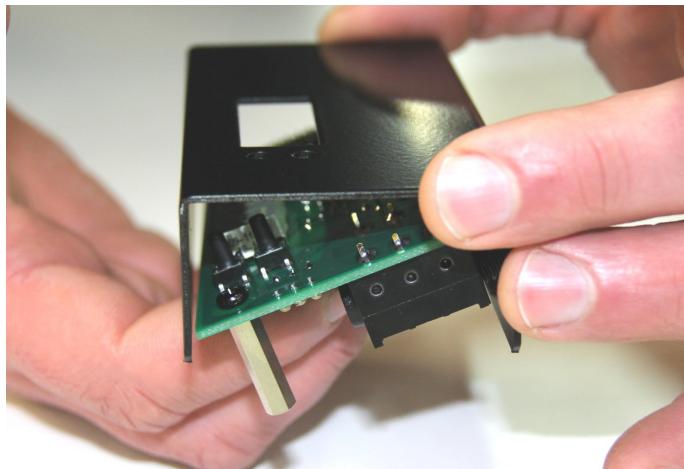
STEP 5: Turn the board upside down and mount the remaining two tactile switches and the three 7-segment LED displays. Pay attention not to mount the displays upside down. The dot in each digit must be facing down.



STEP 6: Mount the three metal spacers that will hold the board in the enclosure. Refer to the picture below.



STEP 7: Pair the board with the top piece by sliding the TRS jack shaft through the hole, then fix the plastic bolt. Keep the bolt loose, this will be tightened at the end of the process.



STEP 8: mount the bottom piece trying to reach the spacers with the screws, then tight all screws and the plastic bolt.



PROGRAMMING THE CPU WITH ARDUINO

PLASE NOTE: The Arduino board supplied with the MJU comes pre-programmed with the default MJU sketch. If you don't need to modify the source code yourself, you can skip this whole section.

To program an Arduino board, you need a computer with Arduino IDE installed. Download it from the following URL:

<https://www.arduino.cc/en/Main/Software>

Install it on your computer following all the instructions given by the Arduino documentation.

If you haven't downloaded it yet, go to www.gmlab.it and download the Arduino sketch for the MJU from the Document section.

PLEASE NOTE: before compiling the sketch, make sure you have installed the required libraries. The default sketch for the MJU uses the following libraries (that should be installed separately using the IDE library functions, in case they aren't pre-installed):

- MIDI Library by *Fourty Seven Effects* -
https://github.com/FortySevenEffects/arduino_midi_library Used to generate MIDI messages to be sent to the UART PORT
- MIDIUSB by *Gary Grewal* - <https://www.arduino.cc/en/Reference/MIDIUSB> Used to generate MIDI messages to be sent via USB
- EEPROM (built-in) Used to store and recall the current status into the internal EEPROM
- MillisTimer by *Brett Hagman* - <https://github.com/bhagman/millisTimer> Used to obtain two timers with millisecond precision

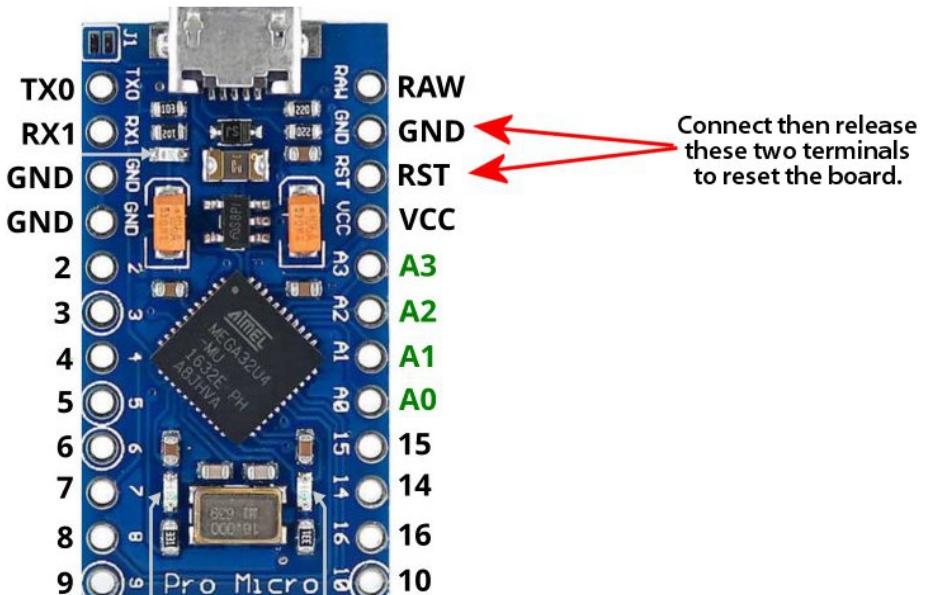
Connect the MJU to your comptuer using the provided USB cable, start the IDE, load the sketch, then:

1. From the TOOLS menu, select BOARD -> Arduino Leonardo;
2. From the TOOLS menu, make sure the selected PORT points to Leonardo;
3. click the icon with the arrow pointing right, this will compile the sketch and upload it to the board.

PLEASE NOTE: once the sketch is properly uploaded to the board, the USB port will not be seen by the IDE because it changes its function to USB-MIDI. To check that it is actually seen as a MIDI device, if you're using Windows (preferred), download and install the free application MIDI-OX and check that the ARDUINO MIDI device is listed among your MIDI ports, select it as an input port to MIDI-OX and check that it is correctly sending the expected CC messages.

An alternative to installing MIDI-OX is this quick and easy web-based MIDI Monitor utility (requires Chrome): <https://www.gsidsp.com/midimonitor/>

In case you need to reprogram the board, one second before clicking the "LOAD" icon in the Arduino IDE, you have to "reset" the board by making a contact between the terminals labeled RST and GND using a small screwdriver.



PLEASE NOTE: The MJU has an internal button for the reset, you can push the button before starting to send the sketch to the board.

GMLAB MJU is sold on-line on the web sites www.gmlab.it and on www.MyRigShop.com by V.M.Connection, an enterprise based in Veneto, Italy.

Get the source code for this and other project on our GitHub account:
<https://github.com/zioGuido>

If you need spare parts or blank PCBs, just let us know.

All trademarks mentioned in this document belong to their respective owners and are used for reference purposes only.