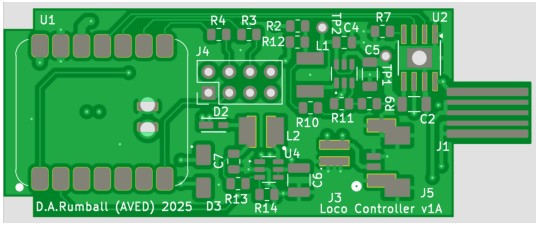
Loco Controller Construction guide

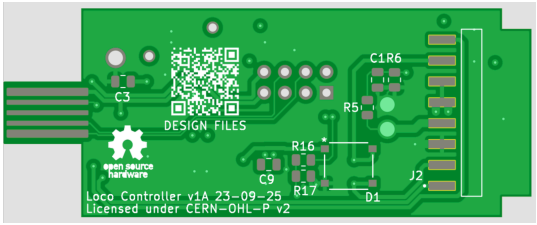
Introduction

This document contains some brief notes to assist with the assembly of the Loco Controller PCB. It assumes that the reader has some experience with fine pitch SMD soldering, if not don't panic as there are any number of online resources and YouTube videos that go over the basics.

As with all SMD assembly it's important to follow good safety procedures particularly to ensure that you have good protection from the fumes generated from soldering and from any hot solder splashes, again there is a wealth of online information concerning protection from these and other hazards, follow it!

The tips and suggestions below are intended to be guidelines and don't need to be followed slavishly, if you have a better way of doing things then by all means use it.

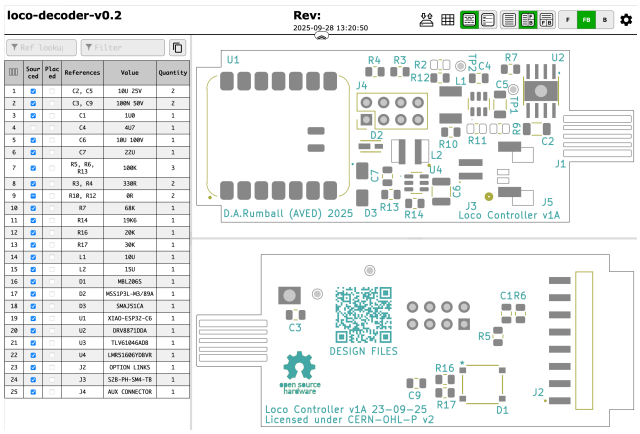




The PCB has been designed with an eye towards manual assembly and has a number of features to simplify the process including the use of relatively large surface mount components with generous spacing. It's entirely possible to assemble the board using just a temperature controlled soldering iron and without needing to use a hot air gun or reflow plate.

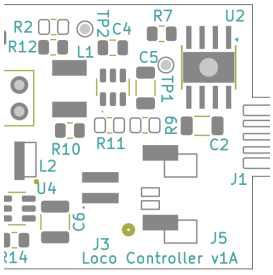
BOM

The component BOM is given in the hardware folder of the release package and as an aid to ordering the BOM is available as a list on DigiKey. For convenience during assembly there is also an 'interactive placement guide' with has the ability to visually correlate and easily search for components and their placements on the PCB.



PCB Options

- R9 and R11 are no fit if using a 12V motor, see the User Notes for details on how to specify these for different motor voltages.



- R12 and R2 select whether the motor boost regulator is powered from the 3V3 rail or directly from the battery. The default is for R12 to be fitted for battery power.

- There are pads on the PCB to allow the fitting of either a 2mm or 1.25mm ‘JST style’ battery connector. The default is for a 2mm connector.

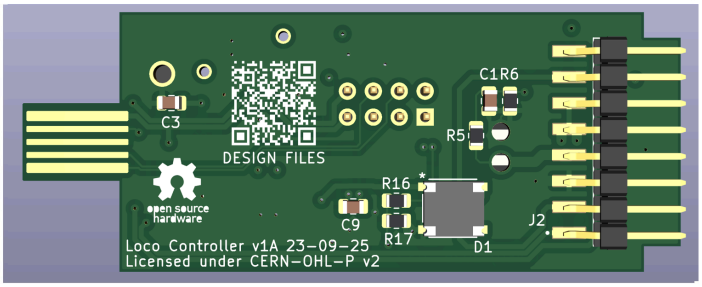
Construction Notes

- The surface finish of the PCB is HASL (Hot Air Solder Levelling) as this eases manual soldering of SMD components. With this finish it’s important to clean the PCB thoroughly before assembly (IPA works well) and to use a good quality flux. I’ve found that one of the no clean ‘gel’ fluxes gives the best results.

- First place all the components on both sides and option link header on the underside of the PCB (leaving off D2 and U1-4 for now). These components do not have a preferred polarity with the exception of D1 (D3 is bi-directional). D1 has marking on the top for the AC inputs marked with a ‘~’ and DC output, it should be installed with the AC inputs closer to the option header and the +VE output next to the ‘star’ mark on the PCB.

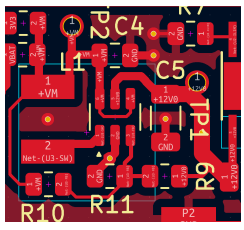
- After placing these components start testing by confirming that power applied to the 2 PWR_IN pins on the option header gives the same voltage across C6 minus approx 1.2V (two diode drops from the bridge rectifier) regardless of polarity of the input.

MOTOR-
GND
N/C
LED COMMON (3V3)
LED_Y

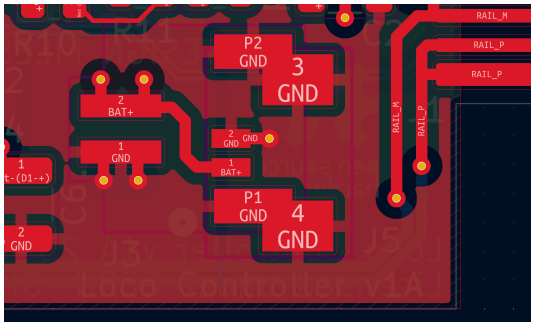


RAIL+
PWR_IN_A
BATT+
BAT_LINK
BAT_LINK
BATT- (GND)
PWR_IN_B
RAIL-

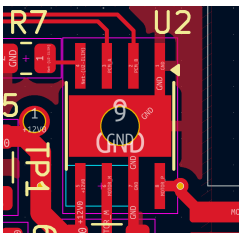
- Next place U4 (input buck regulator), noting the correct orientation of pin one shown by a small triangle on the PCB. The part is marked ‘606Y’ and has a small dot by pin 1 which can be difficult to see without the help of a magnifying glass and strong light. Test the correct function of the regulator by applying power again as before and observing approx +5V on the pad of D2 closest to the inductor. If this test is OK then place D2 noting the polarity with the line marked on the top of the part furthest away from the inductor.

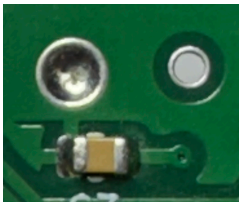


- Now place U3 (boost regulator) again noting the correct polarity. In this case the part is marked ‘1C4F’ and has a vertical line indicating pin 1 and again a small triangle marks pin 1 on the PCB. To test this part connect the battery noting the correct polarity of the battery which changes depending on which size of connector is used as shown below. With the battery connected there should be a steady 12V on TP1 located close to U2.



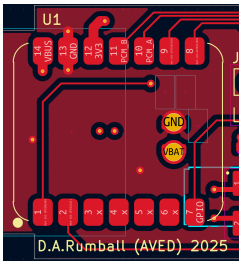
- Next place U2 (motor controller) noting the correct polarity. The part is marked ‘8871’ and has a vertical line denoting pin 1 and pin 1 on the PCB is again denoted by a small triangle. U2 has a ‘hidden’ ground pad underneath its package and so that this device can be attached without reflow there is a large via in the PCB underneath the part allowing the hidden pad to be soldered from the bottom of the PCB. Make sure to use plenty of flux with a hot iron as the ground planes on the PCB will cool the joint.

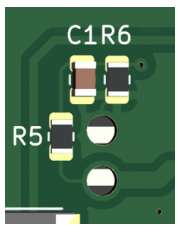




A good joint will look like this, note the smooth concave appearance caused by the molten solder adhering to the base of the part and sides of the via.

- The final part to fit is the ESP32C6 module, U1. Although not essential, testing is simpler if this part is programmed (flashed) before installation (see ‘Flashing the Firmware below). This part should be placed centrally to the PCB pads with the USB-C connector facing the front of the PCB (LHS in this diagram) and the PCB is marked with an outline to aid alignment. Hold the part in position temporarily with a small piece of tape then solder just one of the pads. Check alignment making adjustments if necessary then solder the remaining pads. As with U2 there are two pads underneath the device (adjacent to C1/R6) that need to be soldered to large vias on the bottom side of the PCB.





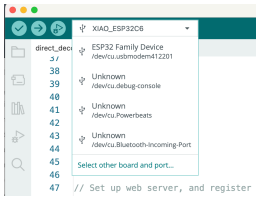
R5/

- Fit the optional expansion connector header pins if required.

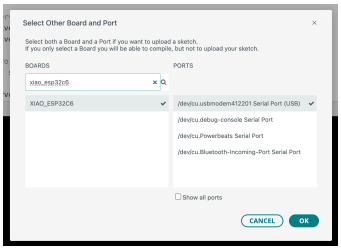
- Finally to complete construction, it’s a good idea to give the PCB a good clean with IPA and an old toothbrush to remove any residue from the flux.

Flashing the firmware

Flashing the Loco Controller's firmware requires the installation of the free Arduino IDE. Details on this can be found in the 'Firmware Build and Flashing Guide'.

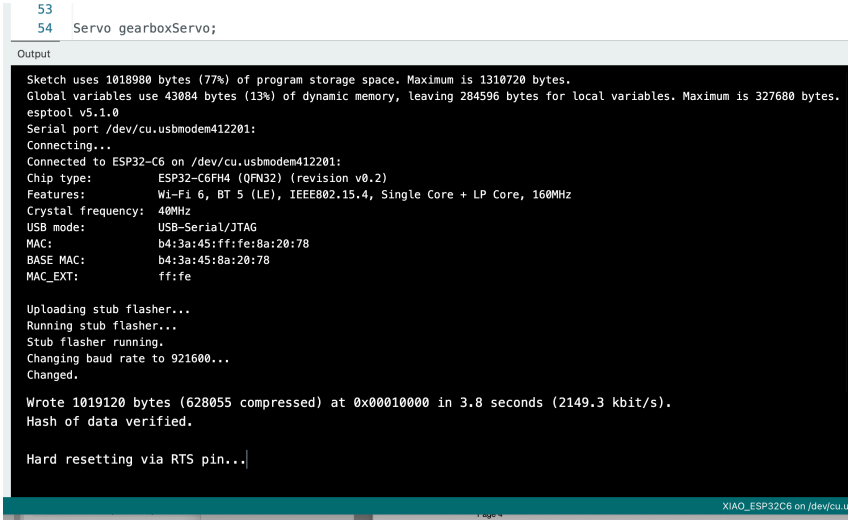


Assuming the IDE is correctly installed then choose one of the example sets of firmware and open the .ino file with the IDE and connect the Loco Controller to the computer with a USB-C cable. In the IDE select 'Select other board and port…' from the dropdown at the top of the window.



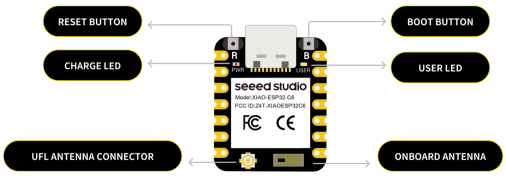
Search for 'XIAO_ESP32C6' and select this board. To the RHS of this dialog will be a list of attached serial devices, one of which will be the Loco Controller. A simple way to determine the correct one is to disconnect the unit and see which entry disappears. Reconnect the board and then select it's entry in the list.

Once the board is selected close the dialog and hit the 'upload' button (right facing arrow). If all is well with the installation the the firmware will compile and begin to download to the device ending with the message 'Hard resetting via RTS pin…'.



At this point the 'it's alive' yellow LED next to the USB-C connector will start flashing indicating that the firmware has downloaded correctly and is running.

If the ESP32C6's boot loader is inactive the IDE won't be able to connect to the device during flashing. If this happens press and hold the 'BOOT' button on the ESP32C6 module whilst applying power, then release. The 'BOOT' button is located next to the USB-C connector opposite pin 1.



Some of the firmware examples will also require the ESP32's flash based file system to be set up, details on this progress are in the 'Firmware Build and Flashing Guide'. The file system only needs to be refreshed if it's contents are changed it does not need to be touched when changes are made just to the firmware.

Final Testing

The Loco Controller can now be tested by connecting an external power source to emulate track power, battery, motor and any other external devices such as LEDs or servos. The exact details of controlling the device will depend on the particular firmware installed and details of the example firmware can be found in the 'User Notes'. Typically this will involve connecting to the controller's inbuilt WiFi access point and then opening a web page to present the particular User Interface for that example. Apart from motion controls the UI will typically present an indication of battery voltage and charging status which will confirm operation of charging from external power.

If you have got to this point with a functional unit then

Congratulations on a successful build!