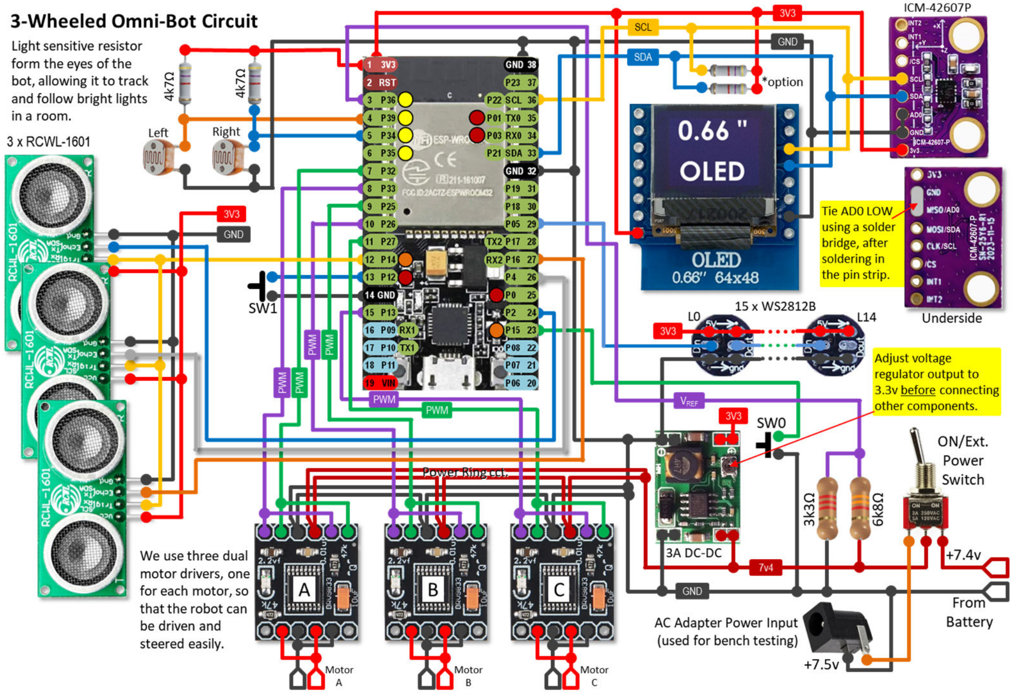
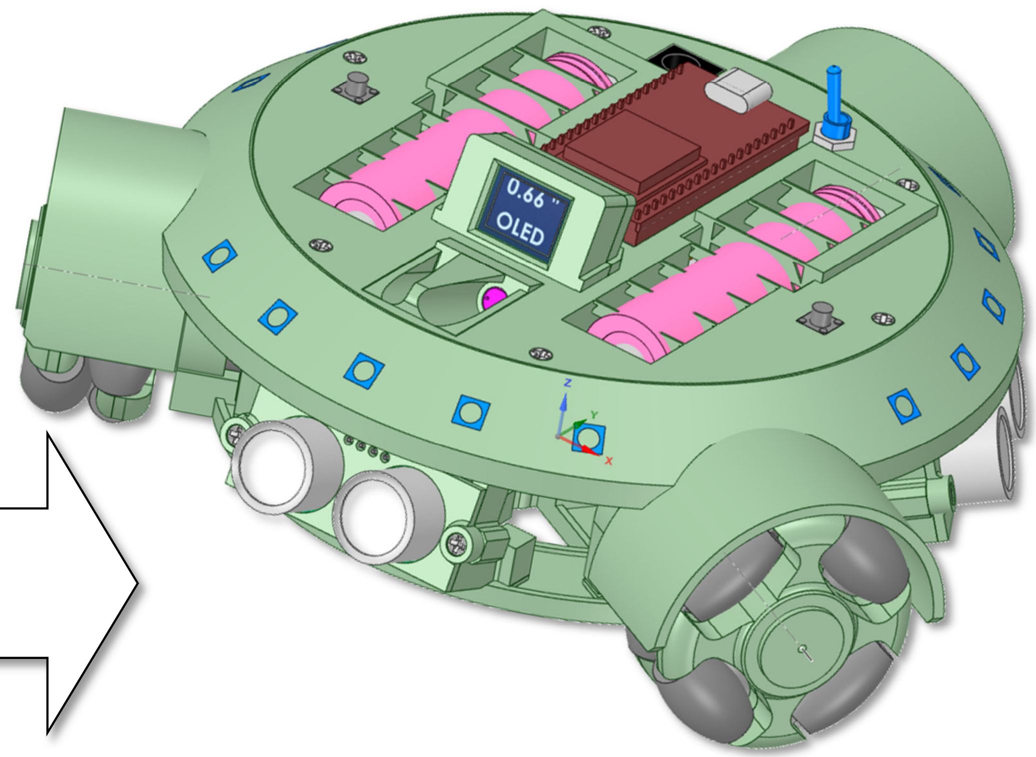
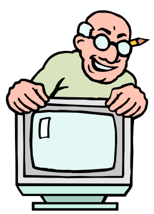
Project: 3-Wheeled Omni-Bot

Circuits & Wiring





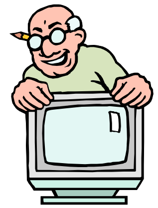


Read through this documentation completely before attempting this project.

+ ESP32

Issue: 1.1 Released: 28/11/2025 TechKnowTone

CAUTION



Lithium batteries can be extremely dangerous, if not handled and cared for properly. This design does not include any form of current limiting circuit, like a fuse. So, care must be taken to ensure that the wiring guidelines are followed accurately, that checks are made for short-circuits, and that battery polarities are marked, and they are inserted the correct way round. Failure to do so, could result in an explosive fire.

Charging Practices: Always remove batteries from your project to charge them. Use a charger, designed for the battery used, and from a trusted supplier. Choose a flat, non-flammable surface to charge on, away from flammable materials. Never leave unattended when charging. Don’t charge overnight. Monitor charging to ensure charge characteristics are as expected. Only pair batteries with similar characteristics. Do not overcharge, or leave charging for prolonged periods. This increases the risk of damage and fire.

Battery care & maintenance: Stop using a battery if it is swollen, damaged, dented or leaking. Never charge a damaged battery. Never allow a Lithium battery to discharge below 3.2 volts, as cell damage will occur. Avoid extreme temperatures. Do not charge or store batteries in very hot or cold environments. Don’t cover batteries whilst charging, as this can trap heat, causing overheating.

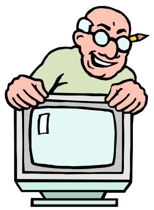
In case of fire: Get out and stay out. If a fire starts, leave immediately, and call the fire brigade. For low voltage Lithium batteries, water is a safe extinguisher.

Built-in Monitoring: Most of my project designs include code, and circuitry, to monitor battery voltage, whilst in use. This code then seeks to alert the operator, when the battery has reached a critical low voltage, before shutting down power consuming circuitry; including the micro. Time should therefore be spent on calibrating this feature, as a precaution, for good battery management and maintenance.

Carefully dispose of batteries that damaged, or discharged below their critical voltage.

Hand Tools:

Recommended:
- Fine nosed pliers
- Side cutters
- 1.5 mm Drill
- 2.0 mm Drill
- 4.0 mm Drill
- Needle files
- Screwdrivers
- Craft knife



Note: Not all items needed are shown here.

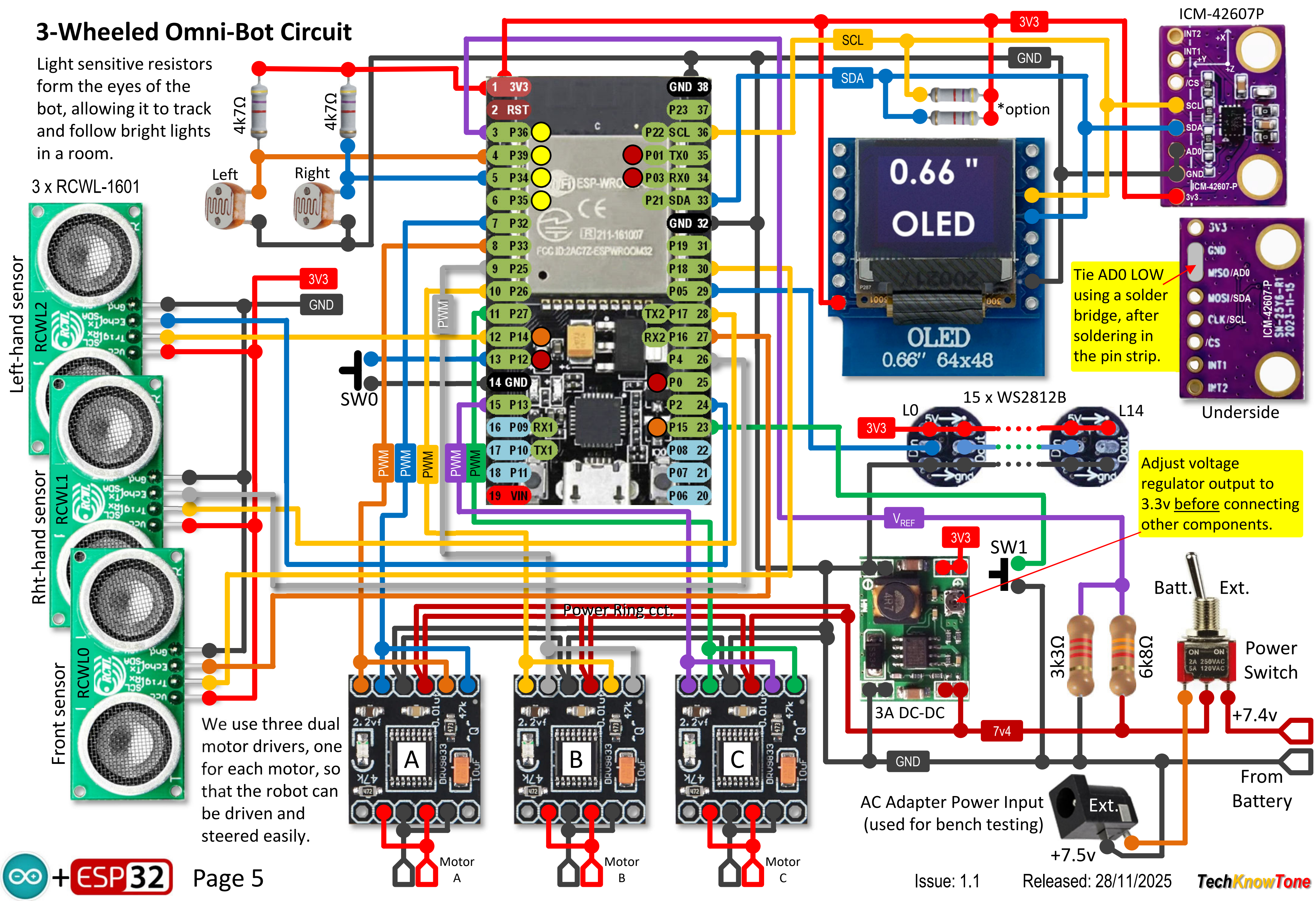
Some printed components act as aids and gauges. Use them.

Tools & Materials:

Temperature controlled iron
Solder flux
Resin cored solder
Hot melt glue gun {optional)
2-part epoxy resin glue
Screw drivers
Tweezers
Wire wrapping tool
Wire wrapping wire 30 AWG
24 AWG stranded wire (red, black & yellow)
Multimeter

3-Wheeled Omni-Bot Circuit

Light sensitive resistors form the eyes of the bot, allowing it to track and follow bright lights in a room.



We use three dual motor drivers, one for each motor, so that the robot can be driven and steered easily.

Chassis Base Wiring

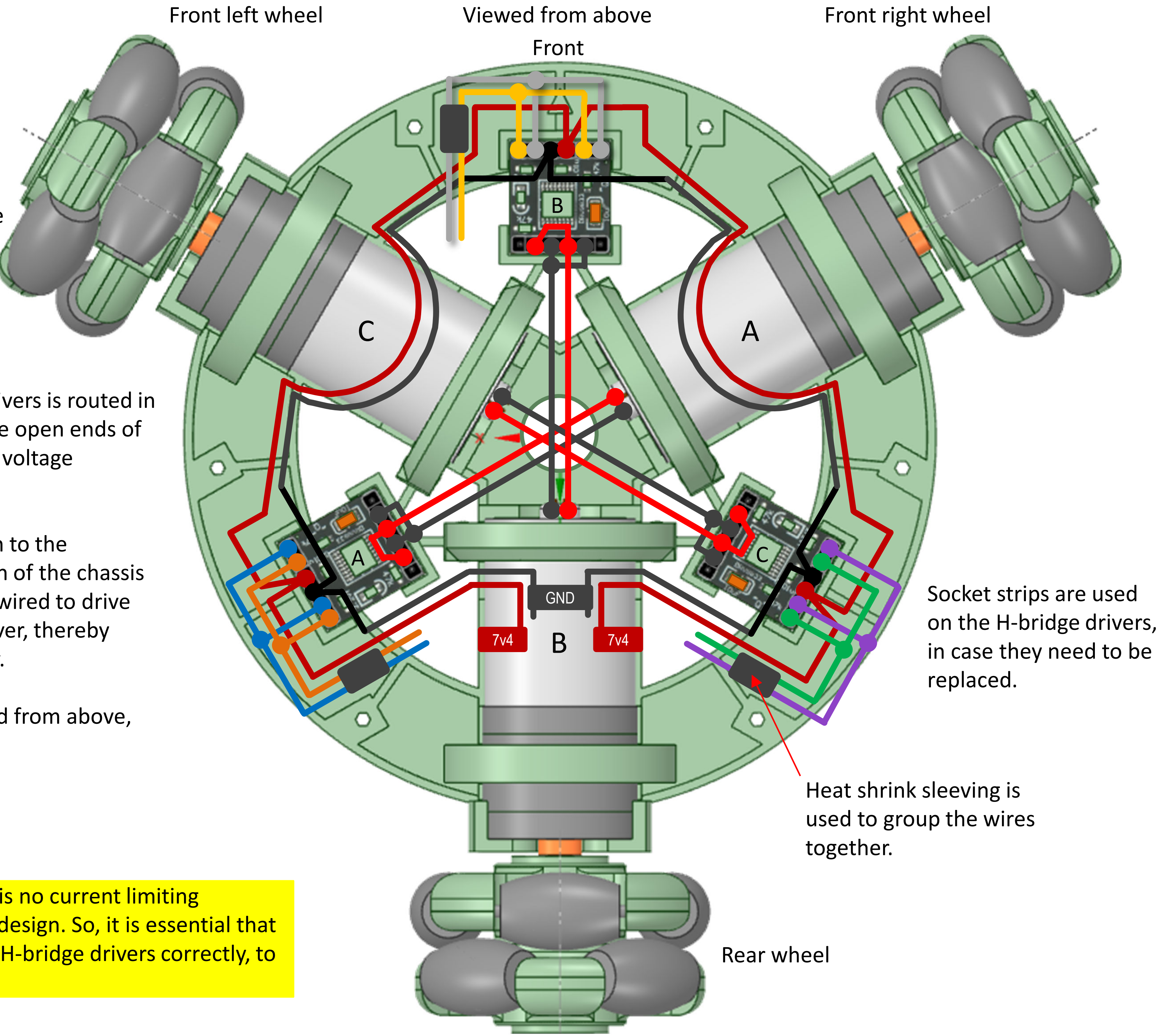
This diagram shows the wiring of the chassis motors and their associated H-bridge drivers.

Note that the nominated H-bridge drivers are diagonally opposite the assigned motors. And that the 2-channels are grouped together.

The 7v4 power to the H-bridge drivers is routed in the form of a ring circuits, with the open ends of each ring being terminated at the voltage regulator on the micro plate.

The PWM drive signals also return to the microplate, from the centre region of the chassis plate. The PWM drive signals are wired to drive both channels of the H-bridge driver, thereby combining their current capability.

The diagram is shown, as if viewed from above, through the microplate.



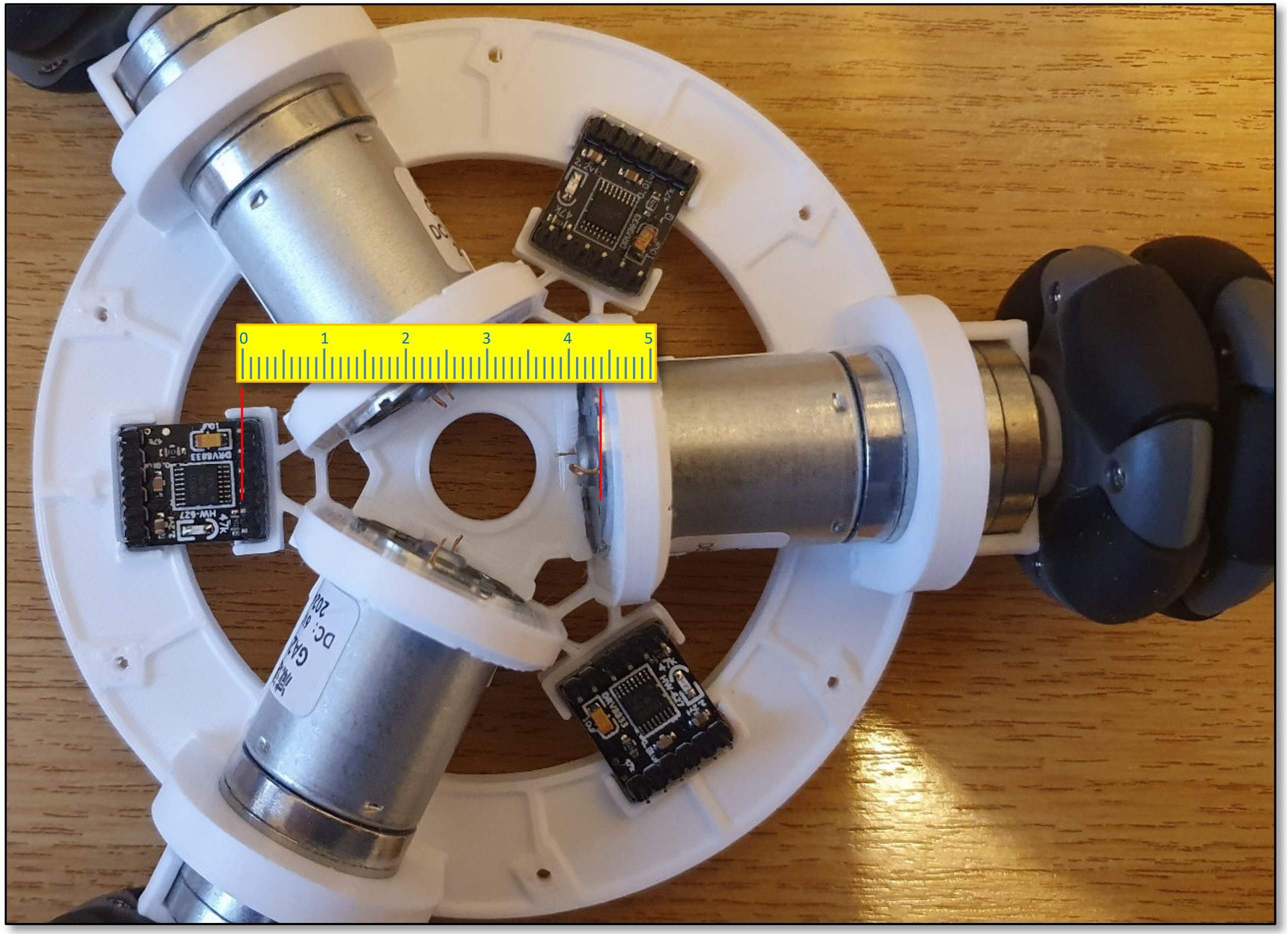
WARNING: There is no current limiting protection in this design. So, it is essential that you wire the dual H-bridge drivers correctly, to avoid damage.

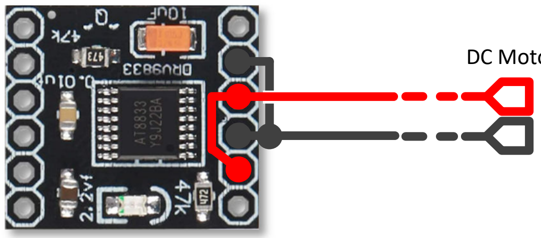
DC Motor Wiring

Each motor is wired to its opposite H-bridge driver board. To keep the wire lengths to a minimum, and nice and tidy, it is recommended that you prepare red and black wire leads in the following manner.

The overall wire lengths are 85mm. At the H-bridge end the insulation is removed for 20mm, and at the motor end 10mm. I used multi-stranded tinned copper wires, which I stripped, twisted and tinned with solder, to stiffen them. That makes it easier to form them on the end of the 4-pin socket strip.

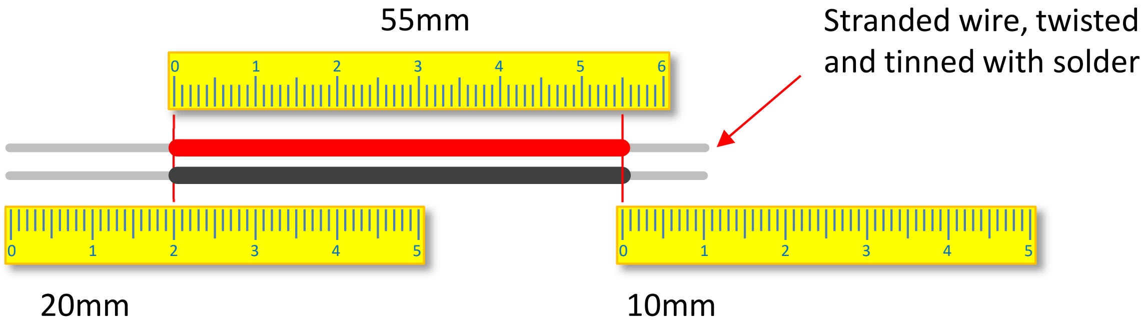
The four wire, 2-channel, outputs of the dual H-bridge are combined in a specific manner, to increase the current capability of the driver. If these are wired incorrectly, then damage can result.



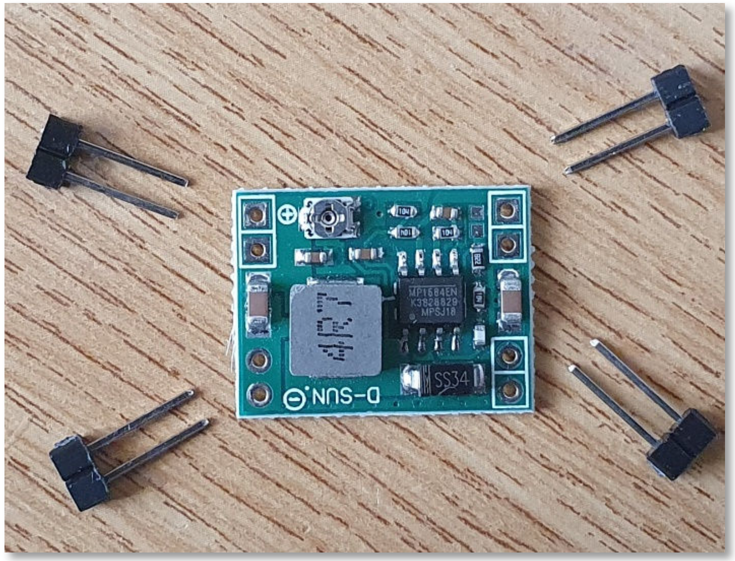


The red and black wires are formed and soldered to the pins as shown. Always check your wiring with a multimeter for short circuits.

Heat shrink sleeving

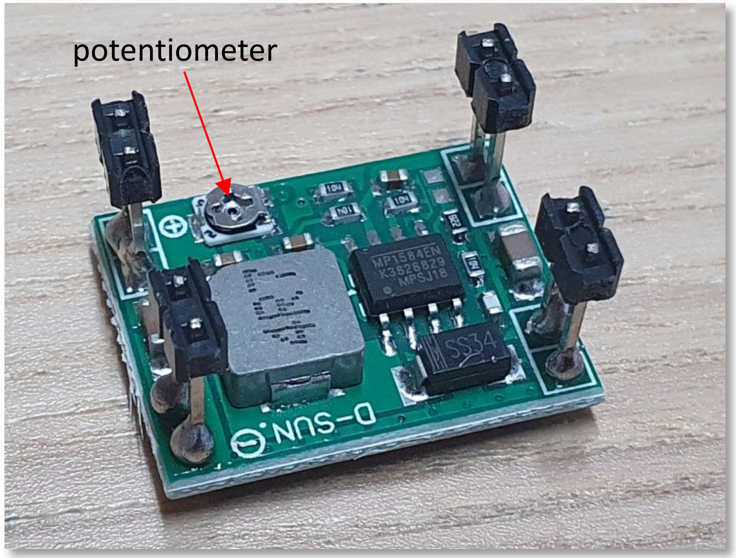


Voltage Regulator



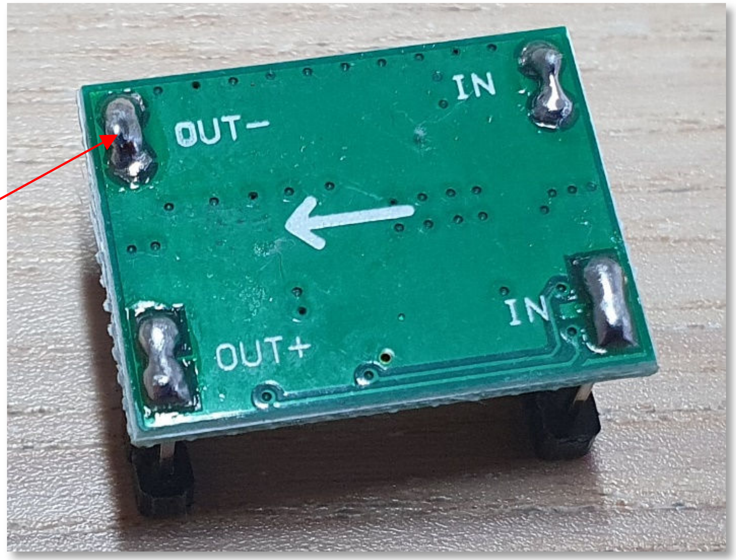
A 3 amp switch-mode voltage regulator is used in this design, given the amount of circuitry connected to the 3v3 rail. The linear regulator fitted to the ESP32 may not cope well with this, and tends to overheat. I wanted long pins on this part, so I used 4 x 2-pin strips, with the insulated bodies pressed to one end.

Apply a small amount of flux to the pins, before inserting them into the pcb. These can be difficult to solder if your iron is not of sufficient wattage, or set too low a temperature.



Once they are soldered into the board, the plastic body ends were removed, to gain better access for wiring. Note that soldering the pins can be a challenge, particularly the negative connections, as the board can absorb quite a lot of heat. You may need to turn up the heat on your soldering iron, and use a broader tip, to achieve good heat flow.

On the underside of the pcb I also applied solder, to bridge the 2-pin connections, to improve connectivity. Connect this regulator to a 7.5 – 8.5v DC supply, and adjust the potentiometer to be in the range 3.3 – 3.4v, before fitting it into the Omni-Bot. Note, prior to adjustment, the voltage may be high initially, so don't have it connected to other devices, until it has been set up.

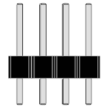


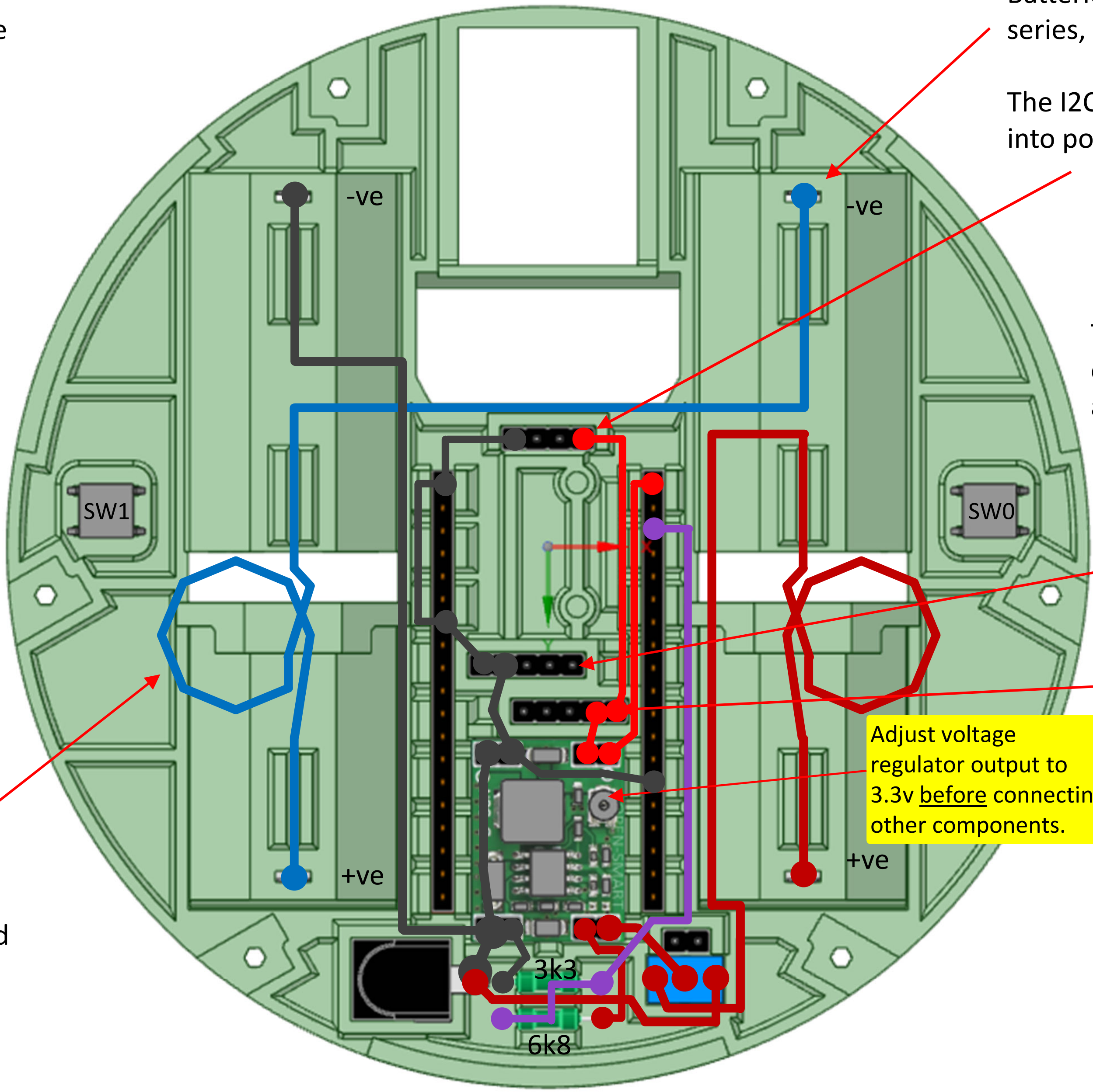
Micro Plate Power Wiring

This diagram shows the initial wiring of the micro plate, where we run in the power wires, from the batteries and external socket, to the voltage regulator. Note, that the voltage regulator should be set to 3v3 before the micro, and other electronic components, are inserted. See photos later in this document.

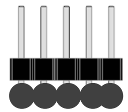
Batteries are connected in series, via the blue wire.

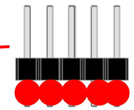
The I2C pin strip is glued into position here.





The GND, 3v3 and Trig common connection strips are pre-wired, and soldered, then glued into position as shown here.





Adjust voltage regulator output to 3.3v before connecting other components.

Wires from the positive ends of the batteries are looped, so that there is sufficient length to allow the case to slide up and out, for the batteries to be removed and inserted.

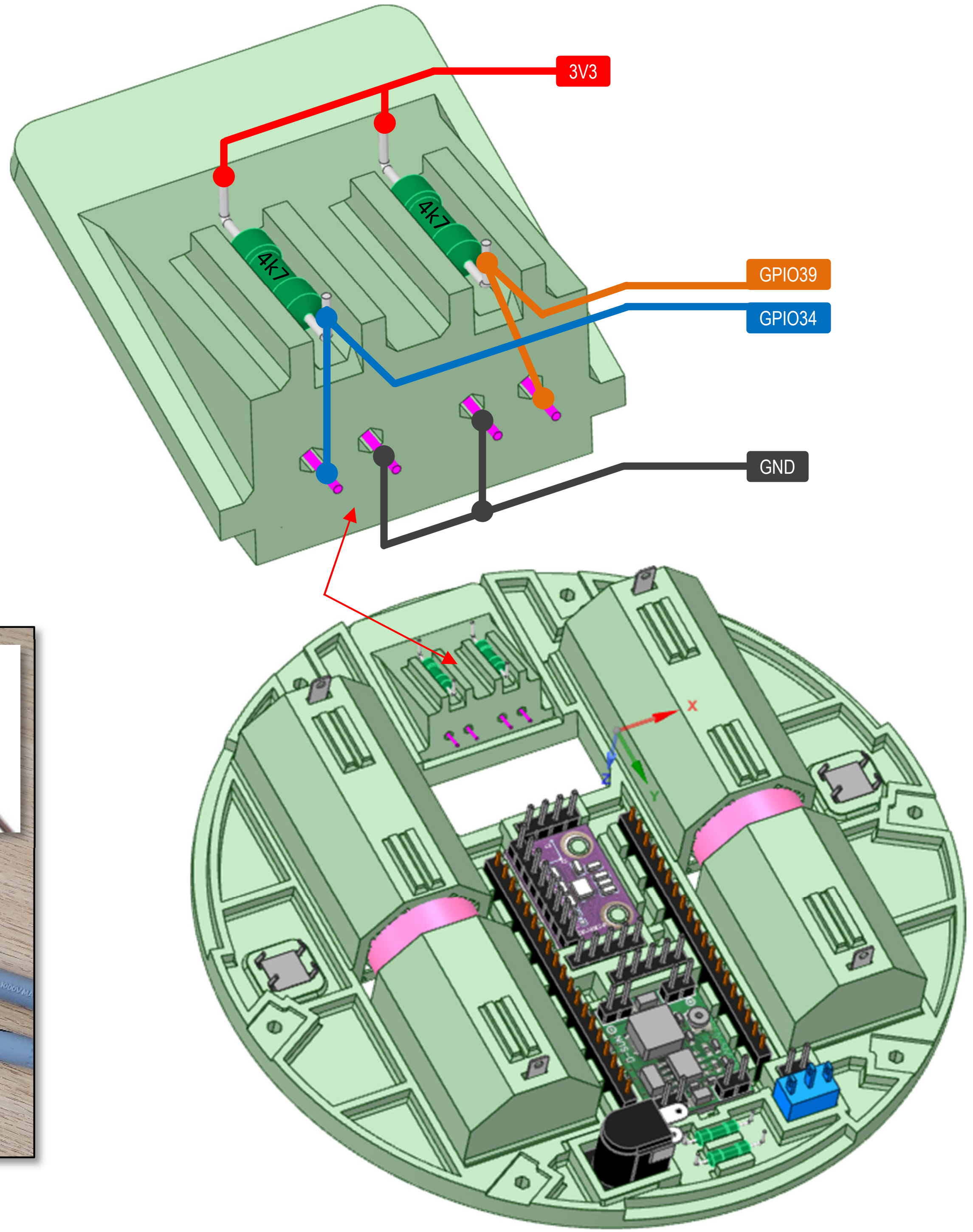
Light Sensor Wiring

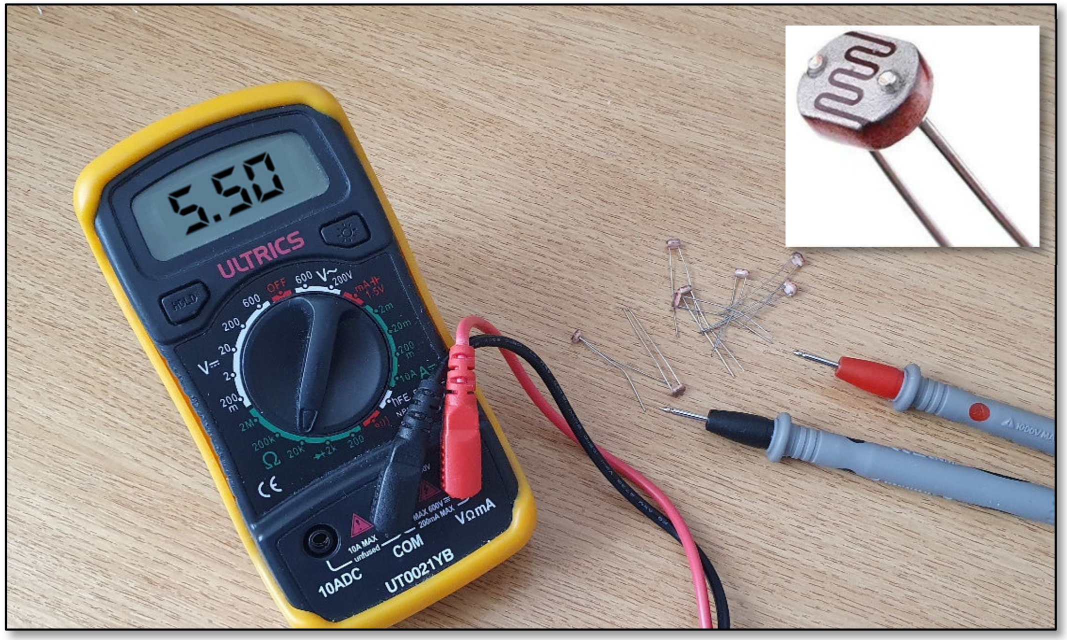
The light sensor assembly , is wired like this. The light sensitive resistors are first selected, then glued into the filter body, with their leads coming through the rear. And two 4k7 resistors, are glued into position, in wells, as shown.

The part is then pre-wired with wire wrap wire, prior to being glued into the underside of the top micro plate, as shown below.

The two light sensitive resistors are first selected from a group, as the closest matching pair. Under normal room lighting conditions, you are looking for matching values, in the range between 5 - 6kΩ.

Wires are wrapped and soldered onto the sensor, before gluing it into the micro plate.



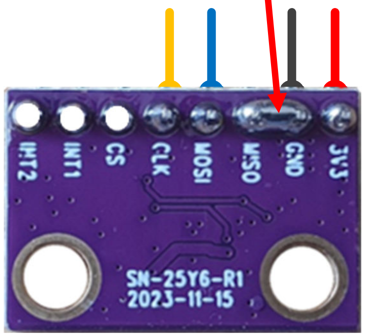


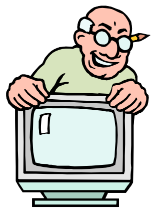
Micro Plate Wiring Continued

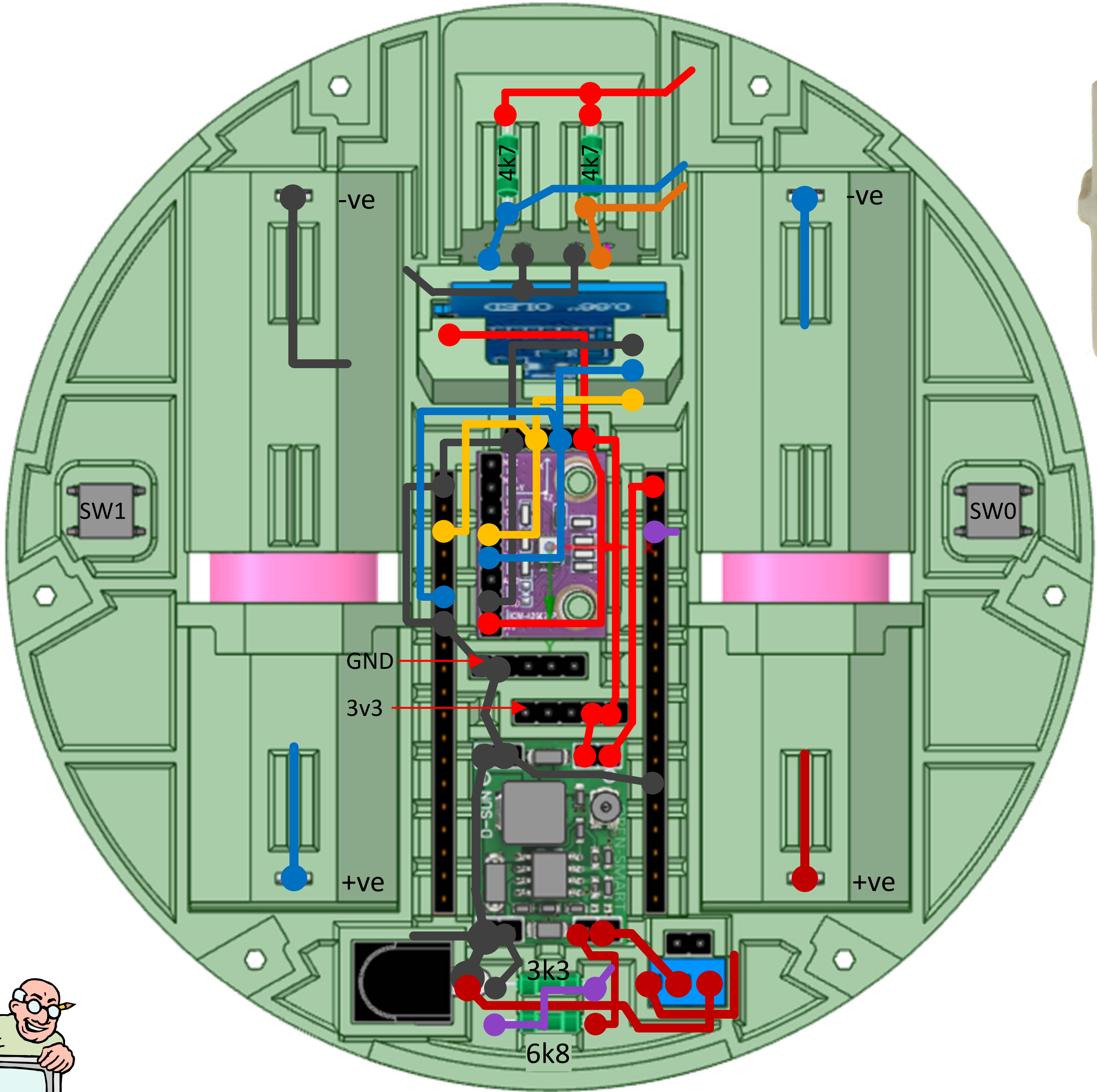
This diagram shows the next phase of wiring. Most of the previous connections have been omitted, to prevent the diagram from becoming too cluttered. I just show the connection points here.

Here the I2C connections are wired to the 4-pin strip common connection points, and then onto the display and ICM motion sensor.

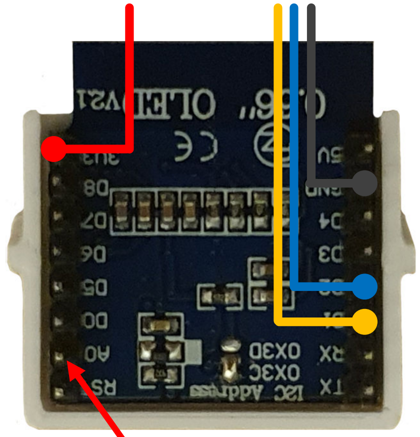
Solder the ICM AD0 pin to GND. It is recommended that you test the ICM sensor before gluing it into the case. I have supplied code for this.







The display assembly is pre-wired before gluing it into the micro plate.



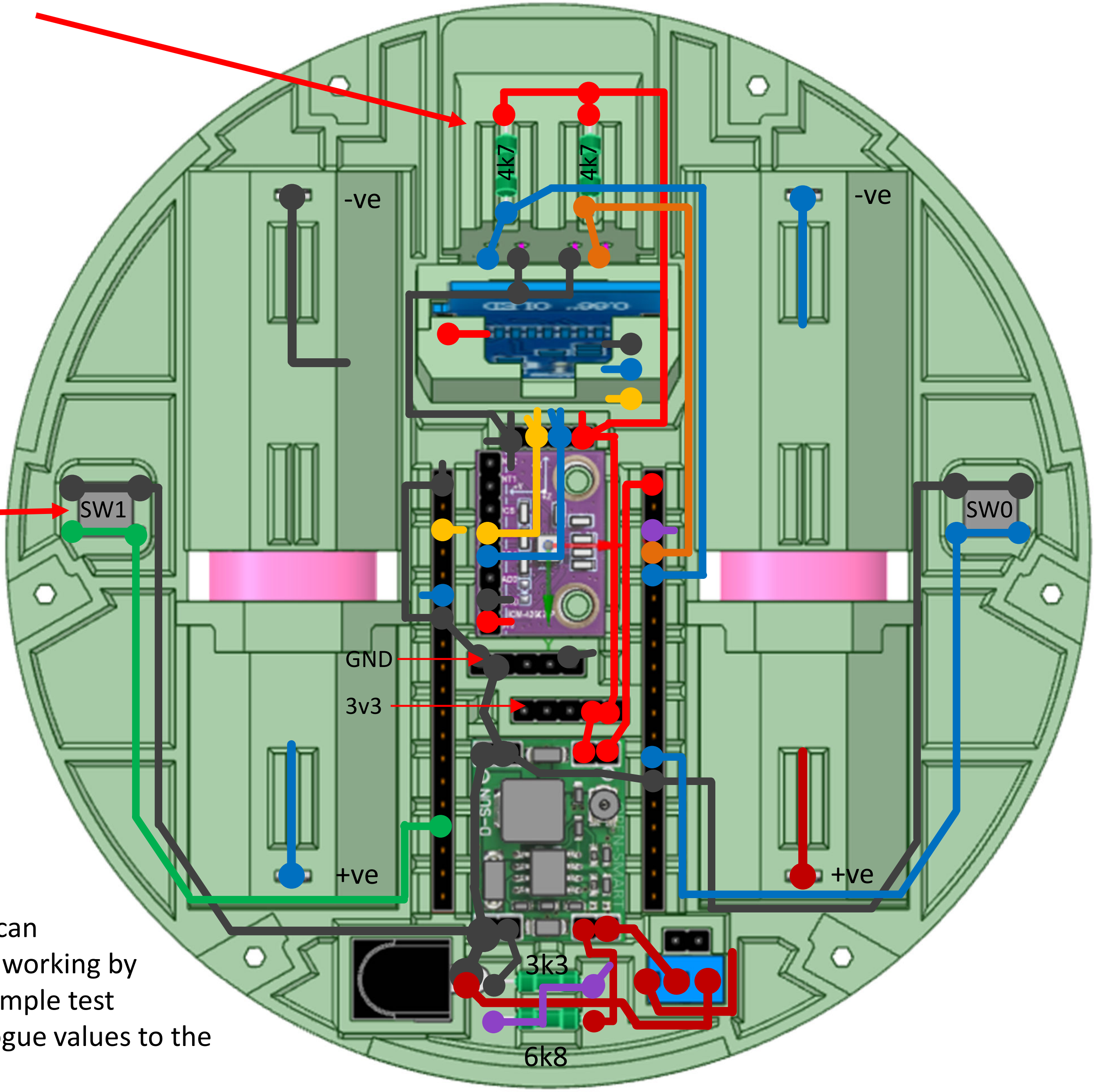
It is recommended that you wire-wrap the pins. Then test the display. Then crop back all pins to about 3mm, so that the back cover will fit. The pins will be too long otherwise. Solder the wire-wrapped pins before closing the assembly

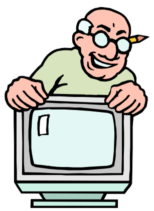
Note that my code package, contains small and simple test programs for all sensors. Which can be used throughout the build process, to confirm functionality, as the build progresses.

Micro Plate Wiring Continued

This diagram shows the next phase of wiring. Here we wire up the light sensitive resistors to the micros ADC inputs.

Then we wire in the two button switchers SW0 and SW1.



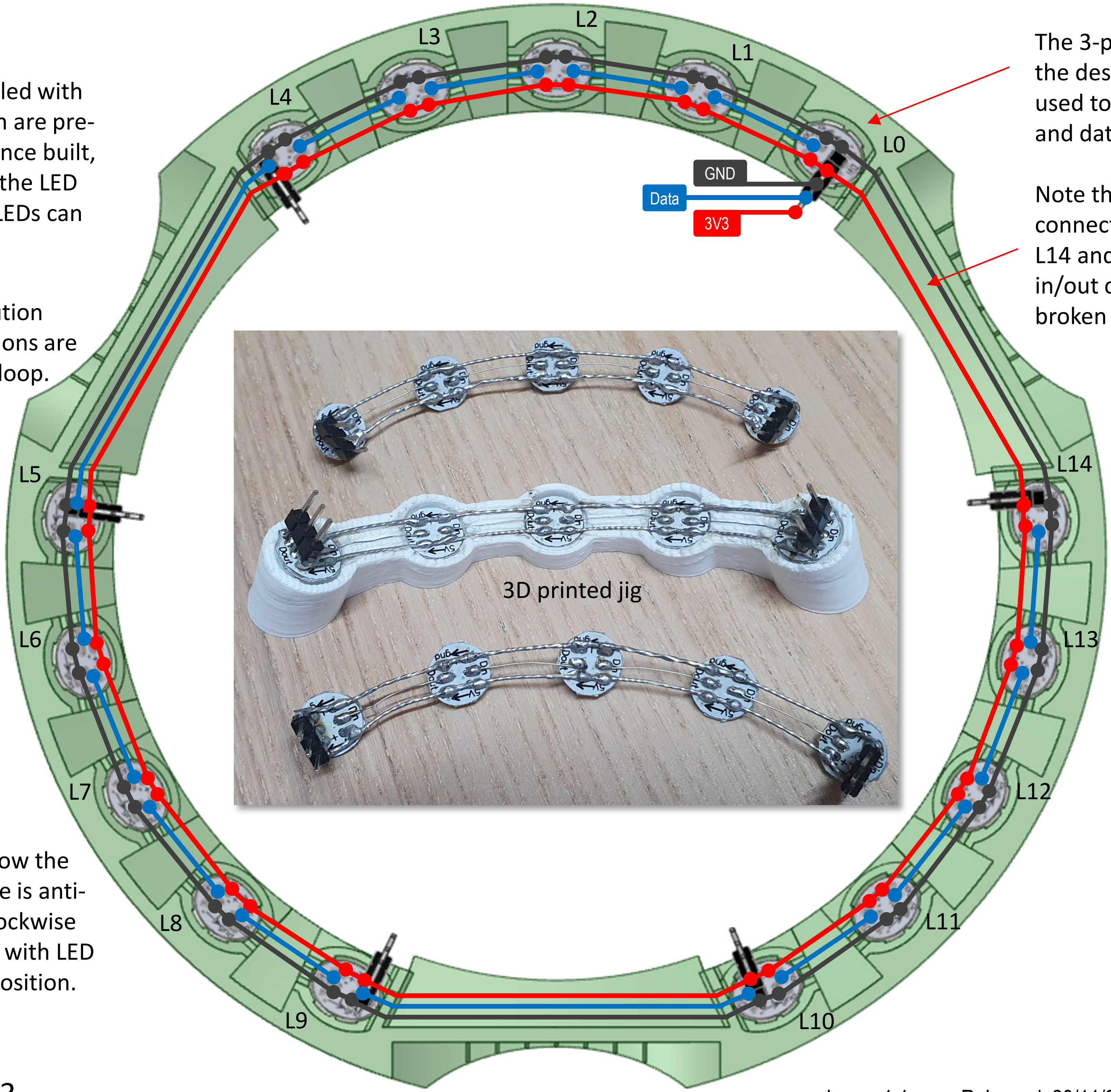


Once this is wired in, you can confirm that the sensor is working by loading and running the simple test program. This sends analogue values to the serial port of the IDE.

Base Total Wiring

The LED ring is preassembled with WS2812B LED chips, which are pre-wired using a simple jig. Once built, and glued into position in the LED ring, the three strips of 5 LEDs can be connected together.

To improve power distribution the 3v3 and GND connections are formed into a continuous loop.



The 3-pin strip, next to the designated L0 LED, is used to feed in the power and data signals.

Note there is no data connection between LEDs L14 and L0, and the data in/out connection is broken at each LED.

Since this view is from below the LED ring, the LED sequence is anti-clockwise. But it will be clockwise when viewed from above, with LED L0 being at the front left position.

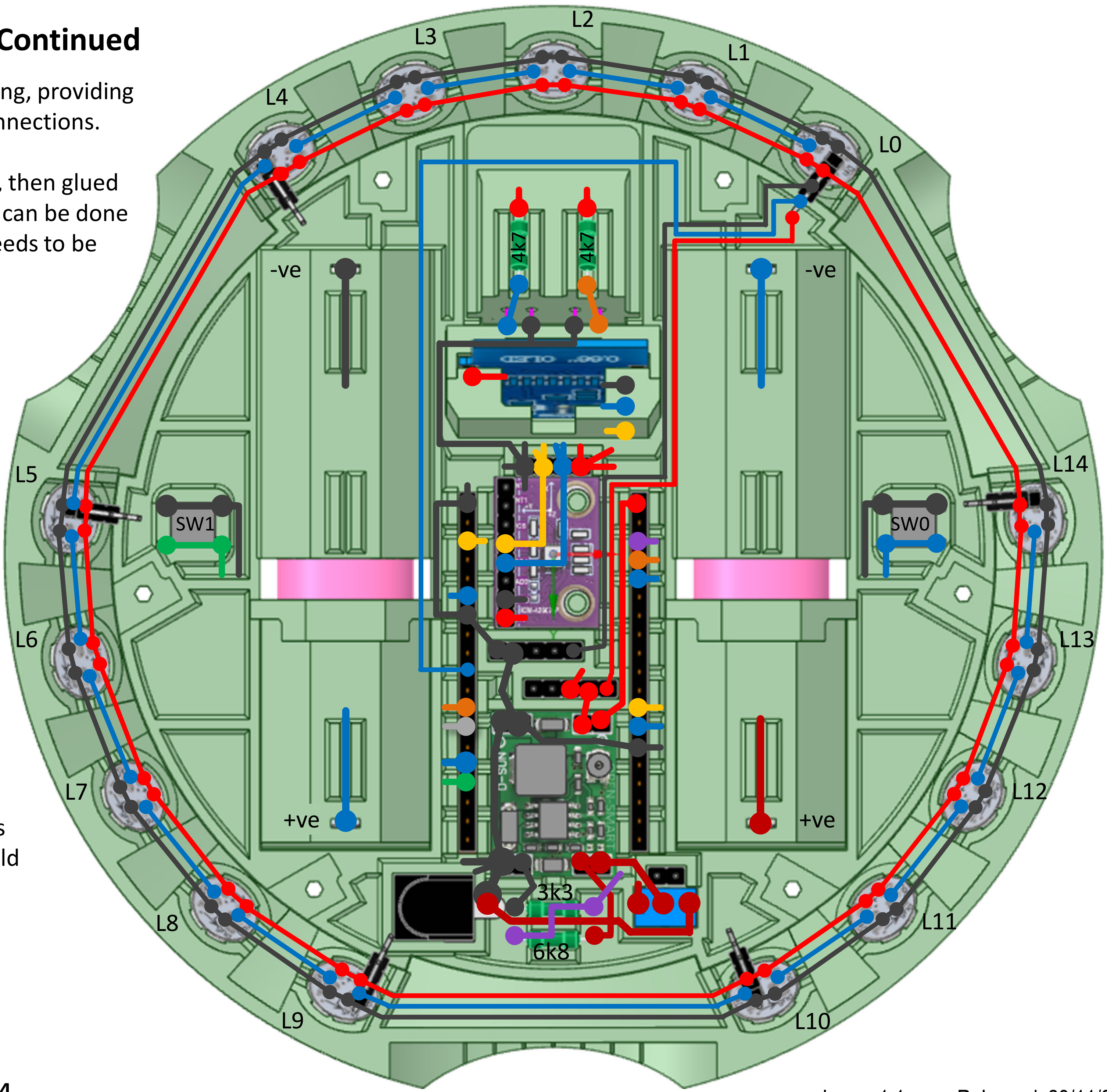
Micro Plate Wiring Continued

Next, we wire in the LED ring, providing a GND, 3v3 and data in connections.

The LED ring, is pre-wired, then glued onto the micro plate. That can be done at the start, or certainly needs to be done at this point.

To simplify the diagram, I have not routed the 3v3 and GND connections, but simply labelled them. In practise they are routed to the pin strip common connection points, in the centre of the plate.

If you want to run the LEDs quite bright then you should make additional power connections.



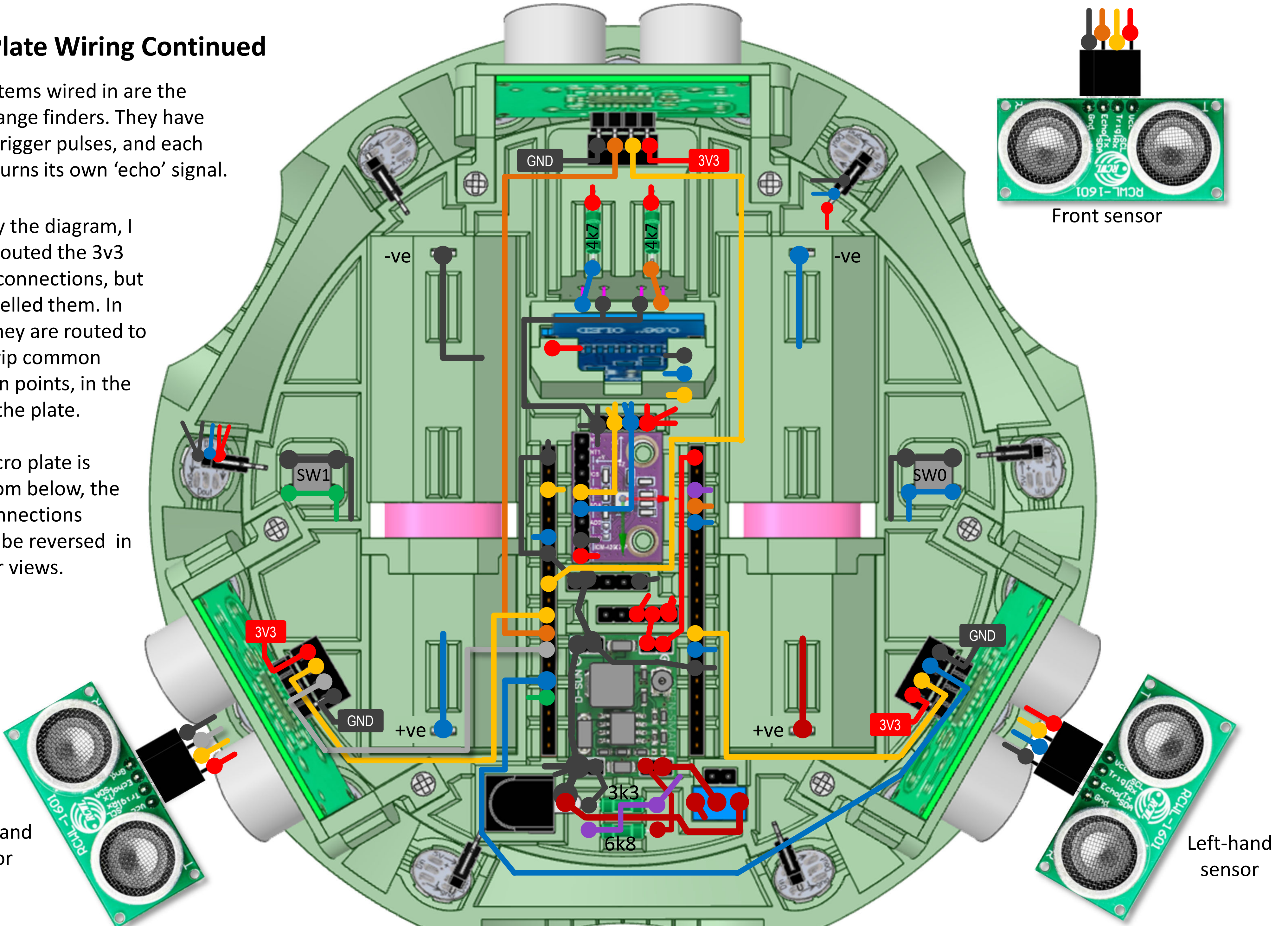
Issue: 1.1 Released: 28/11/2025 TechKnowTone

Micro Plate Wiring Continued

The next items wired in are the acoustic range finders. They have separate trigger pulses, and each sensor returns its own ‘echo’ signal.

To simplify the diagram, I have not routed the 3v3 and GND connections, but simply labelled them. In practise they are routed to the pin strip common connection points, in the centre of the plate.

As the micro plate is viewed from below, the sensor connections appear to be reversed in the sensor views.



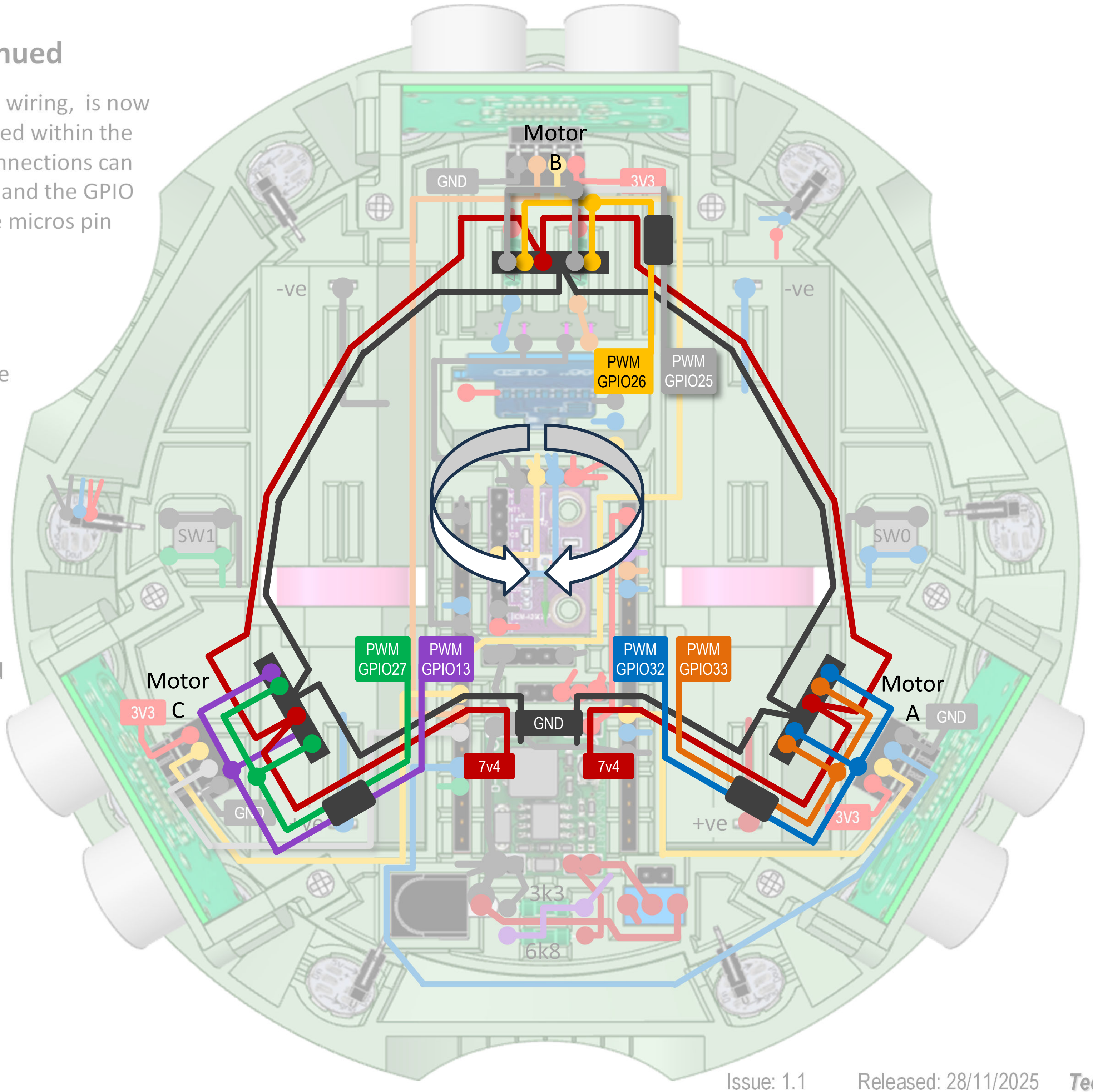
Micro Plate Wiring Continued

The harness, made for the chassis wiring, is now flipped over, left to right, and placed within the micro plate, so that the power connections can be made to the voltage regulator, and the GPIO PWM wires can be wired onto the micros pin strip connections.

In this diagram, the 6-pin sockets appear to be facing away from view, but in practise they would be facing towards you; and the chassis plate, were it in place.

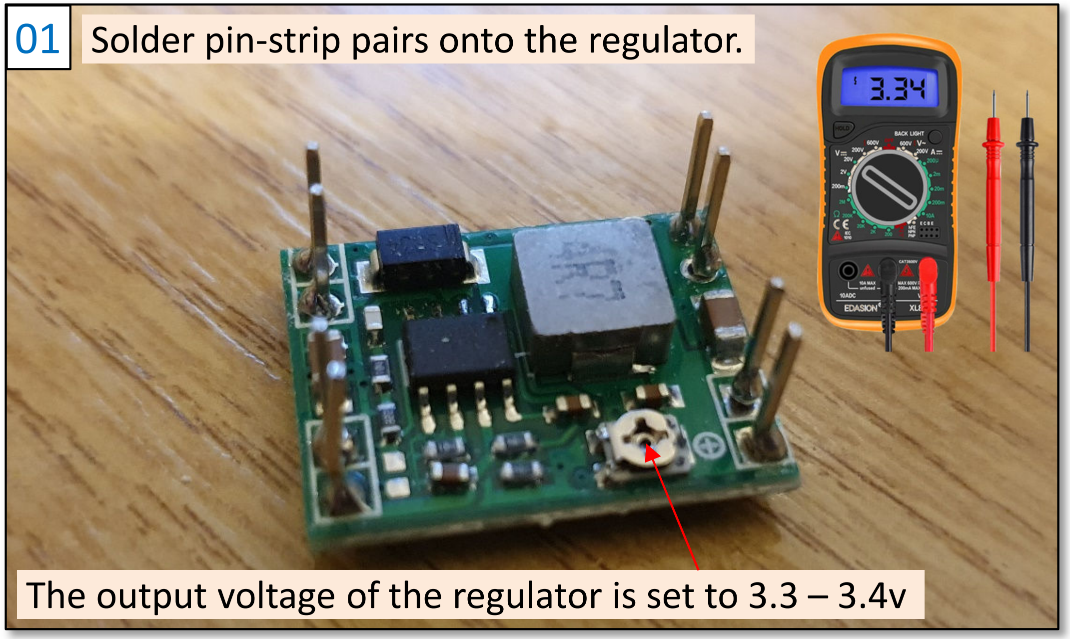
The micro plate and its wiring appear as a faint background image, in order to make the chassis wiring clearer.

Making these connections, completes the wiring process, and the Omni-Bot is ready for testing, with the full version of code loaded.



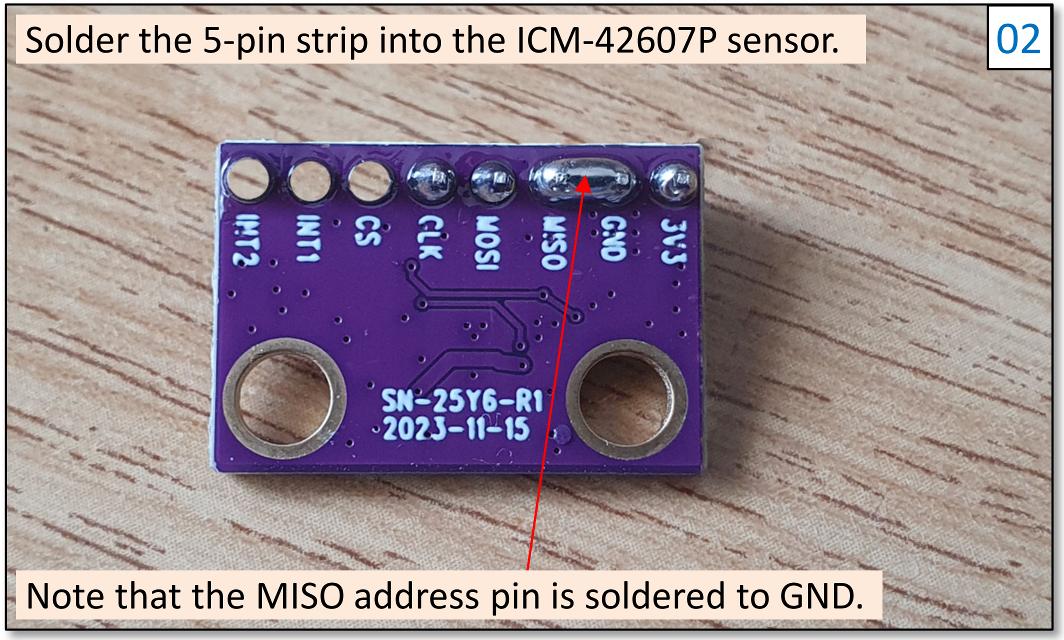
Recorded Build Sequence

01 Solder pin-strip pairs onto the regulator.



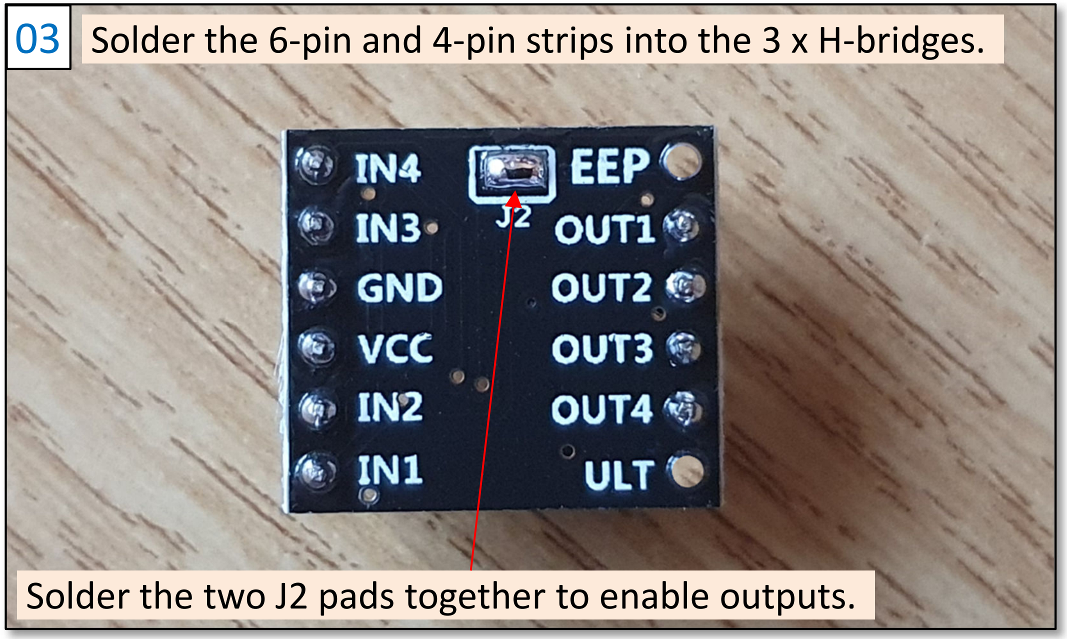
The output voltage of the regulator is set to 3.3 – 3.4v

02 Solder the 5-pin strip into the ICM-42607P sensor.



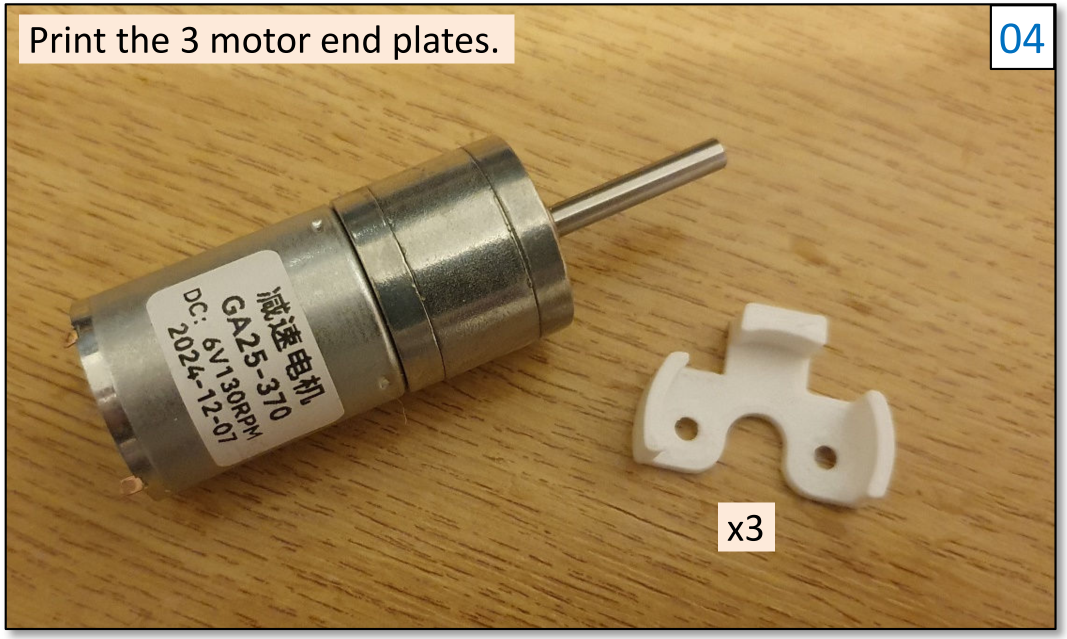
Note that the MISO address pin is soldered to GND.

03 Solder the 6-pin and 4-pin strips into the 3 x H-bridges.



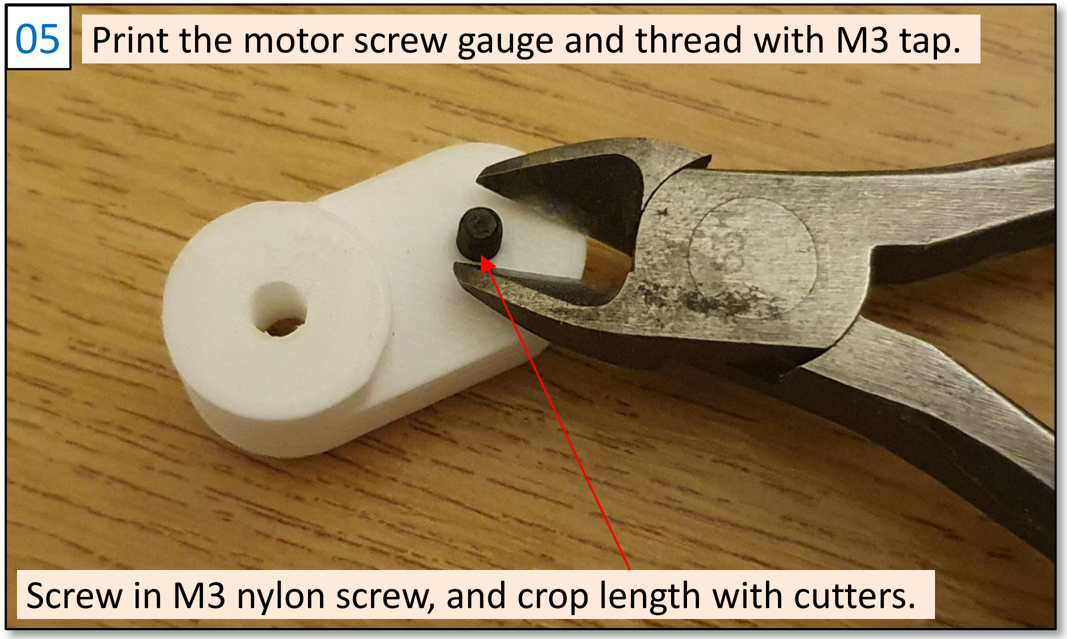
Solder the two J2 pads together to enable outputs.

04 Print the 3 motor end plates.

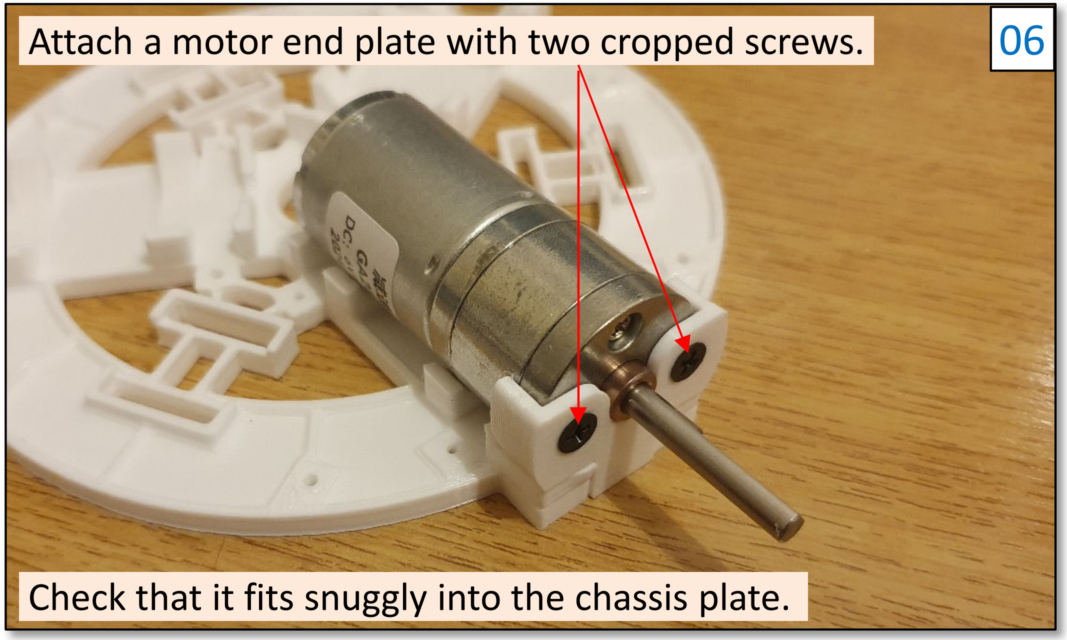


x3

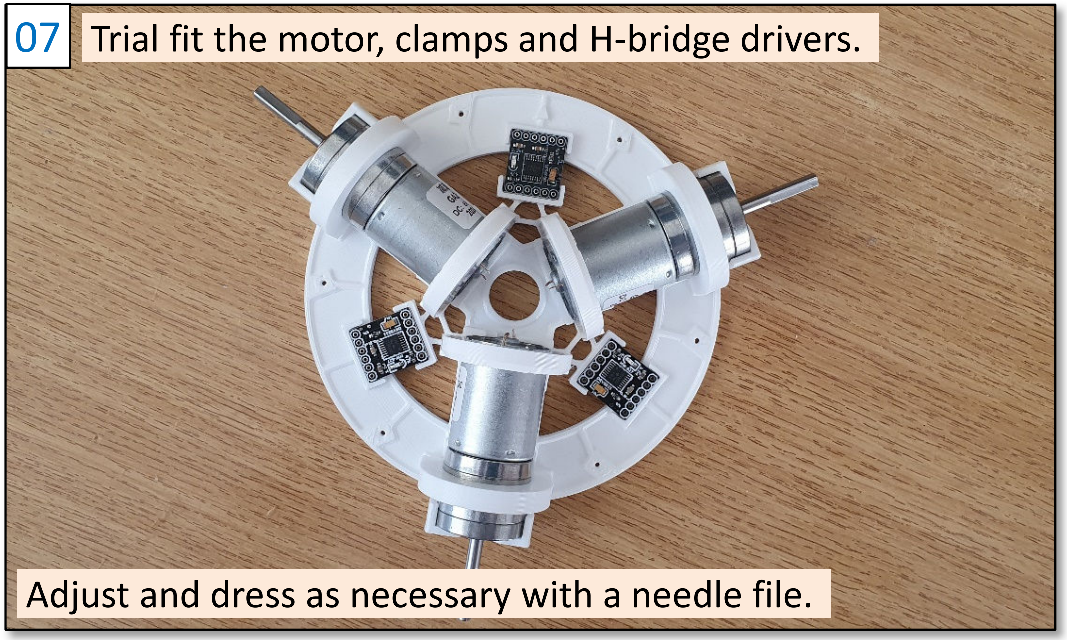
Recorded Build Sequence


05 Print the motor screw gauge and thread with M3 tap.

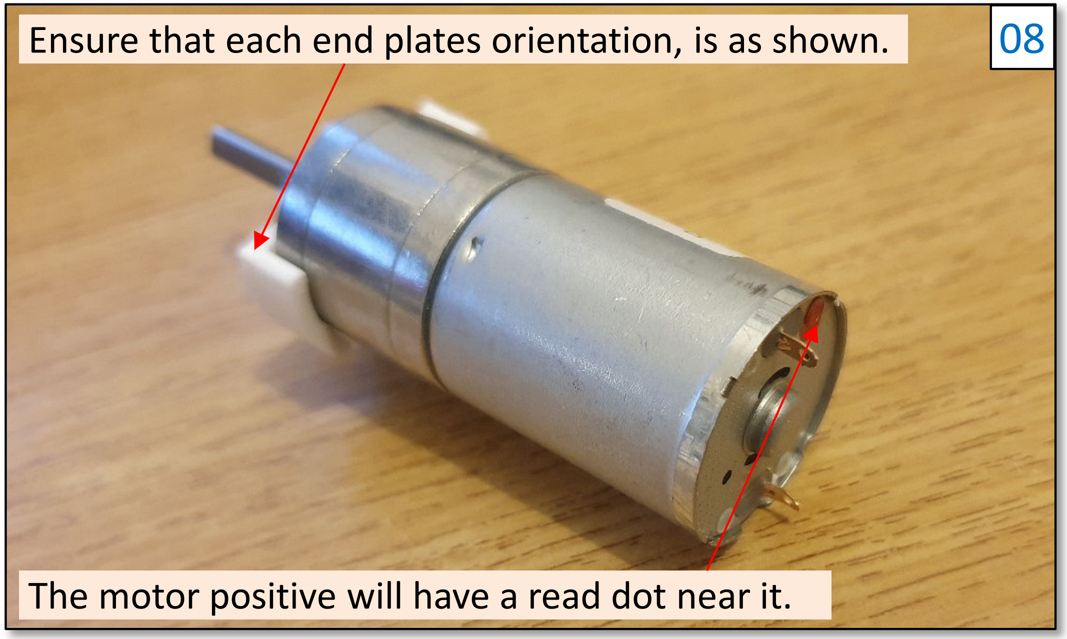
Screw in M3 nylon screw, and crop length with cutters.


06 Attach a motor end plate with two cropped screws.

Check that it fits snuggly into the chassis plate.


07 Trial fit the motor, clamps and H-bridge drivers.

Adjust and dress as necessary with a needle file.

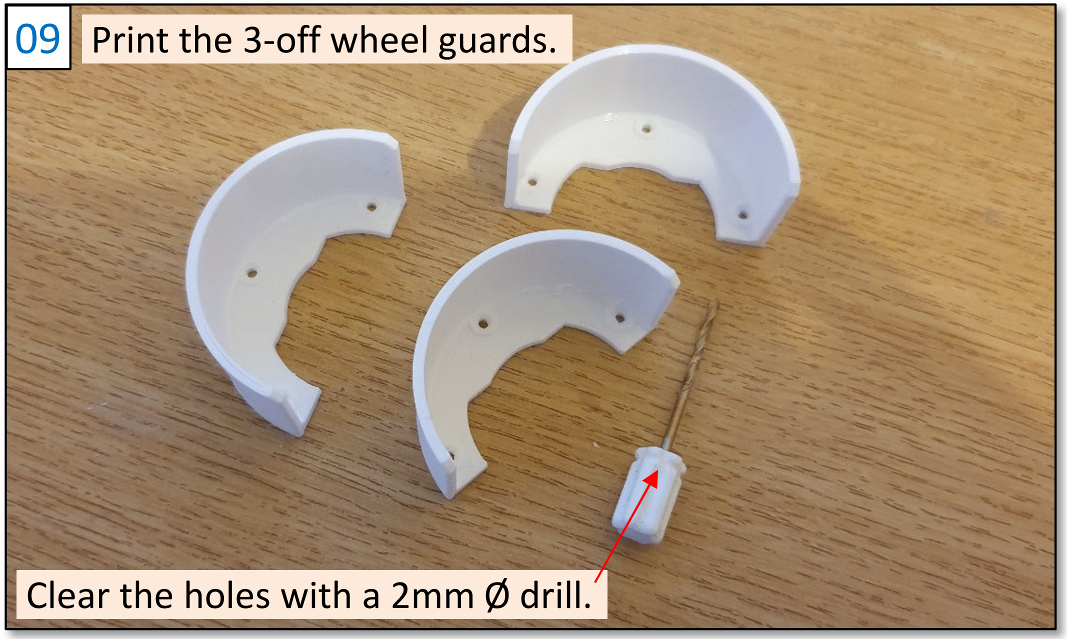

08 Ensure that each end plates orientation, is as shown.

The motor positive will have a read dot near it.

Recorded Build Sequence

09

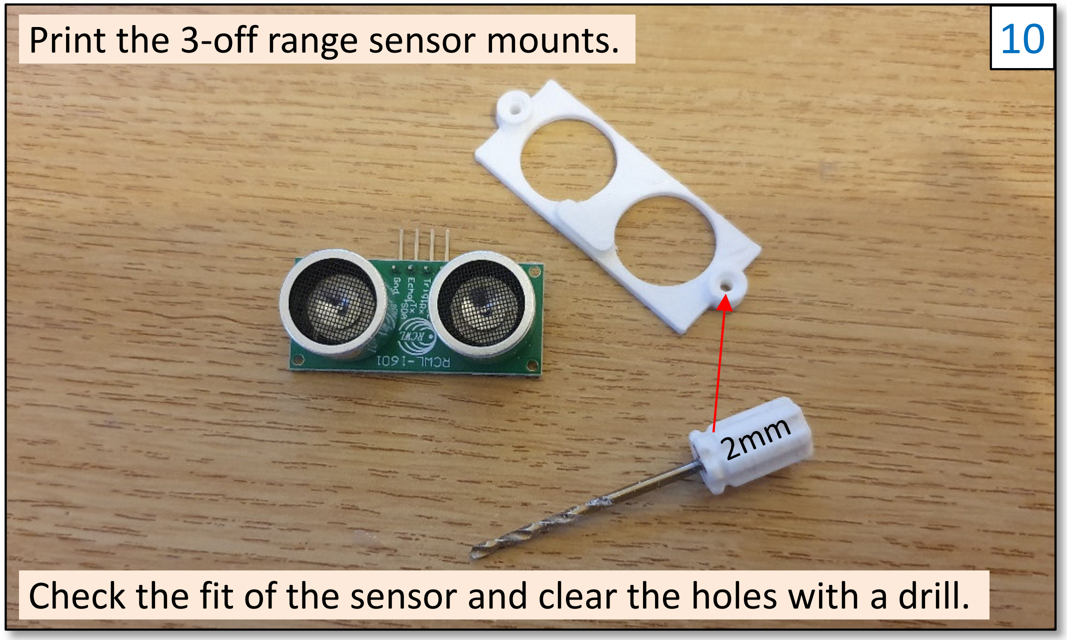
Print the 3-off wheel guards.



Clear the holes with a 2mm Ø drill.

10

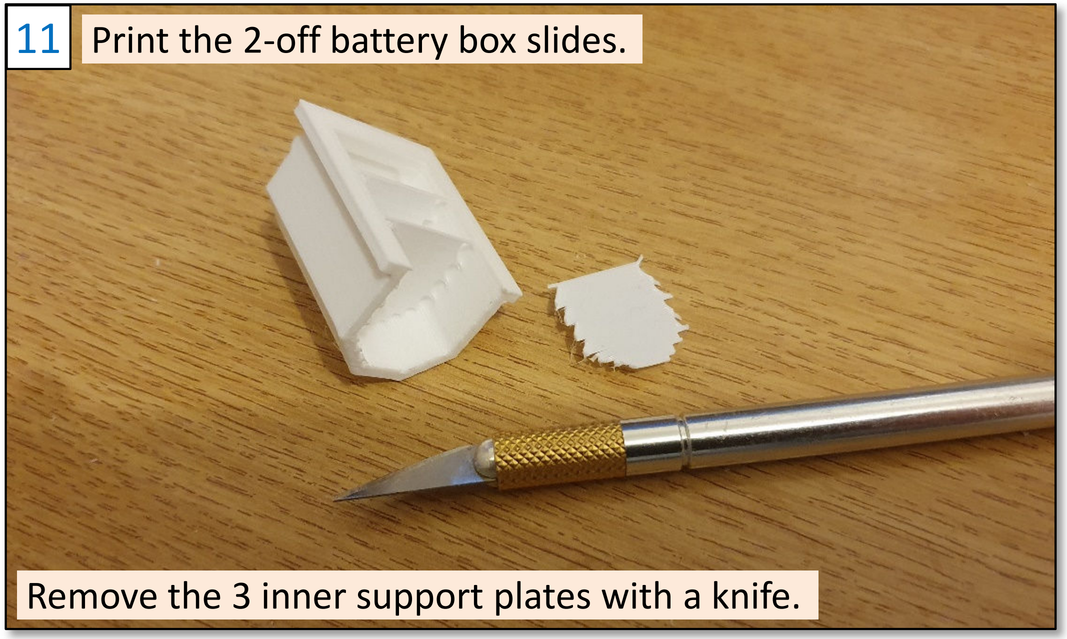
Print the 3-off range sensor mounts.



Check the fit of the sensor and clear the holes with a drill.

11

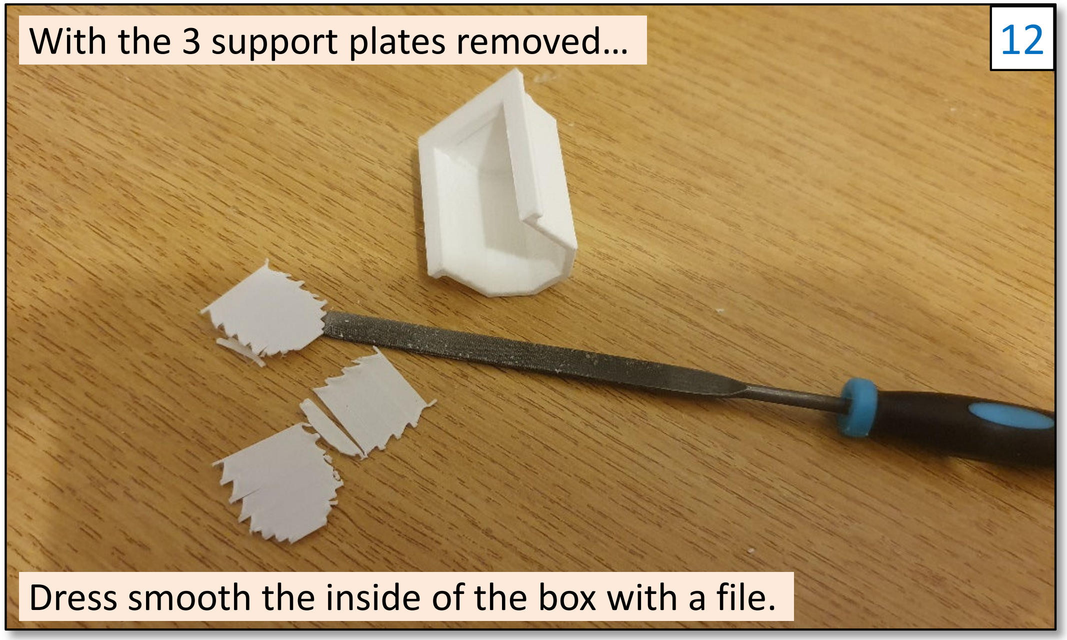
Print the 2-off battery box slides.



Remove the 3 inner support plates with a knife.

12

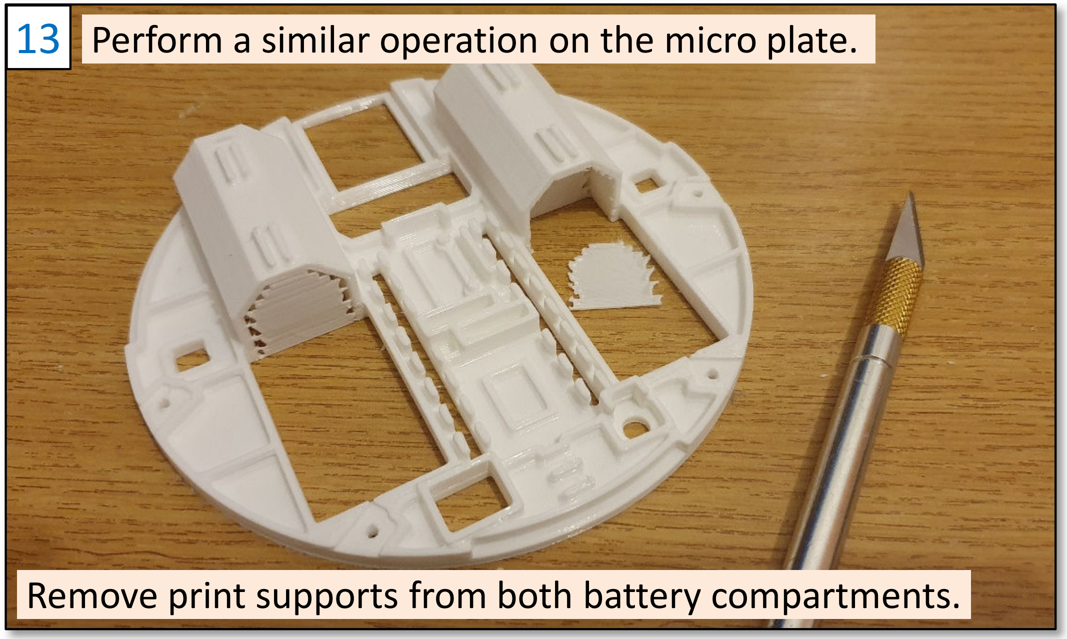
With the 3 support plates removed…



Dress smooth the inside of the box with a file.

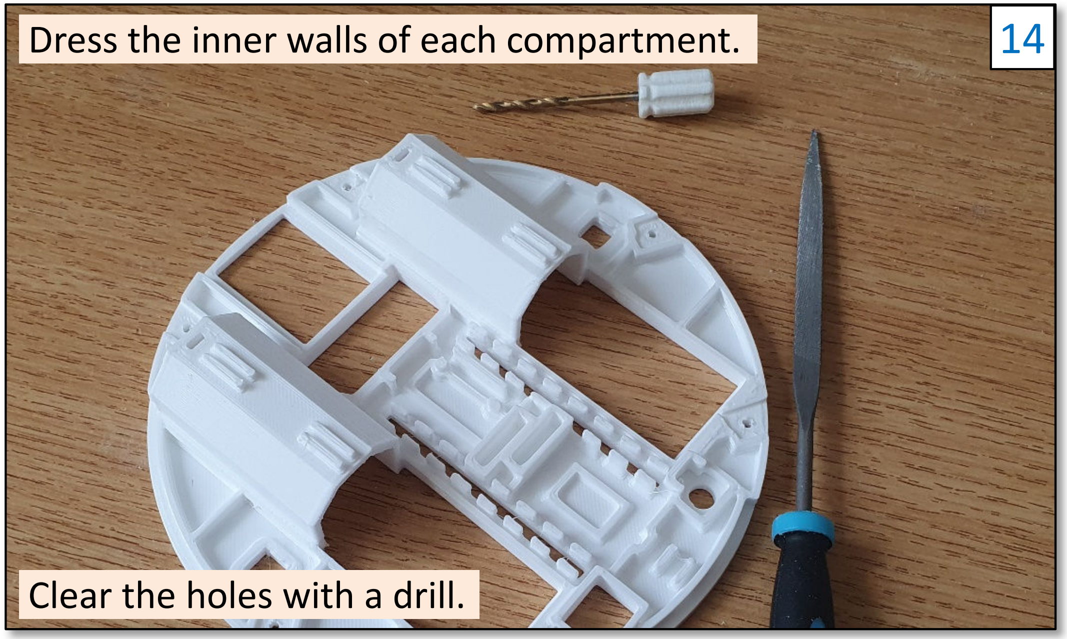
Recorded Build Sequence

13 Perform a similar operation on the micro plate.



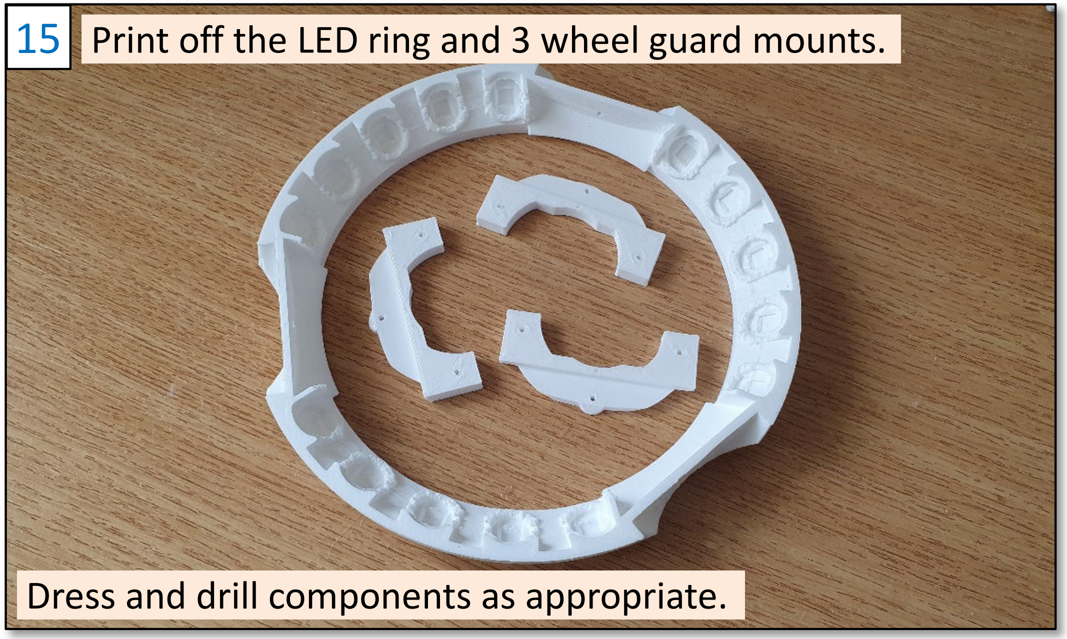
Remove print supports from both battery compartments.

14 Dress the inner walls of each compartment.



Clear the holes with a drill.

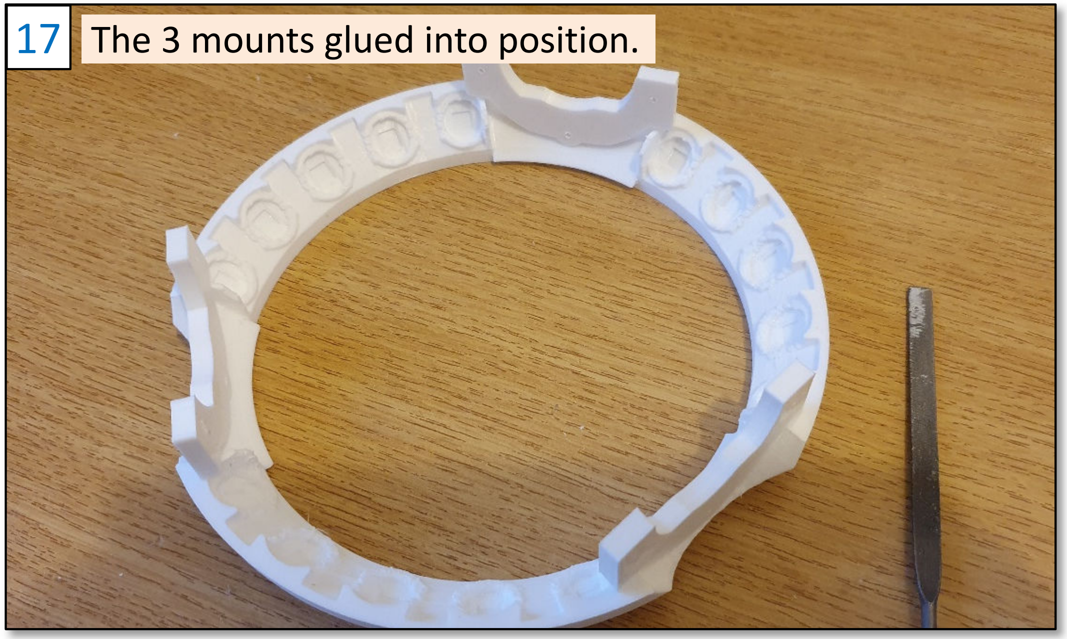
15 Print off the LED ring and 3 wheel guard mounts.



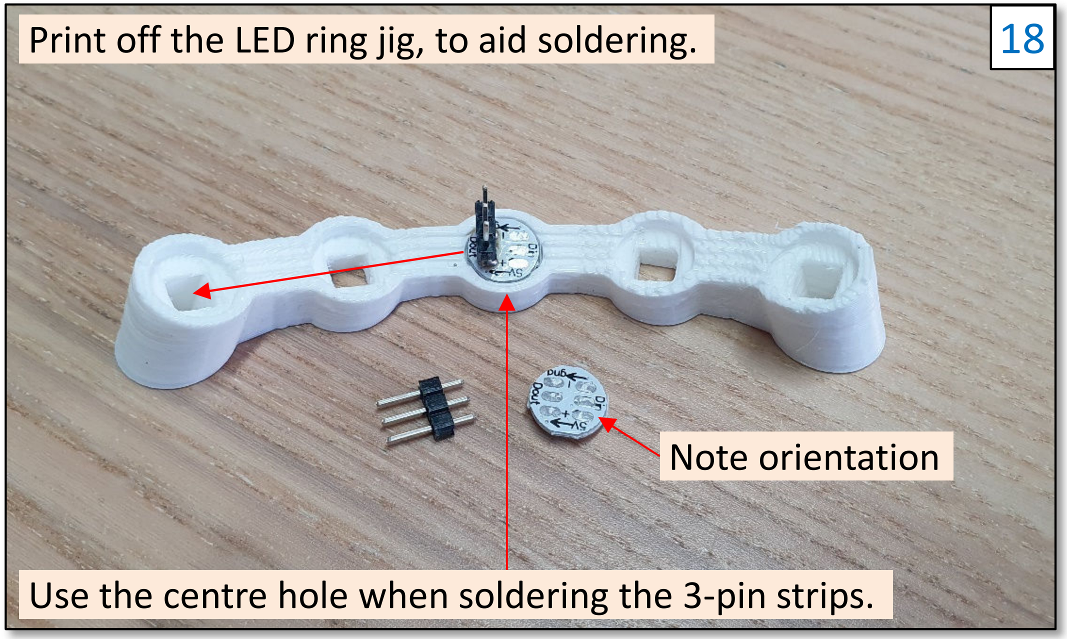
Dress and drill components as appropriate.

16 Apply glue and clamp guard mounts in position.

Recorded Build Sequence

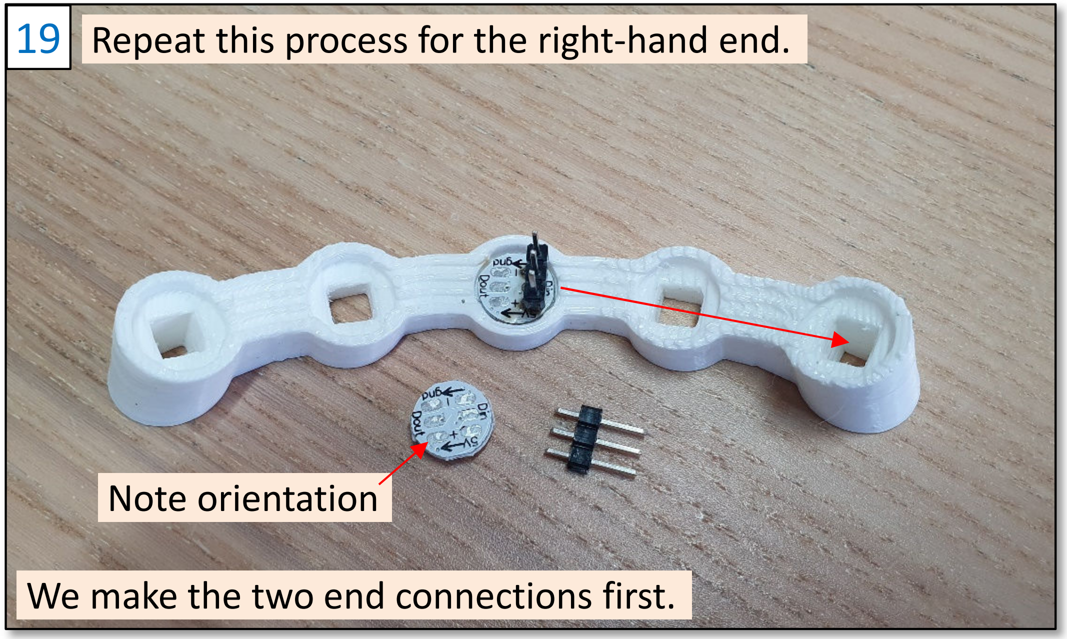


17 The 3 mounts glued into position.



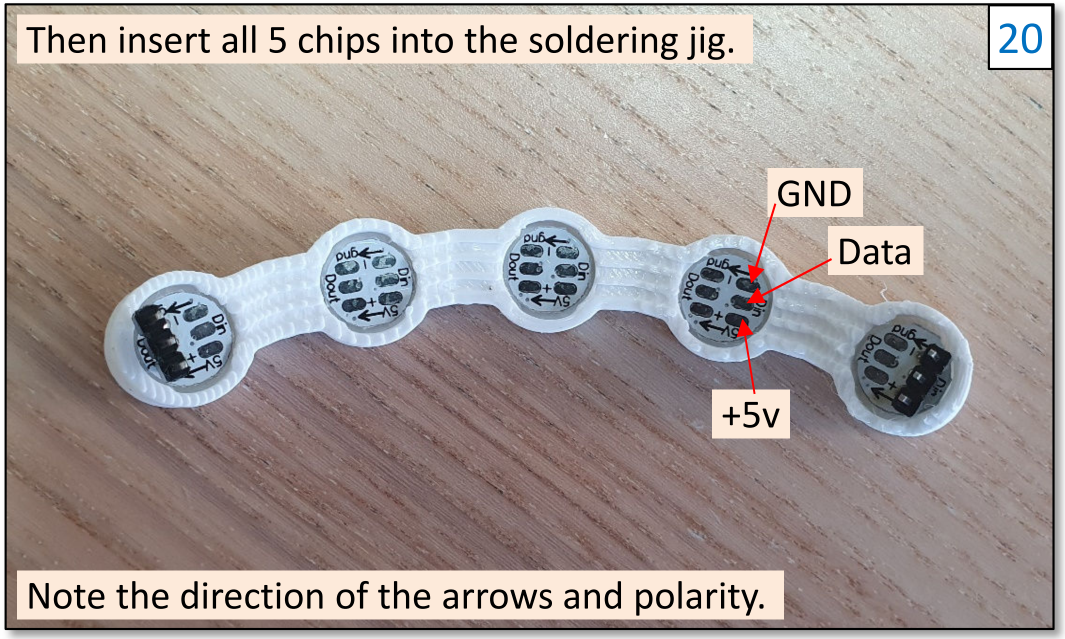
18 Print off the LED ring jig, to aid soldering.

Use the centre hole when soldering the 3-pin strips.



19 Repeat this process for the right-hand end.

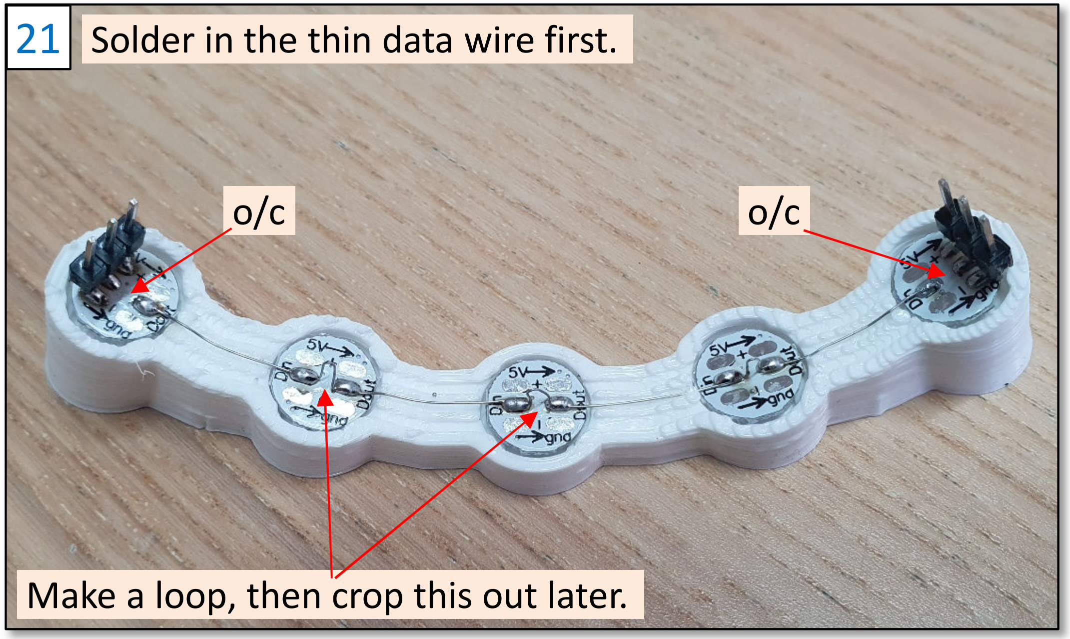
We make the two end connections first.



20 Then insert all 5 chips into the soldering jig.

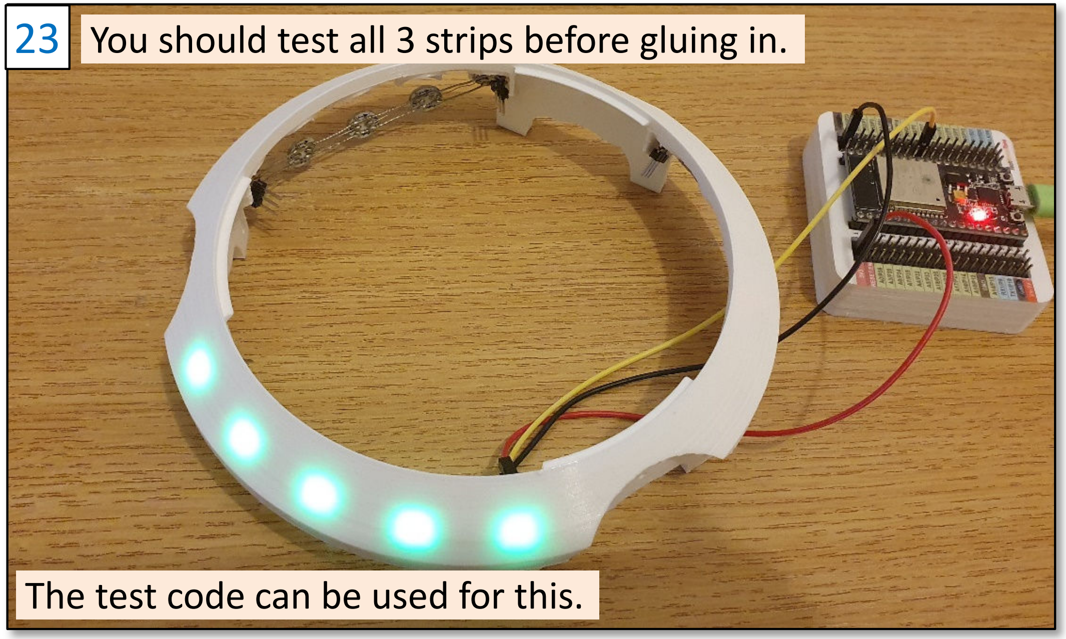
Note the direction of the arrows and polarity.

Recorded Build Sequence

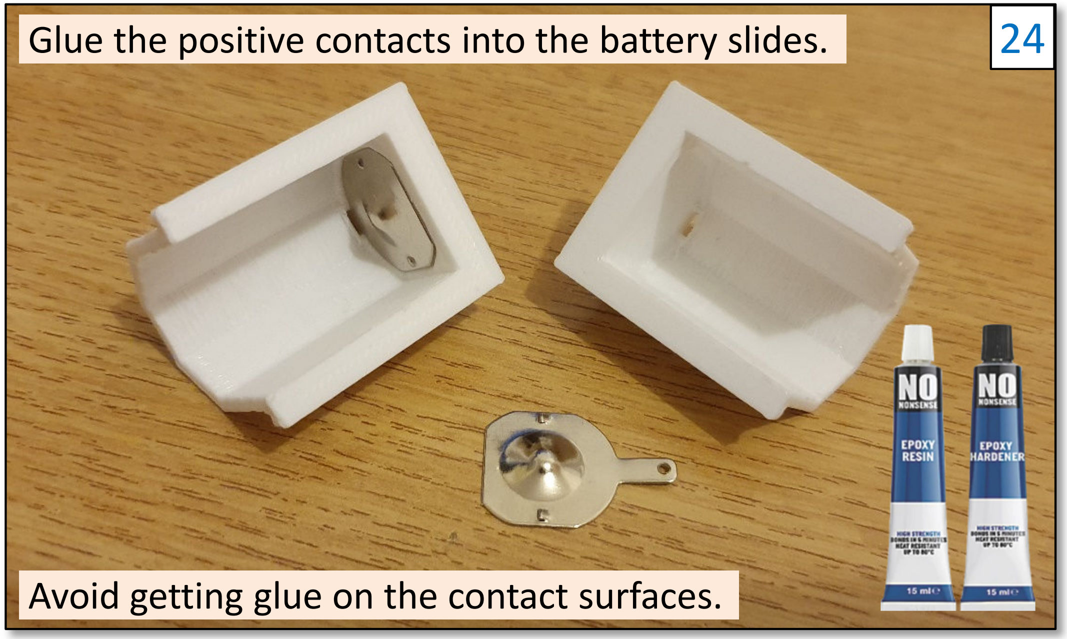


Solder in the thin data wire first.
Make a loop, then crop this out later.

Then solder in the thicker +5v and GND wires.
Complete the 3 sets as shown.



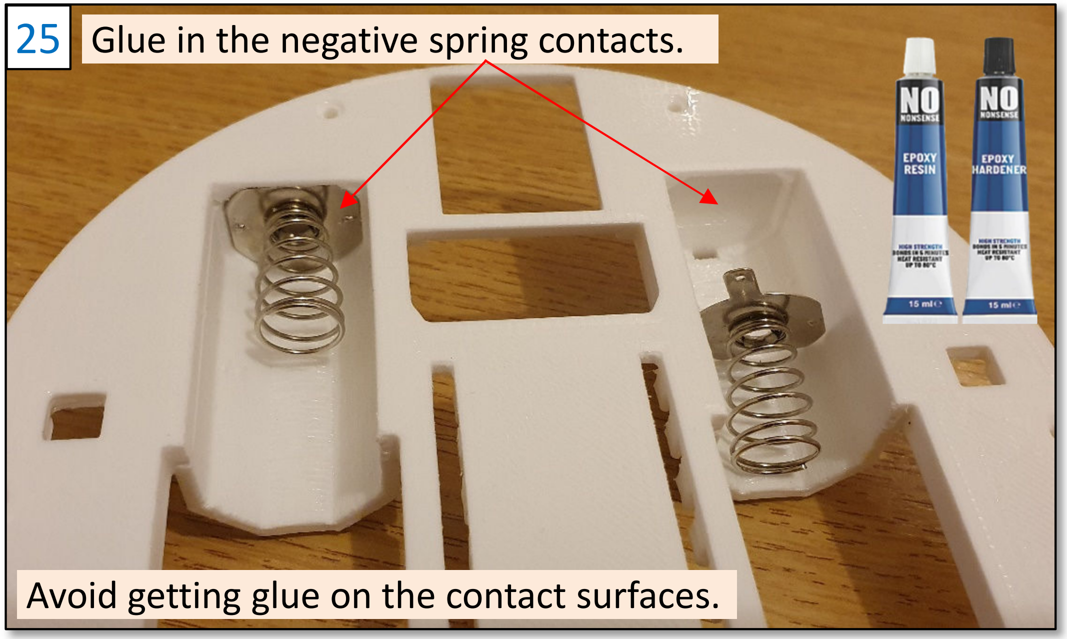
You should test all 3 strips before gluing in.
The test code can be used for this.



Glue the positive contacts into the battery slides.
Avoid getting glue on the contact surfaces.

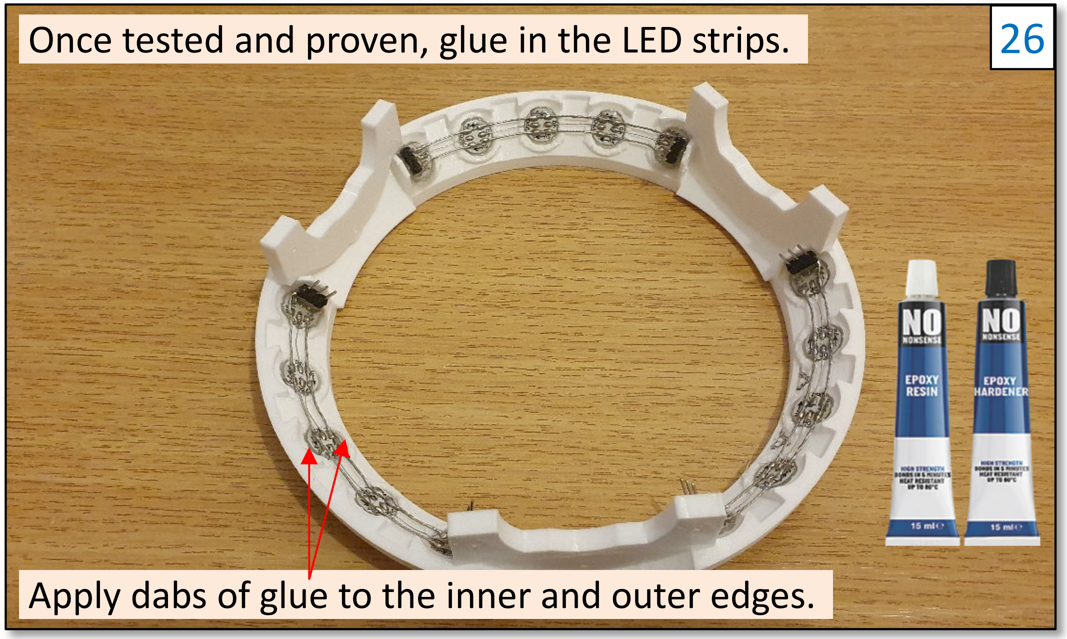
Recorded Build Sequence

25 Glue in the negative spring contacts.



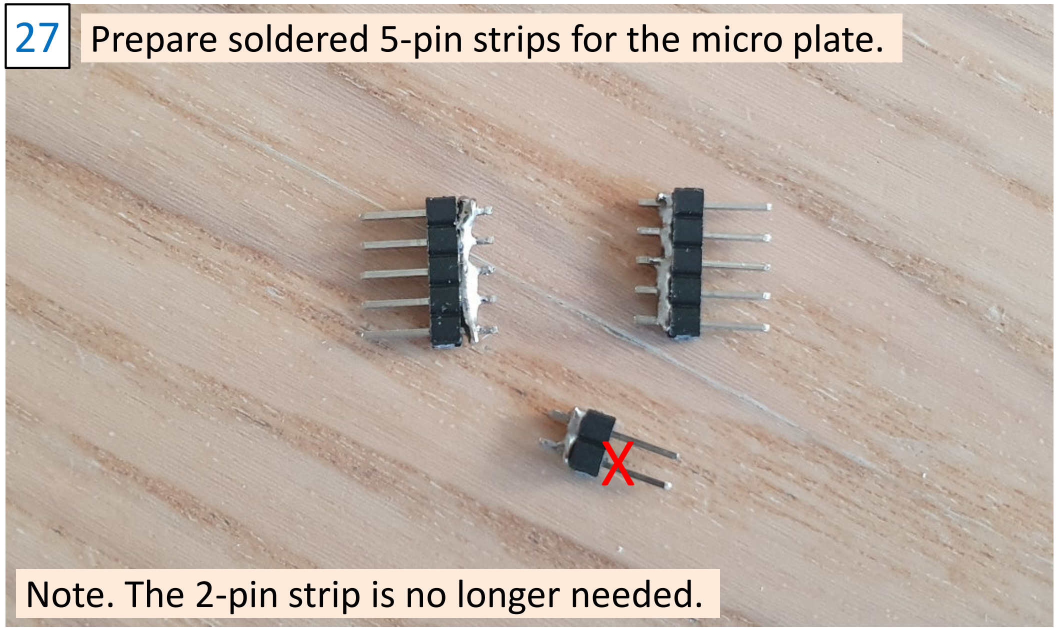
Avoid getting glue on the contact surfaces.

26 Once tested and proven, glue in the LED strips.



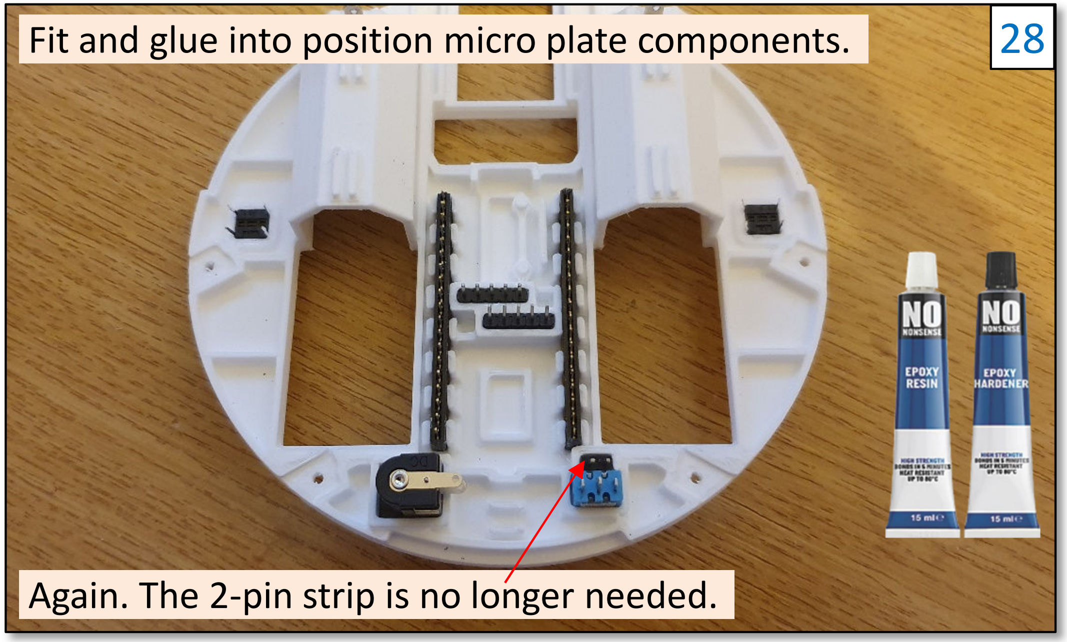
Apply dabs of glue to the inner and outer edges.

27 Prepare soldered 5-pin strips for the micro plate.



Note. The 2-pin strip is no longer needed.

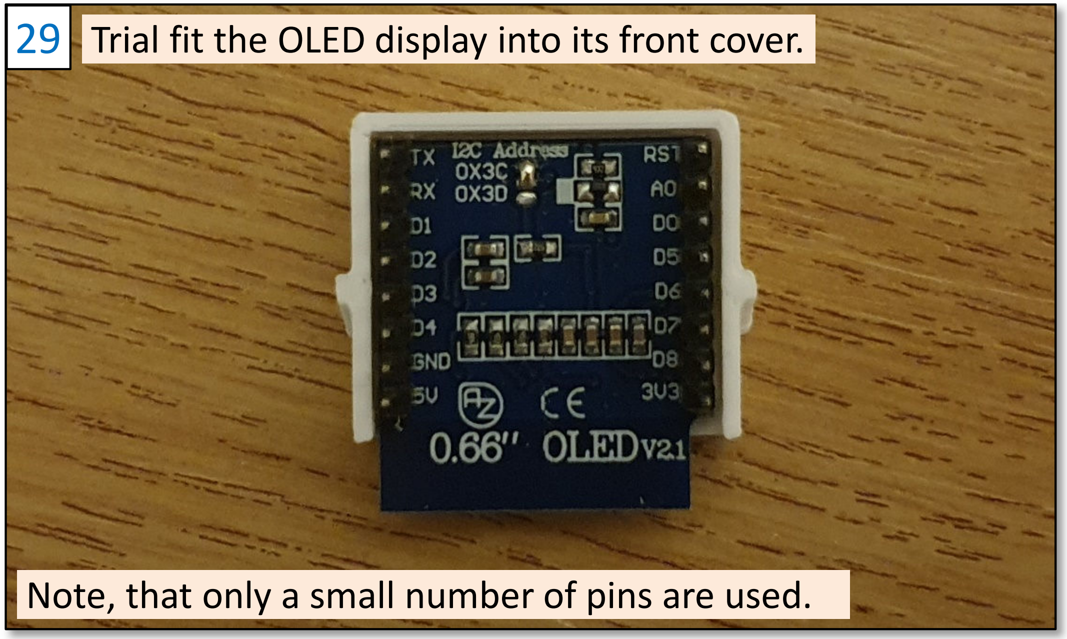
28 Fit and glue into position micro plate components.



Again. The 2-pin strip is no longer needed.

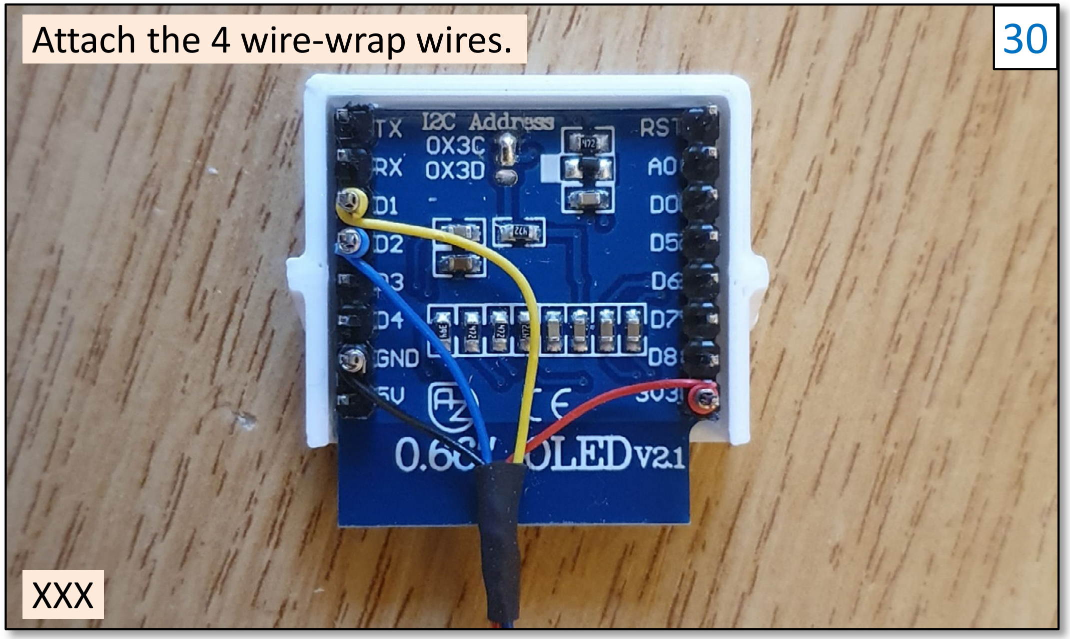
Recorded Build Sequence

29 Trial fit the OLED display into its front cover.



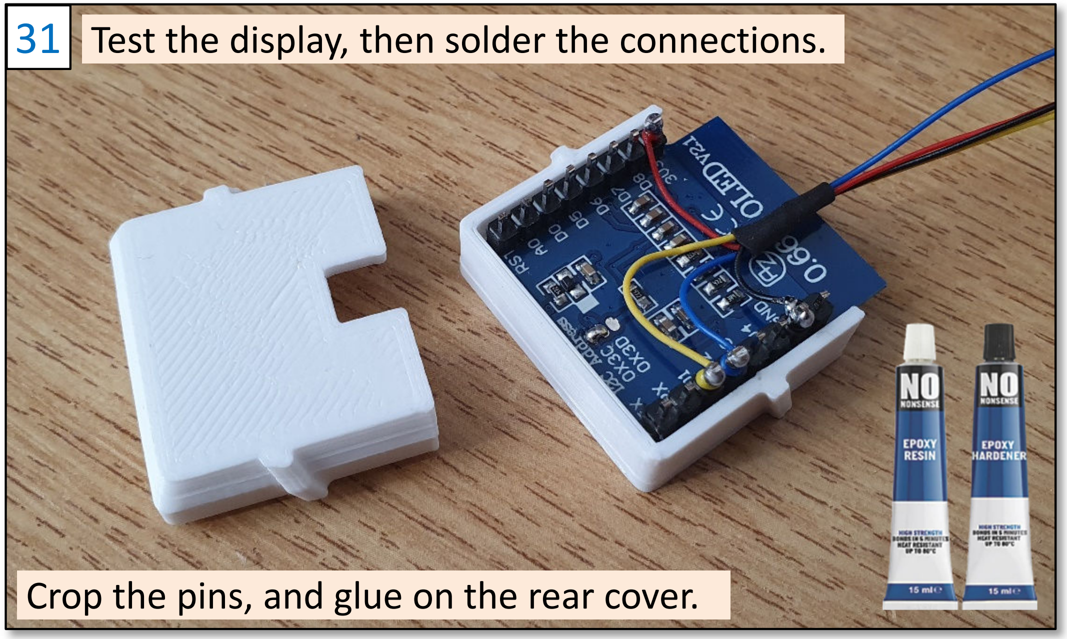
Note, that only a small number of pins are used.

30 Attach the 4 wire-wrap wires.



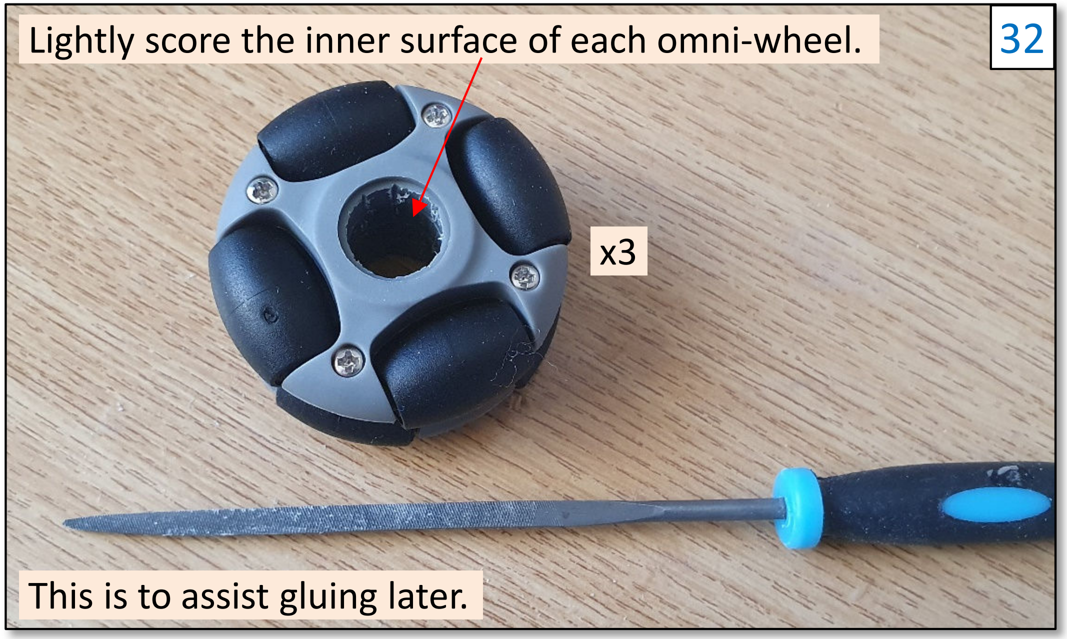
XXX

31 Test the display, then solder the connections.



Crop the pins, and glue on the rear cover.

32 Lightly score the inner surface of each omni-wheel.

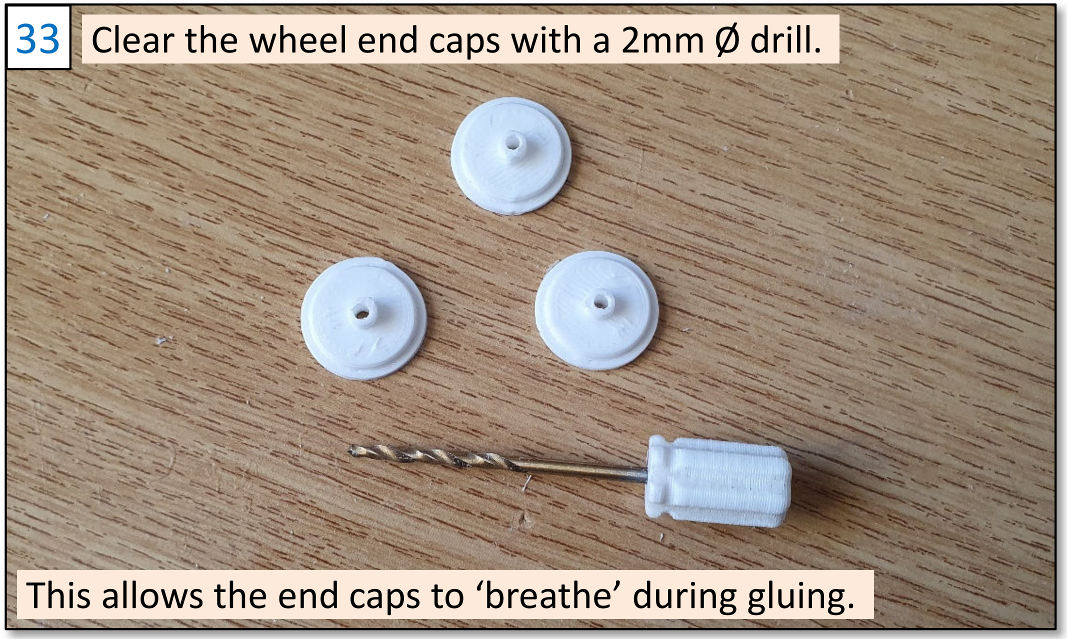


x3

This is to assist gluing later.

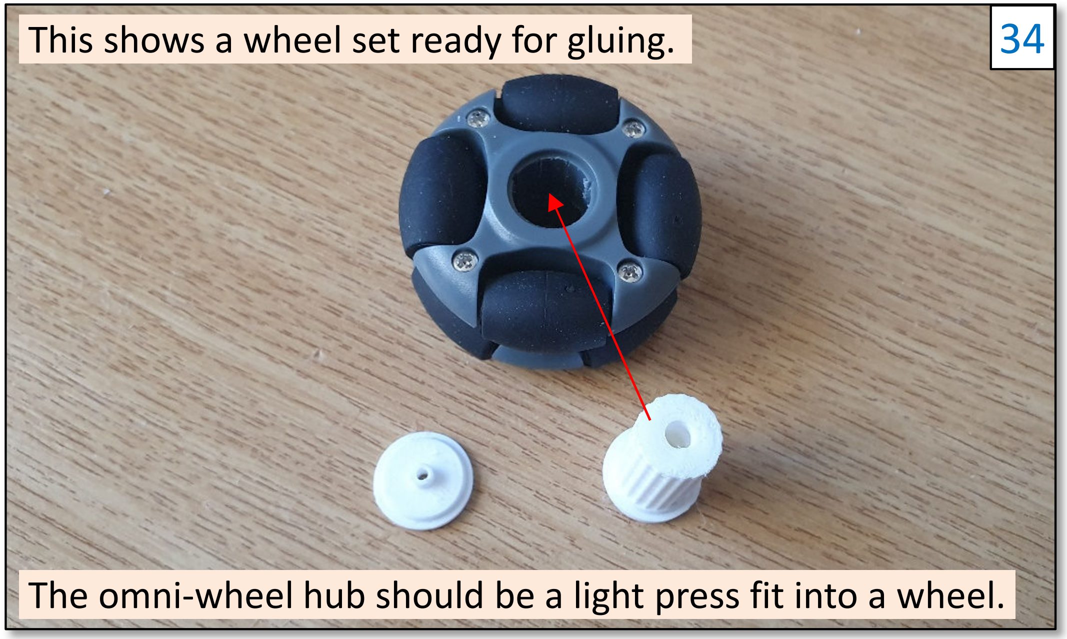
Recorded Build Sequence

33 Clear the wheel end caps with a 2mm Ø drill.



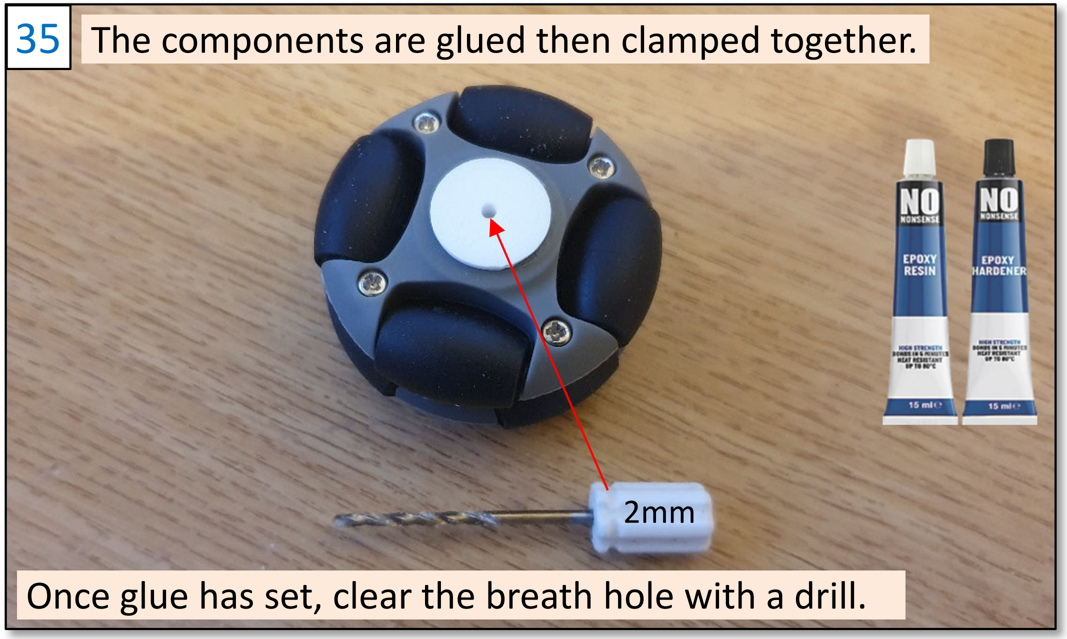
This allows the end caps to ‘breathe’ during gluing.

34 This shows a wheel set ready for gluing.



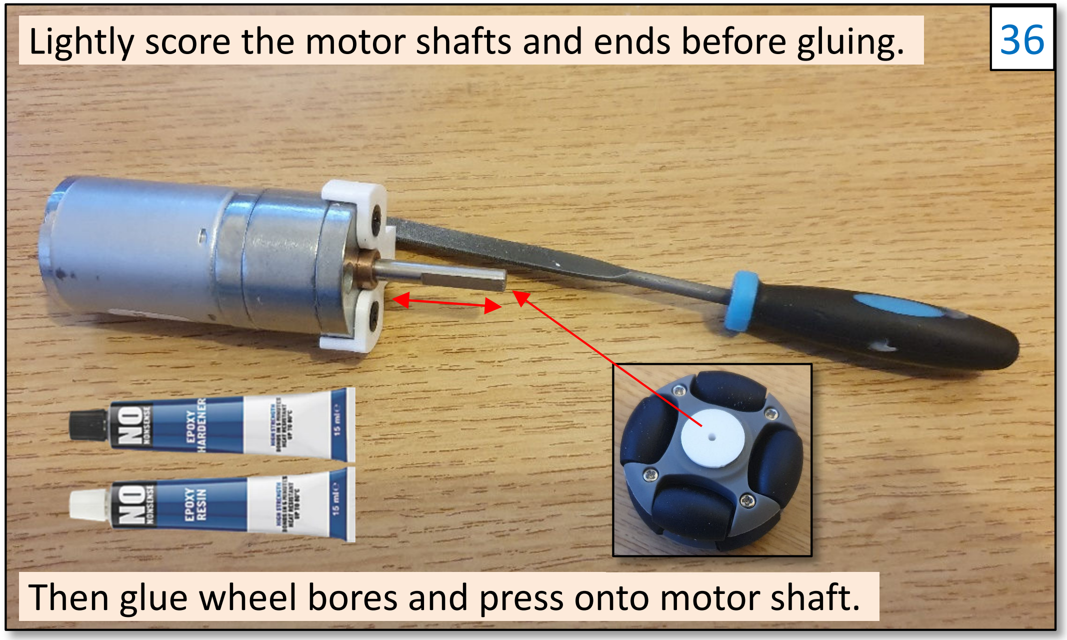
The omni-wheel hub should be a light press fit into a wheel.

35 The components are glued then clamped together.



Once glue has set, clear the breath hole with a drill.

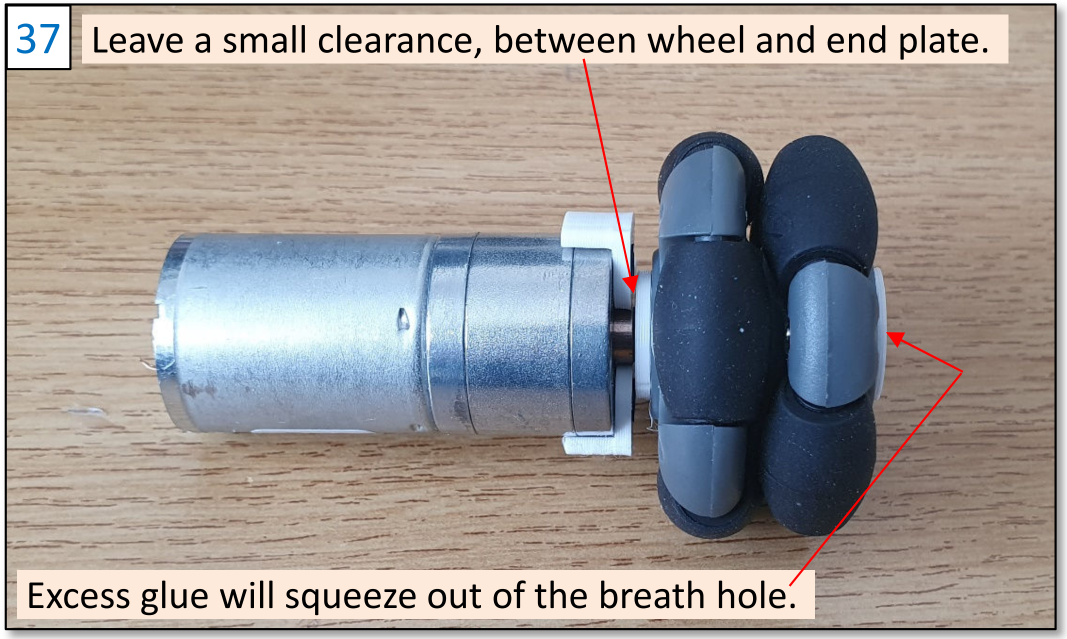
36 Lightly score the motor shafts and ends before gluing.



Then glue wheel bores and press onto motor shaft.

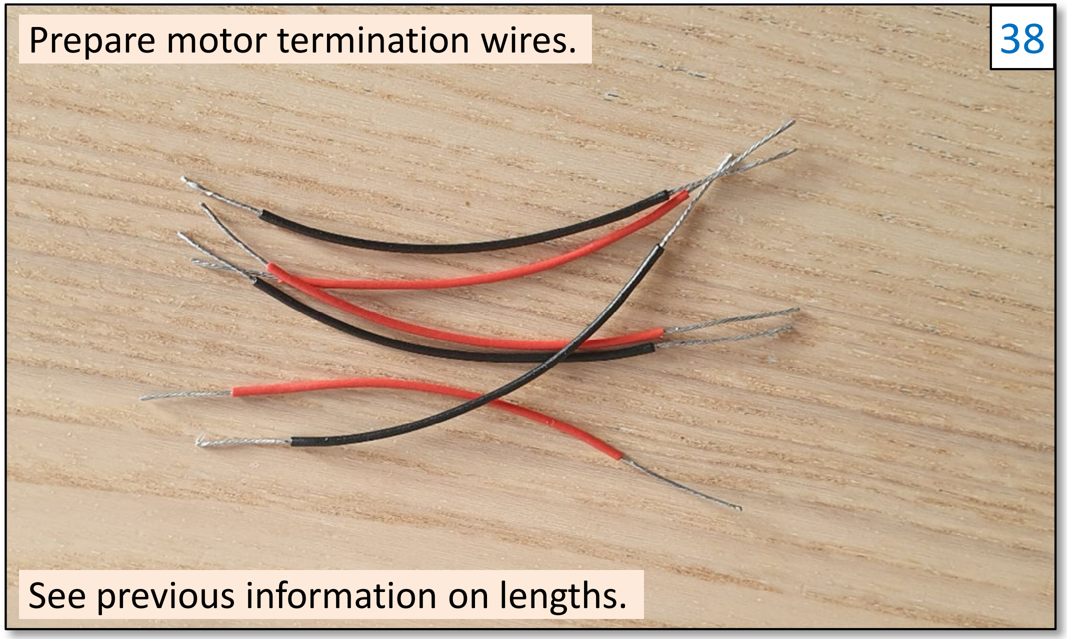
Recorded Build Sequence

37 Leave a small clearance, between wheel and end plate.



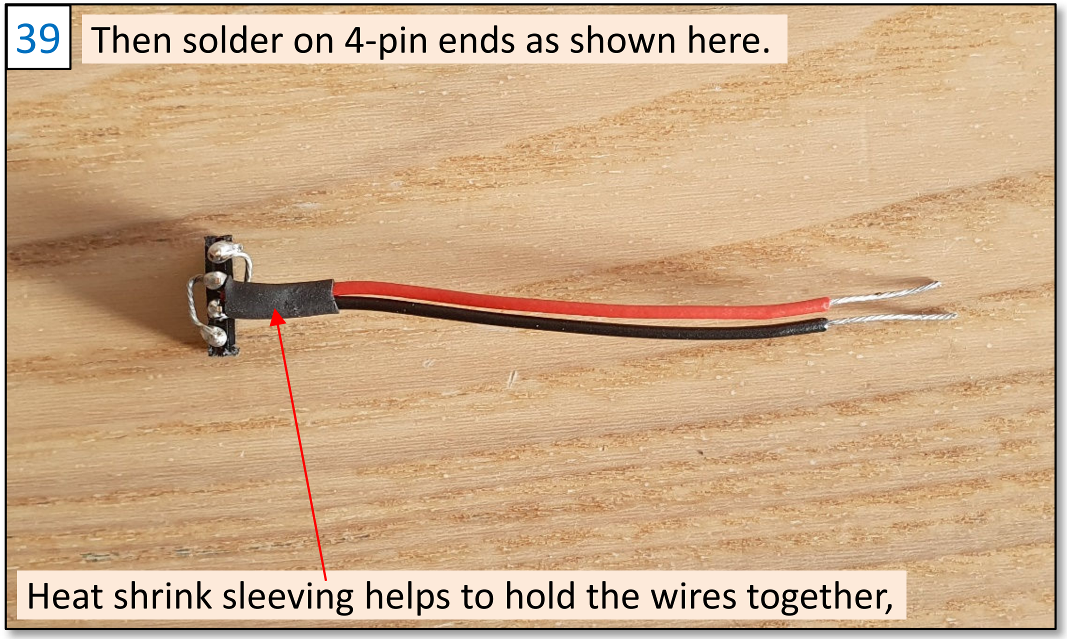
Excess glue will squeeze out of the breath hole.

38 Prepare motor termination wires.



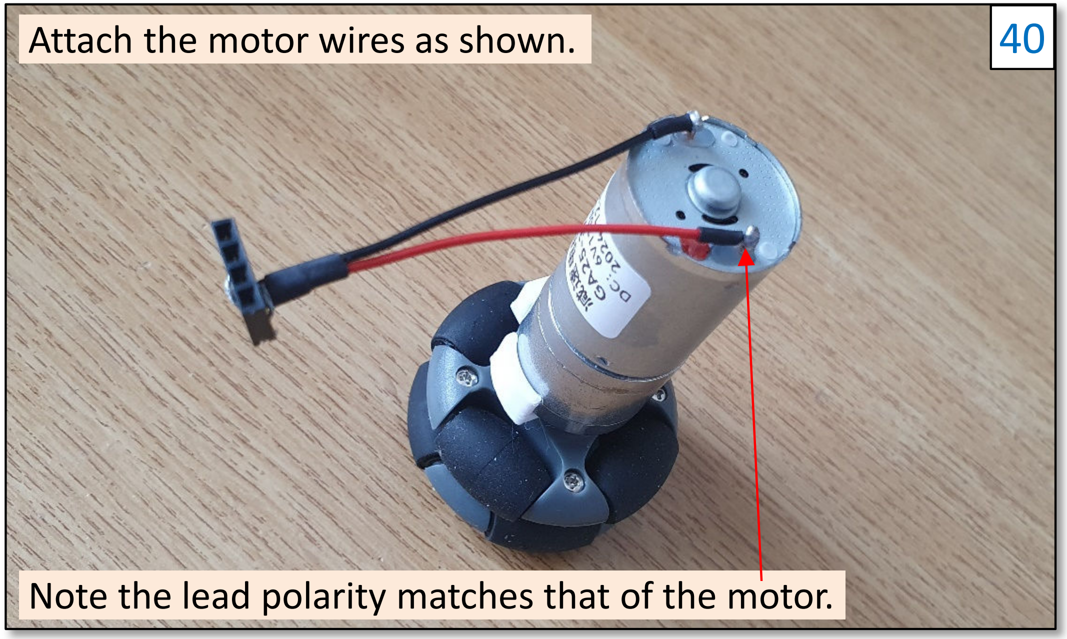
See previous information on lengths.

39 Then solder on 4-pin ends as shown here.



Heat shrink sleeving helps to hold the wires together,

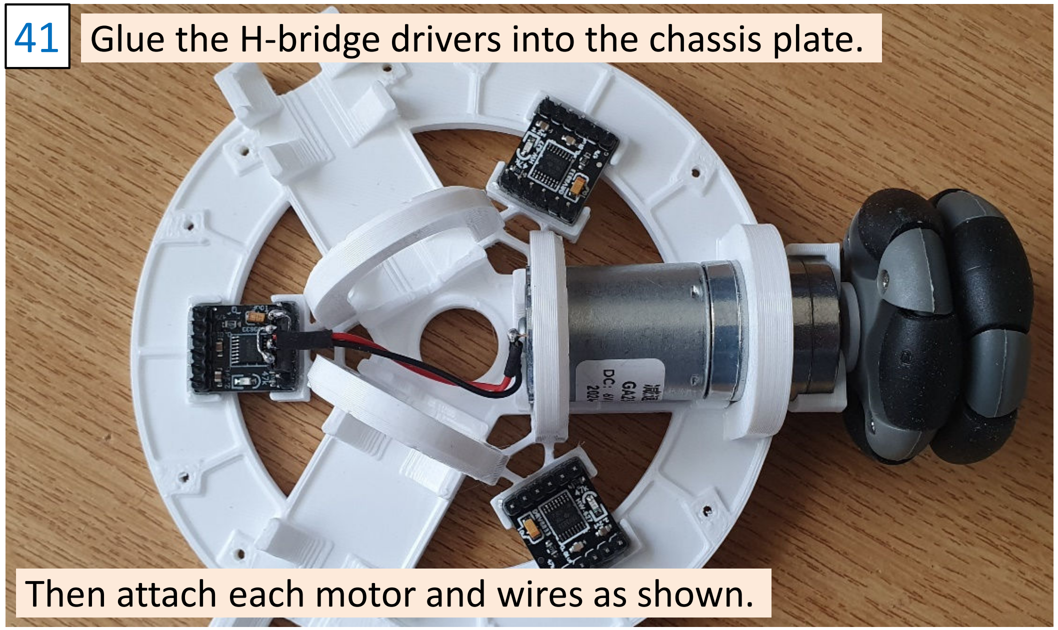
40 Attach the motor wires as shown.



Note the lead polarity matches that of the motor.

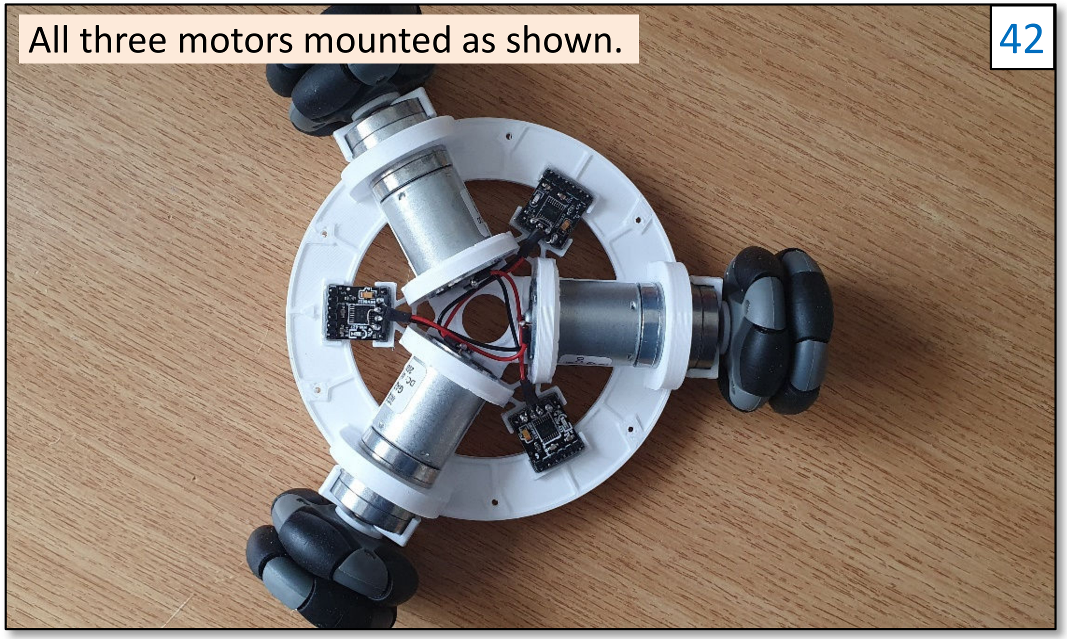
Recorded Build Sequence

41 Glue the H-bridge drivers into the chassis plate.

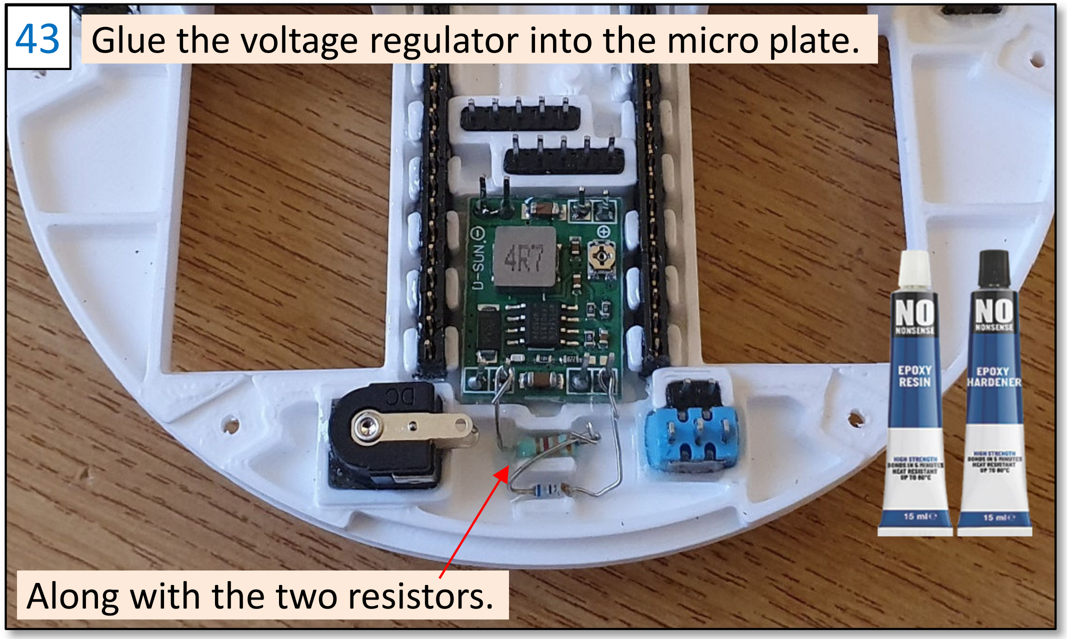


Then attach each motor and wires as shown.

42 All three motors mounted as shown.

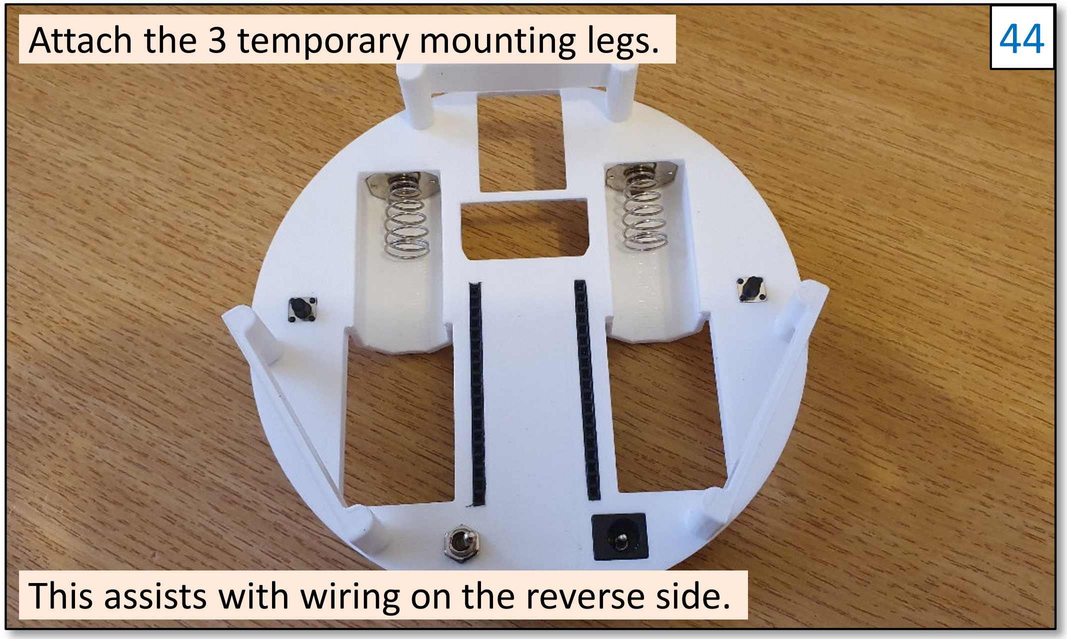


43 Glue the voltage regulator into the micro plate.



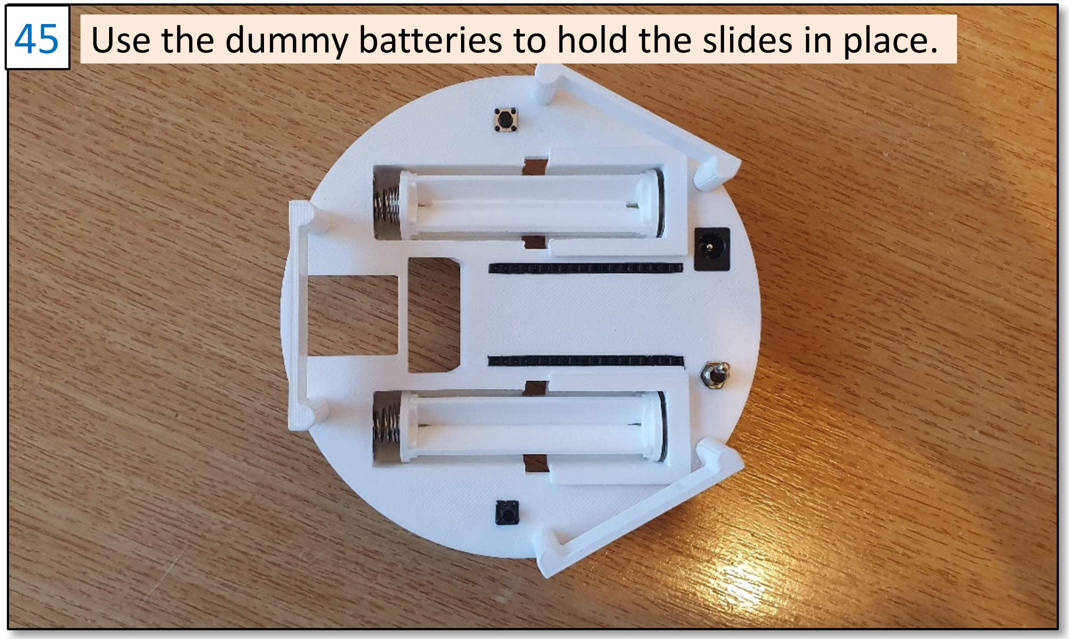
Along with the two resistors.

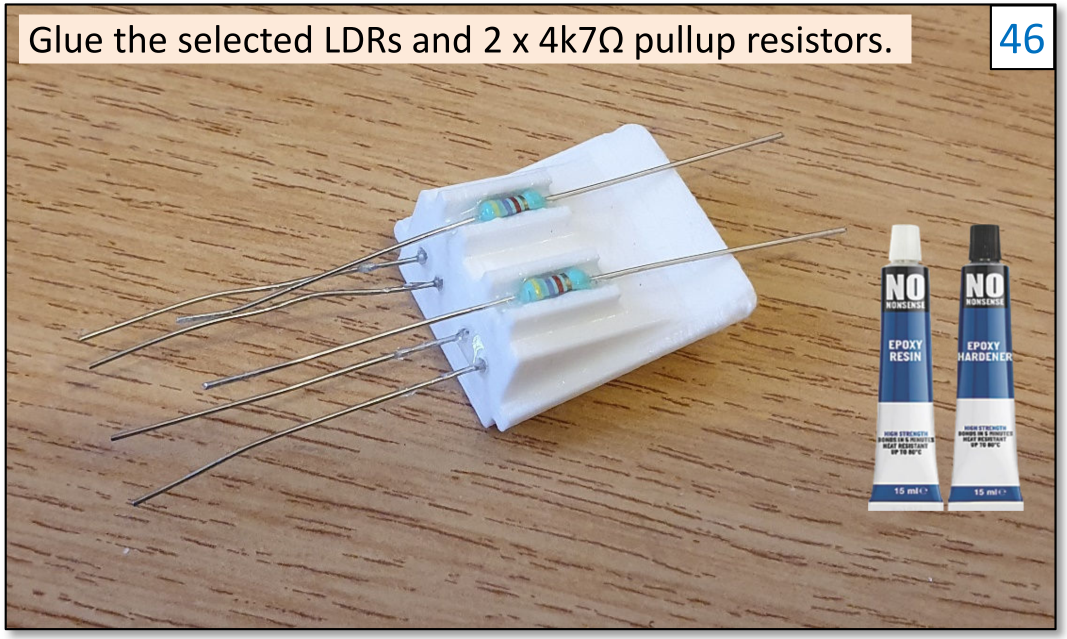
44 Attach the 3 temporary mounting legs.

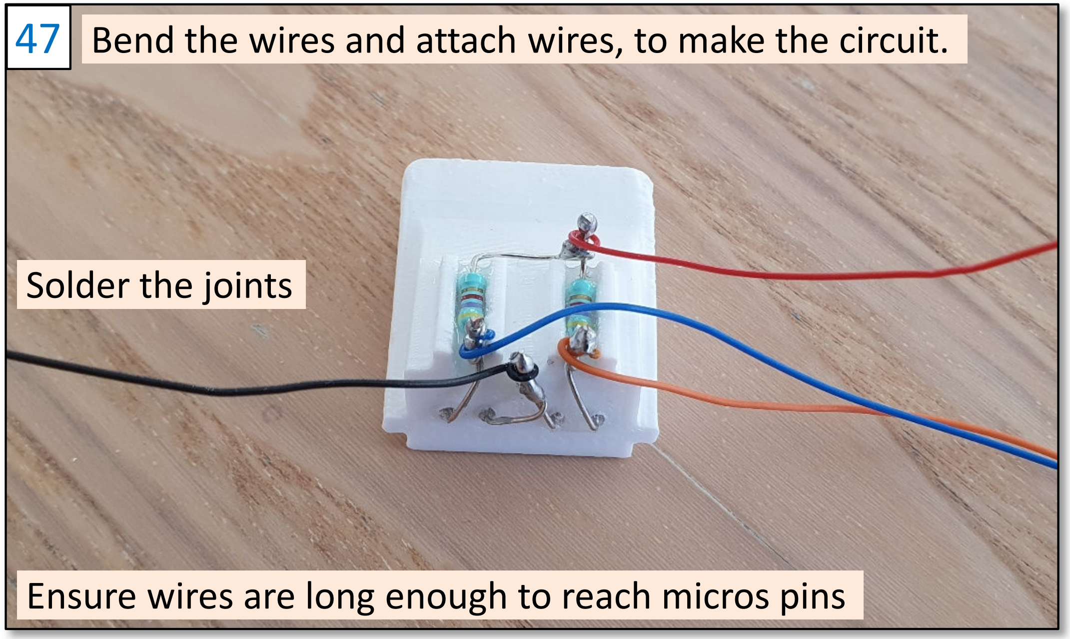


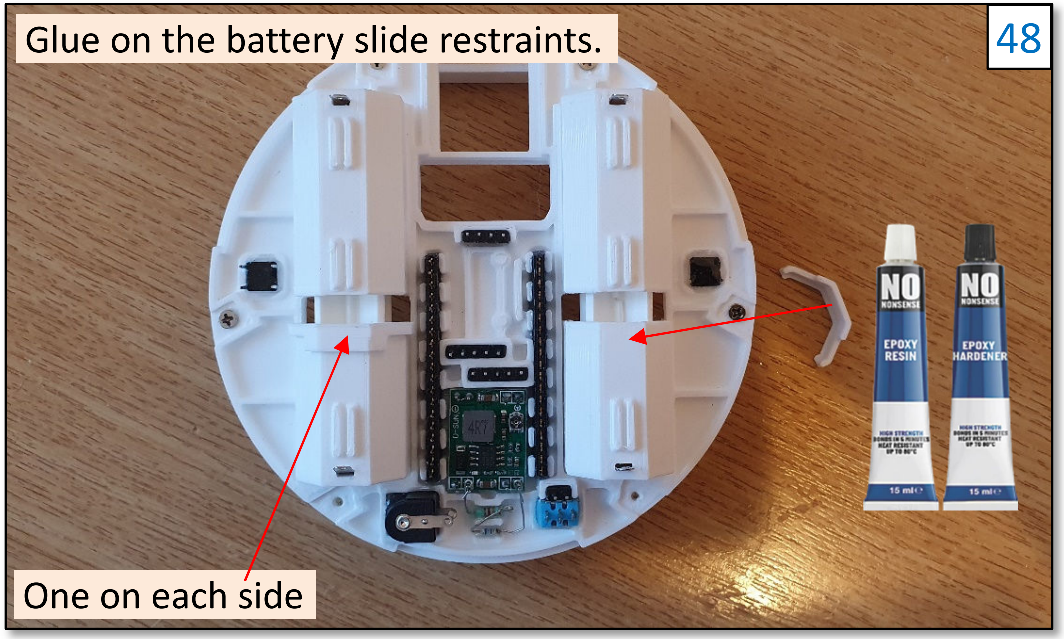
This assists with wiring on the reverse side.

Recorded Build Sequence

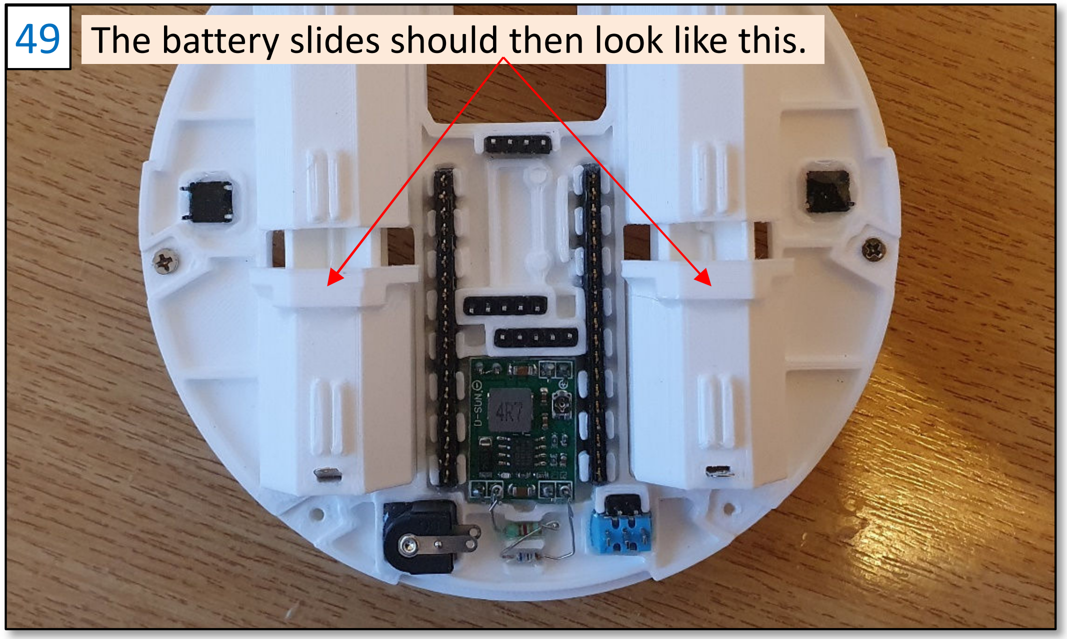




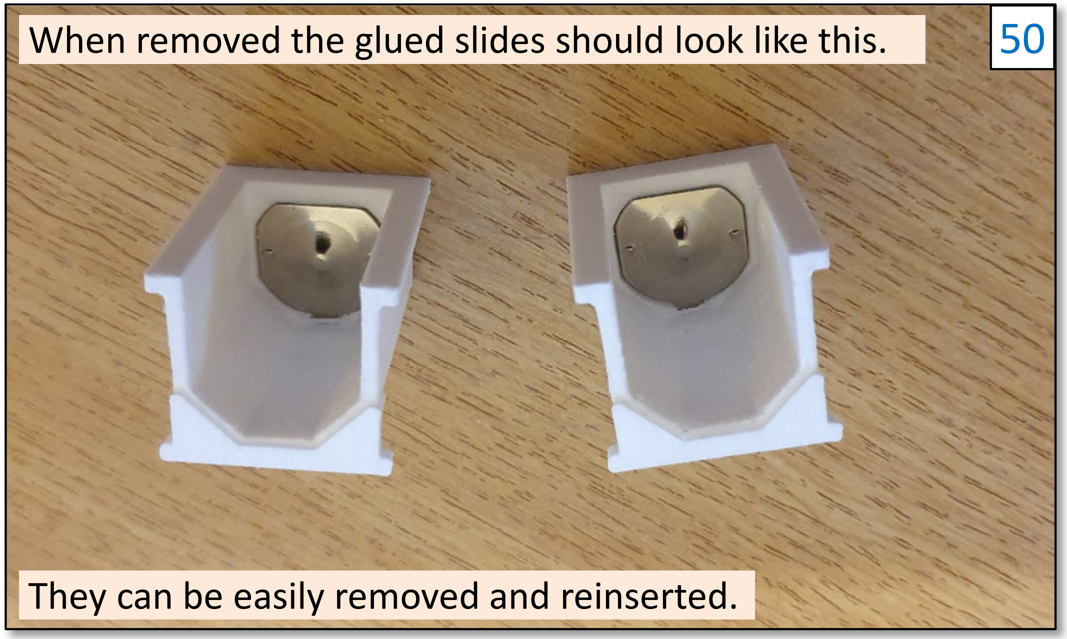




Recorded Build Sequence

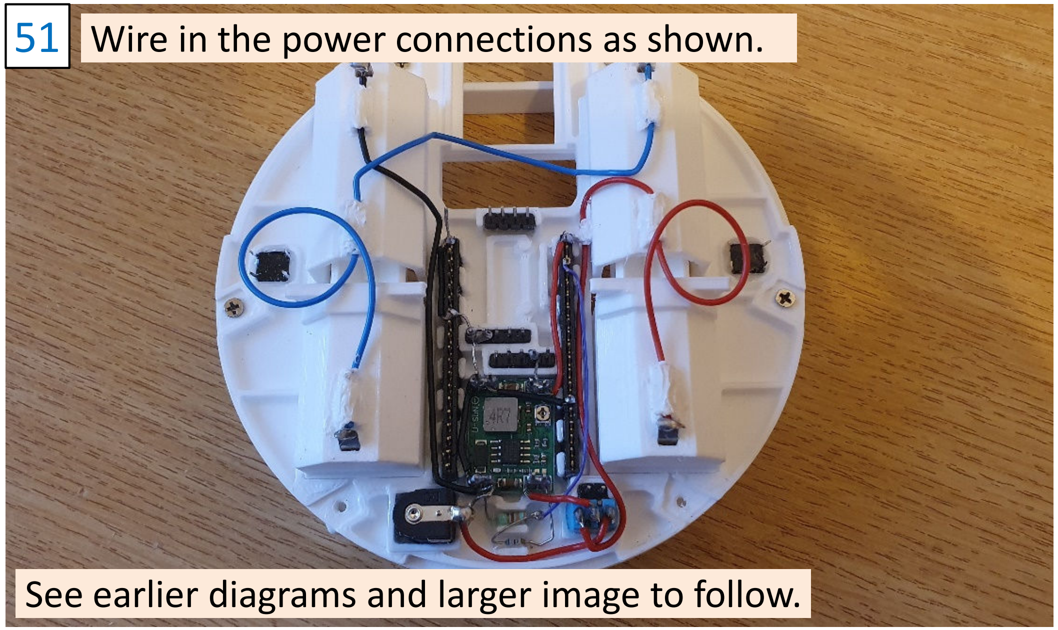


49 The battery slides should then look like this.



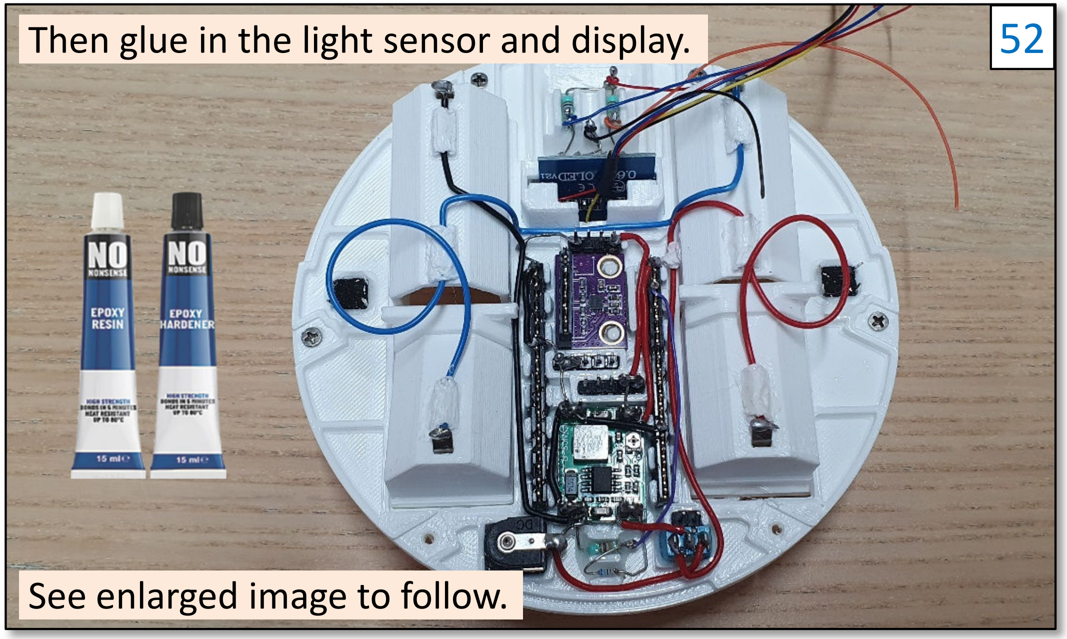
50 When removed the glued slides should look like this.

They can be easily removed and reinserted.



51 Wire in the power connections as shown.

See earlier diagrams and larger image to follow.

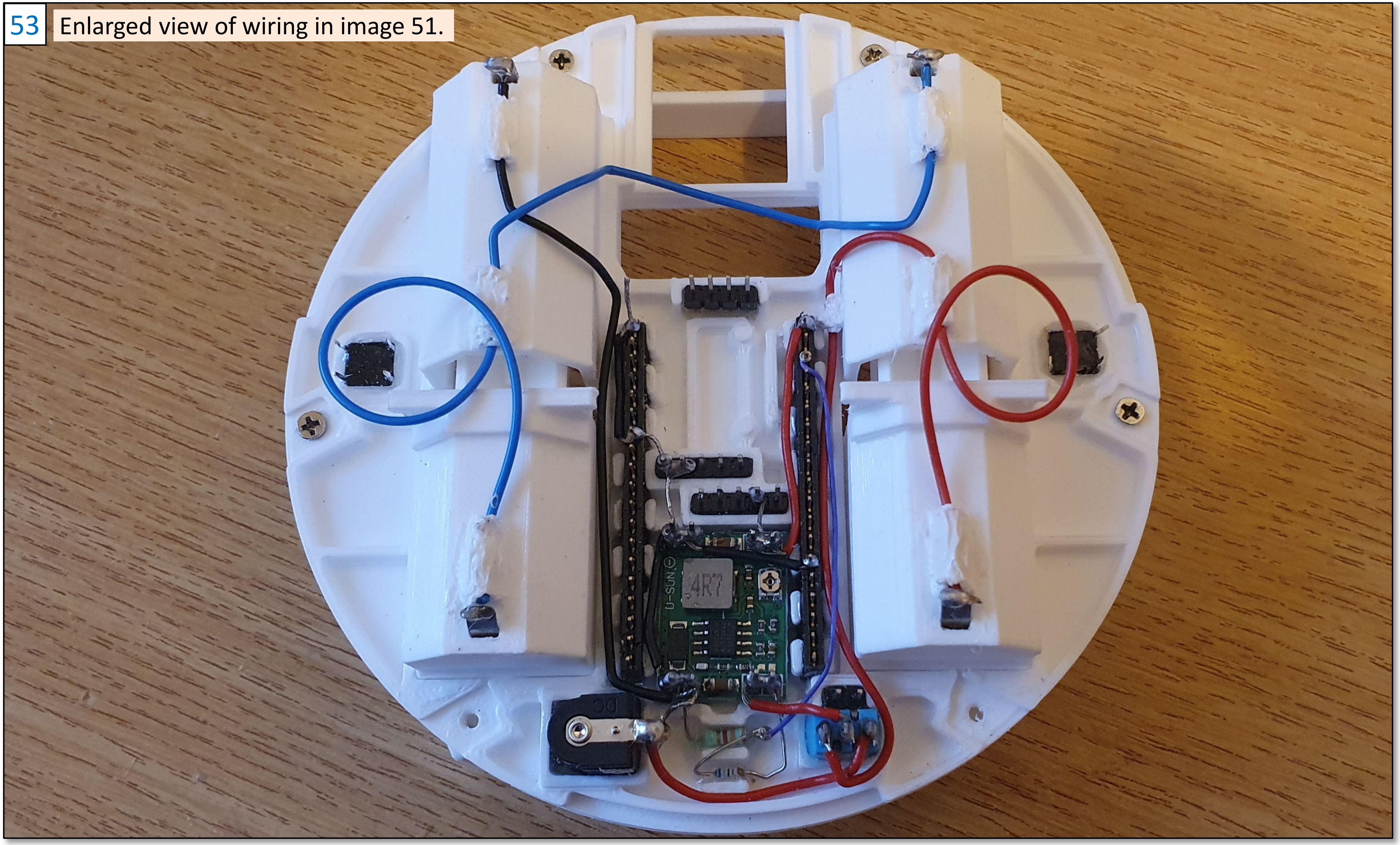


52 Then glue in the light sensor and display.

See enlarged image to follow.

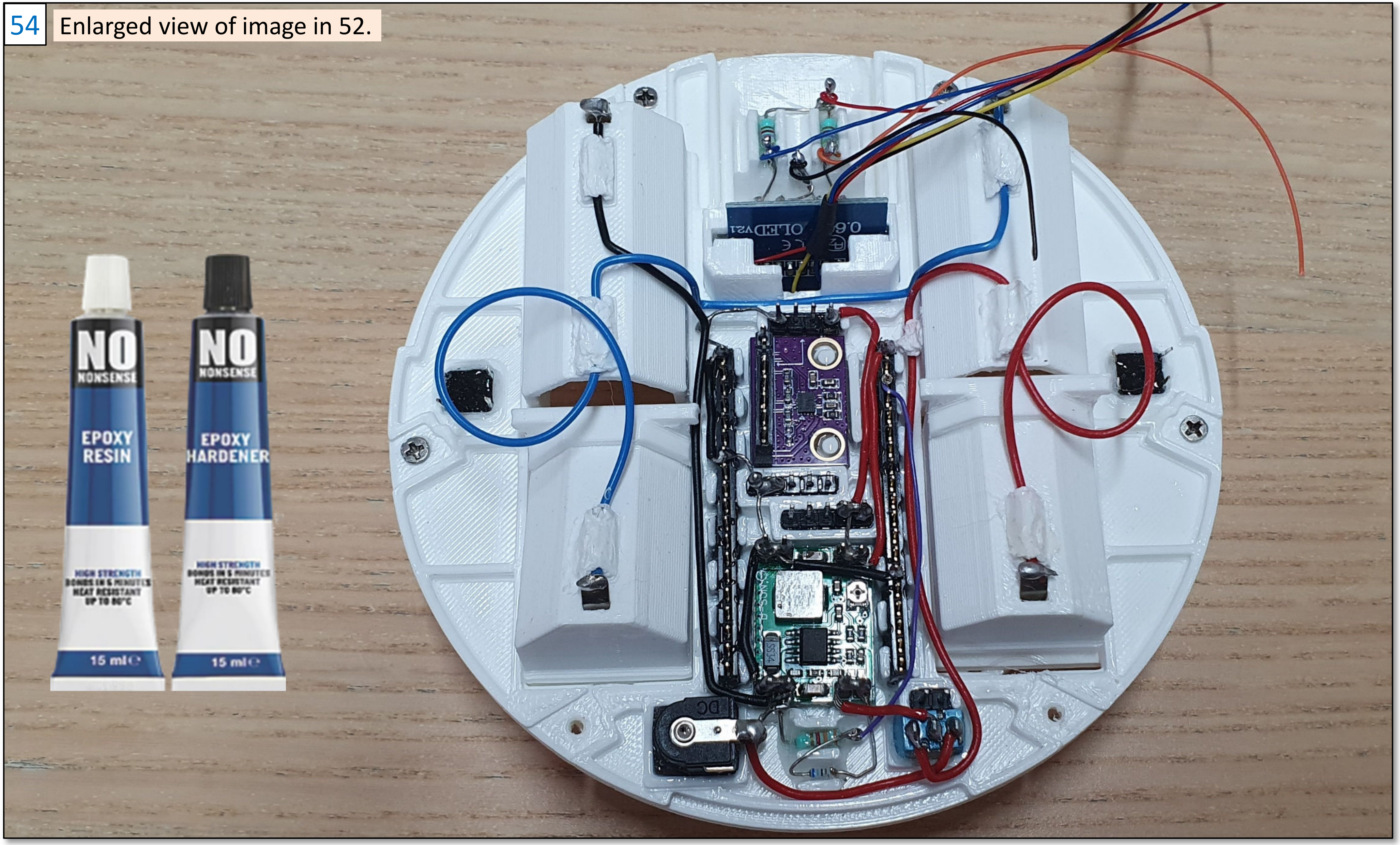
Recorded Build Sequence

53 Enlarged view of wiring in image 51.

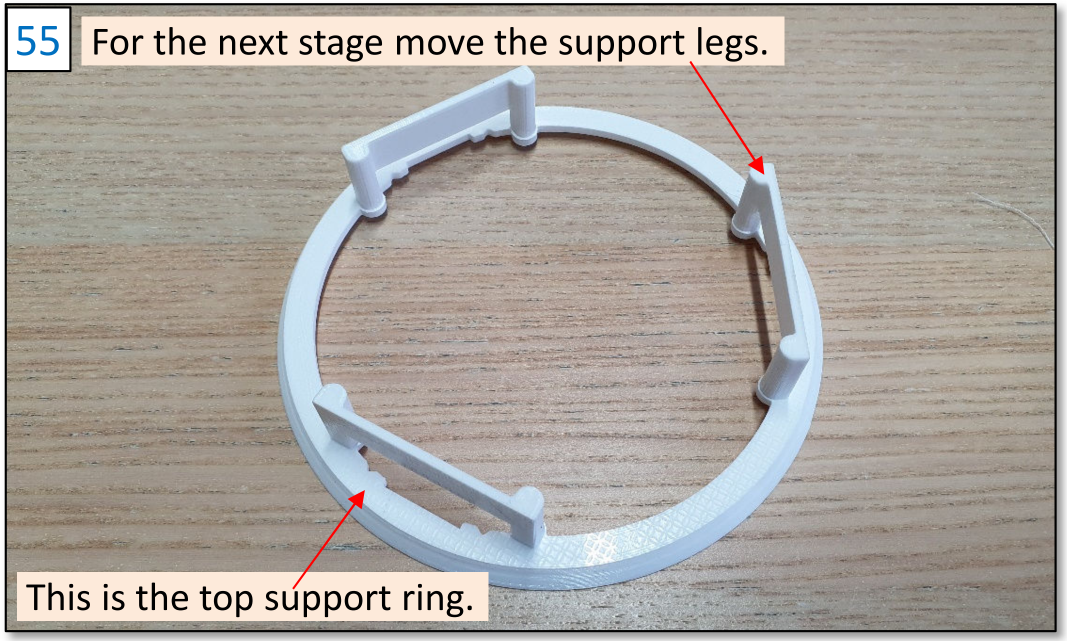


Recorded Build Sequence

54 Enlarged view of image in 52.

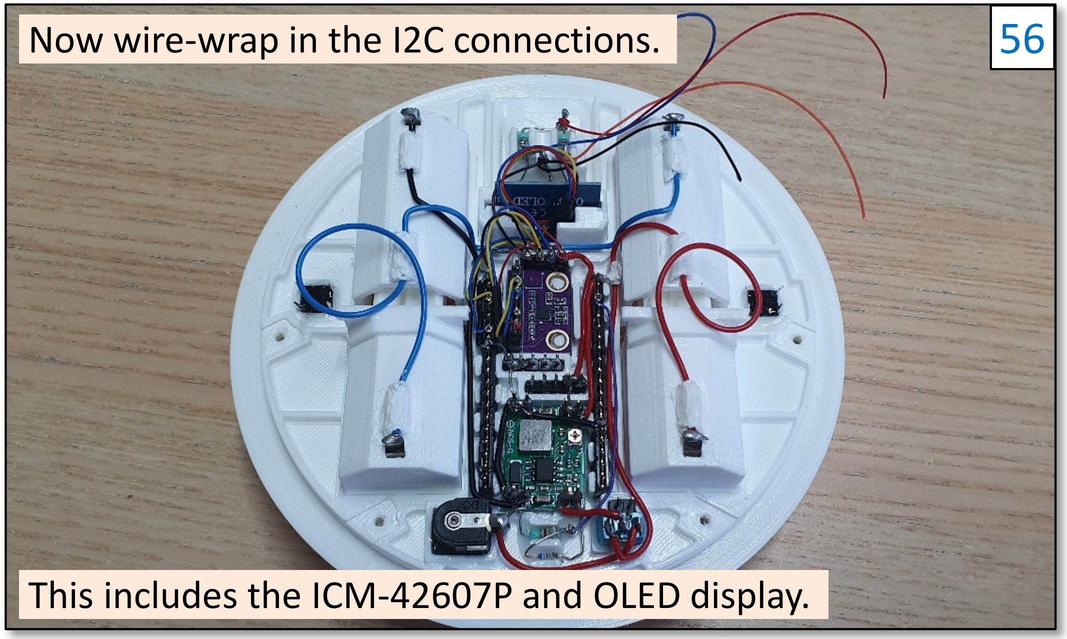


Recorded Build Sequence



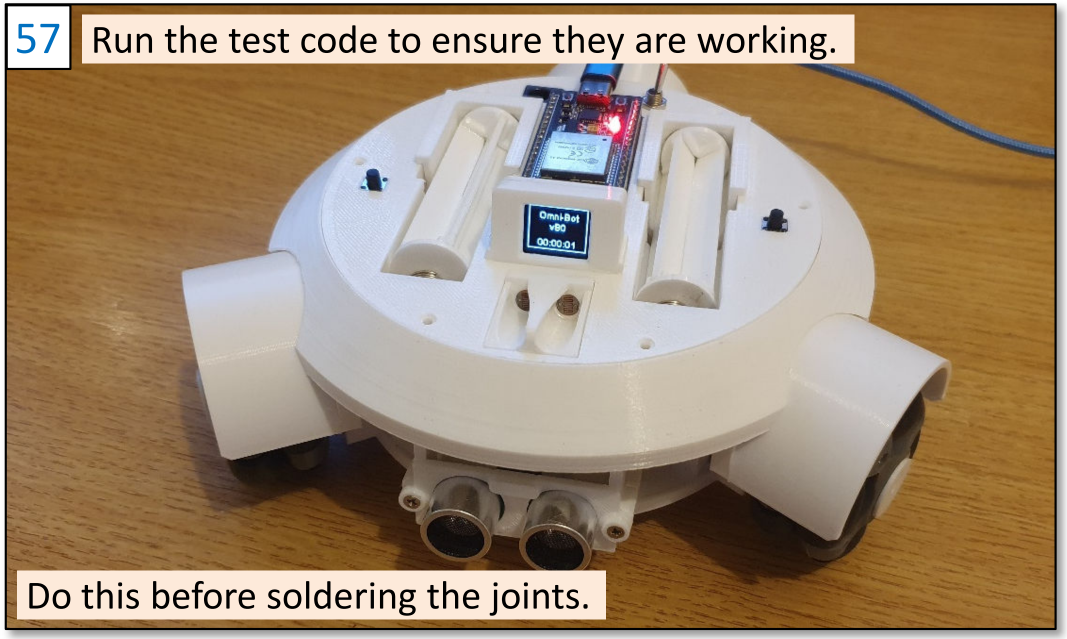
55 For the next stage move the support legs.

This is the top support ring.



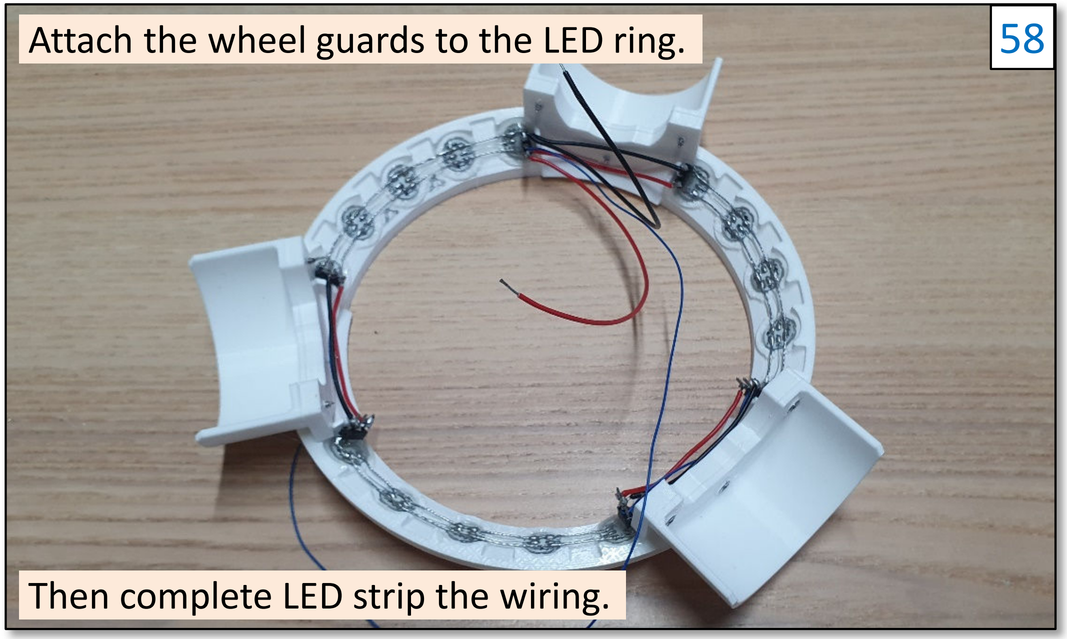
56 Now wire-wrap in the I2C connections.

This includes the ICM-42607P and OLED display.



57 Run the test code to ensure they are working.

Do this before soldering the joints.

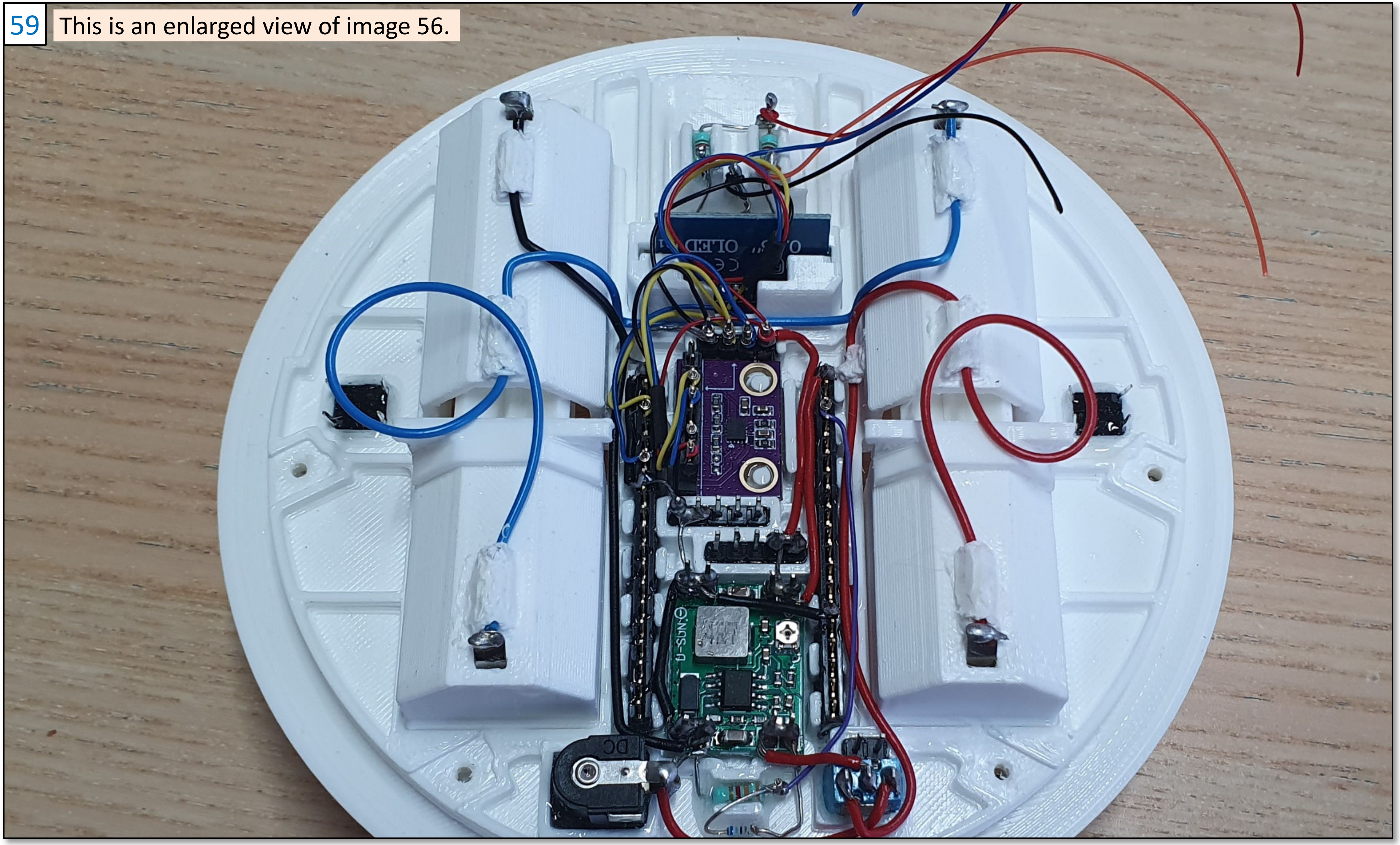


58 Attach the wheel guards to the LED ring.

Then complete LED strip the wiring.

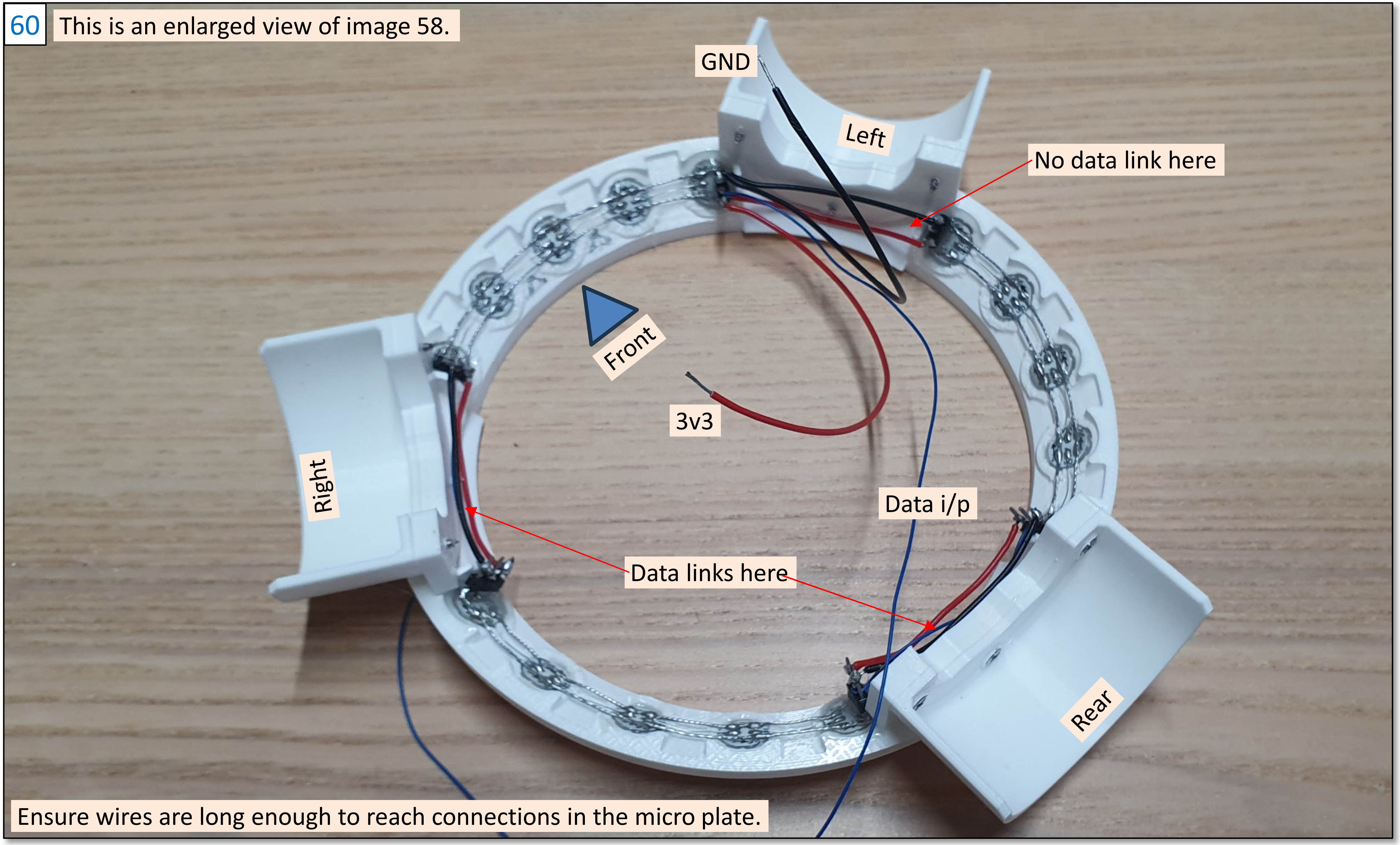
Recorded Build Sequence

59 This is an enlarged view of image 56.



Recorded Build Sequence

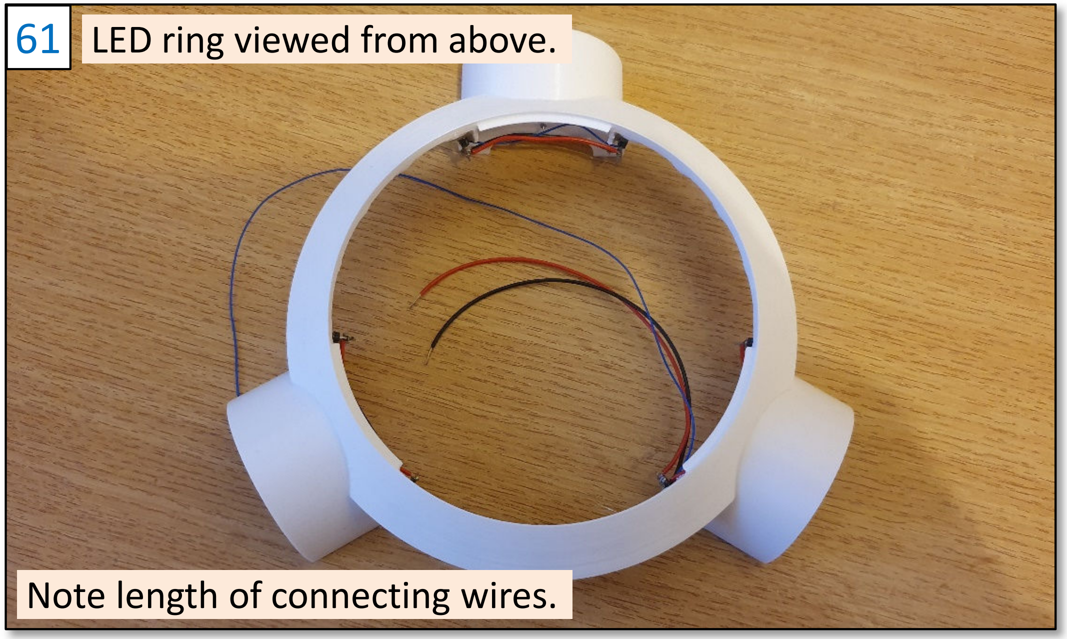
60 This is an enlarged view of image 58.



Ensure wires are long enough to reach connections in the micro plate.

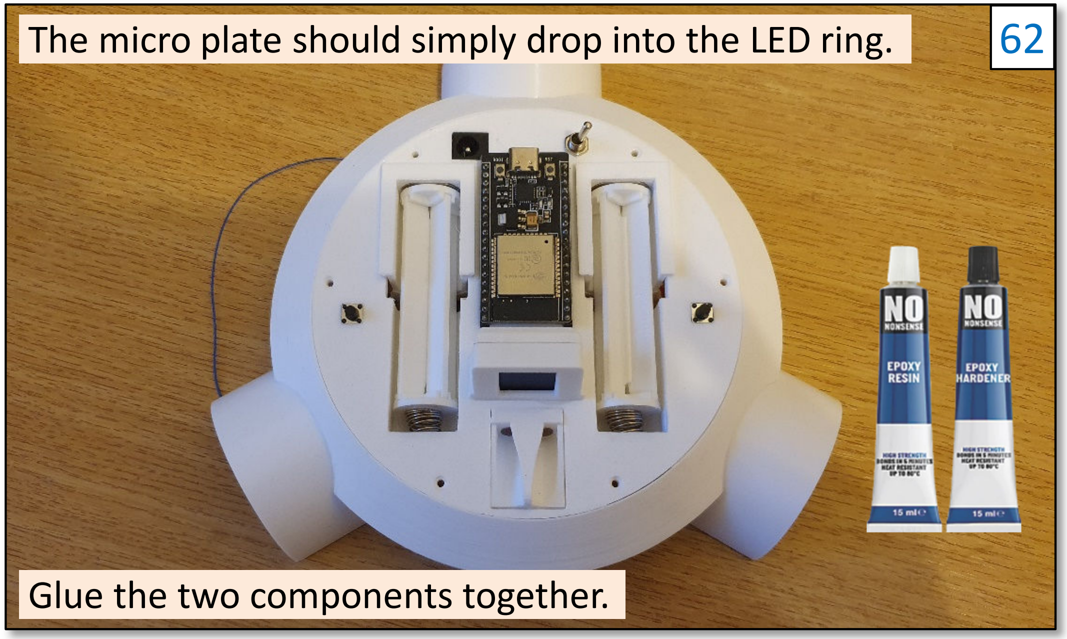
Recorded Build Sequence

61 LED ring viewed from above.



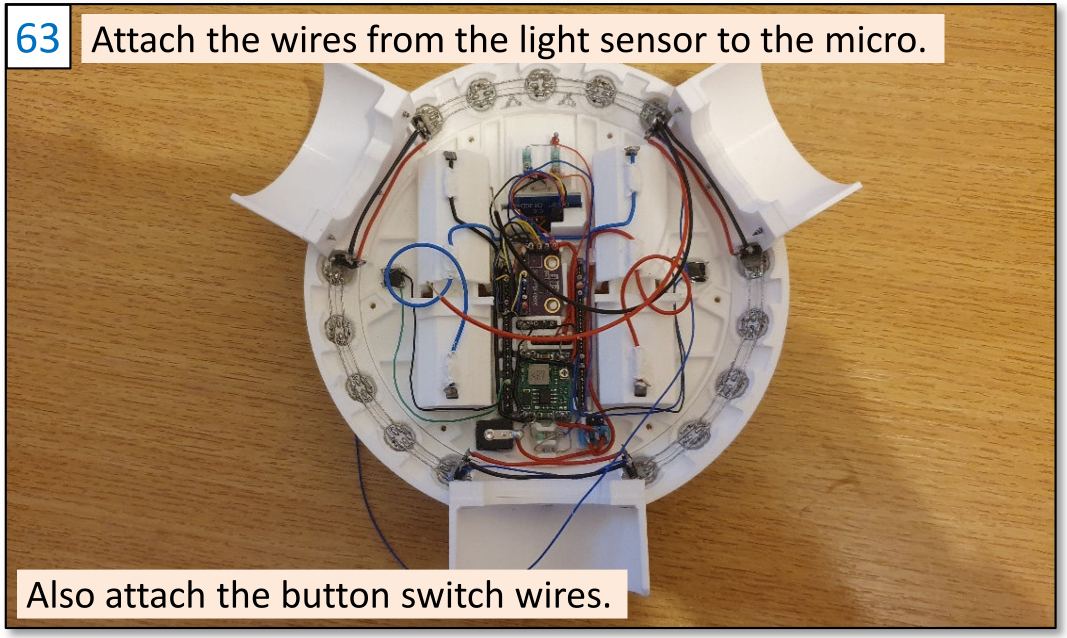
Note length of connecting wires.

62 The micro plate should simply drop into the LED ring.



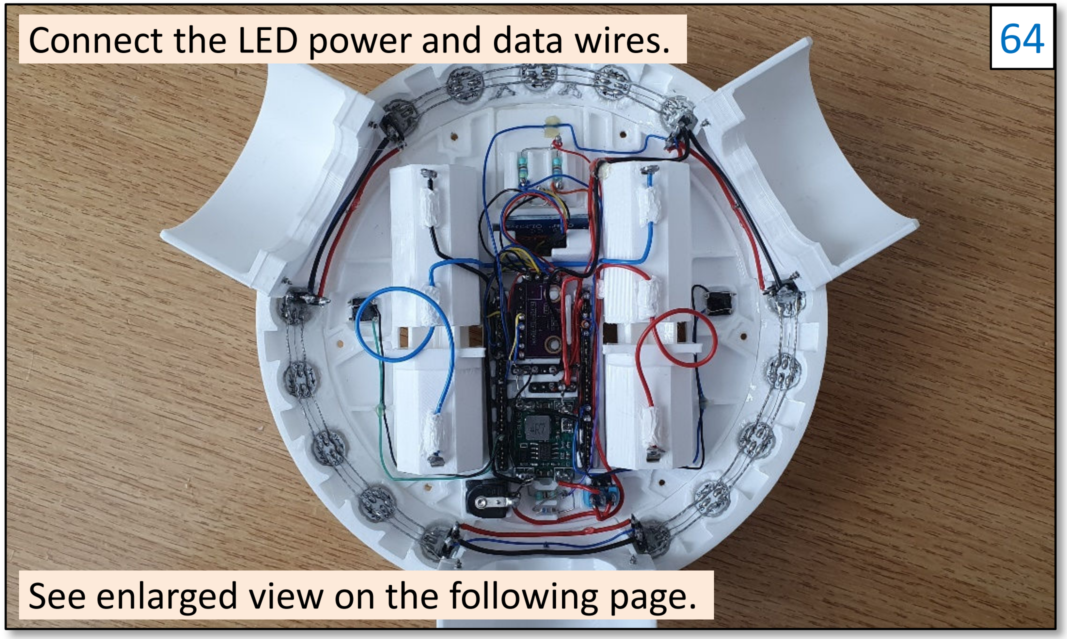
Glue the two components together.

63 Attach the wires from the light sensor to the micro.



Also attach the button switch wires.

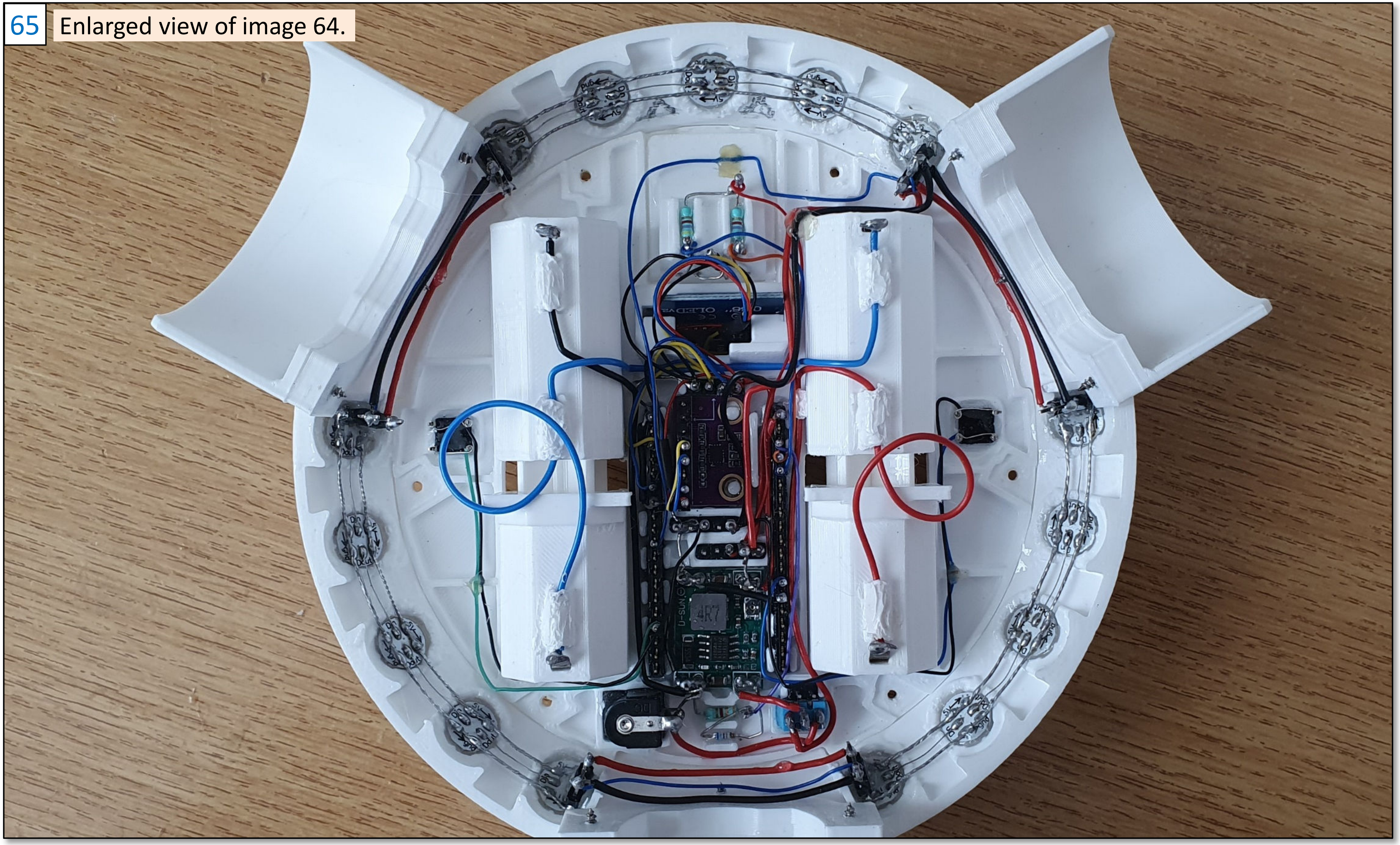
64 Connect the LED power and data wires.



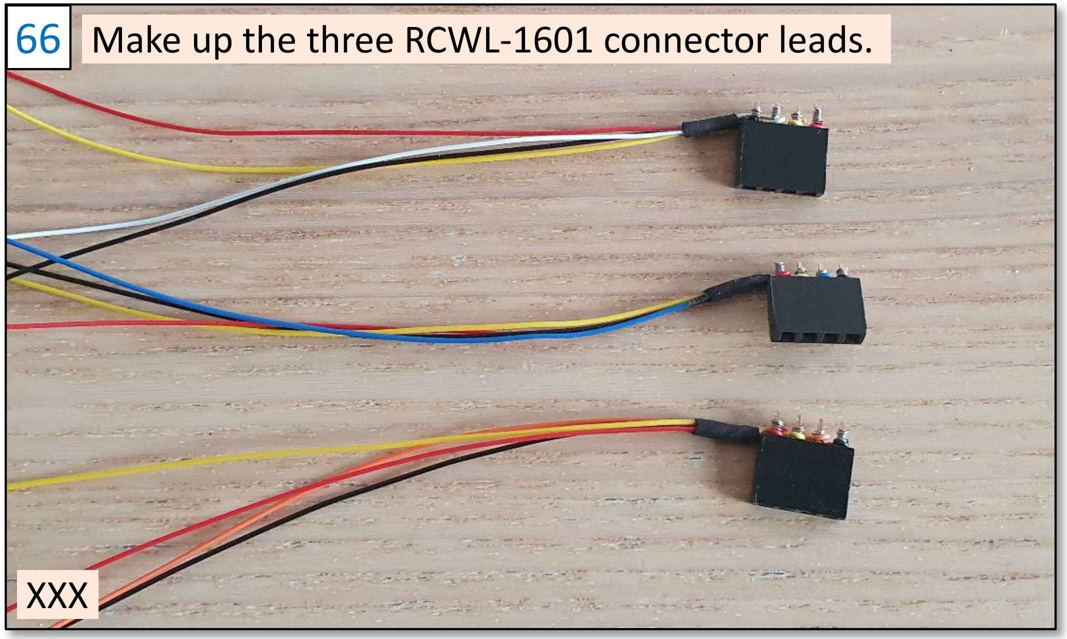
See enlarged view on the following page.

Recorded Build Sequence

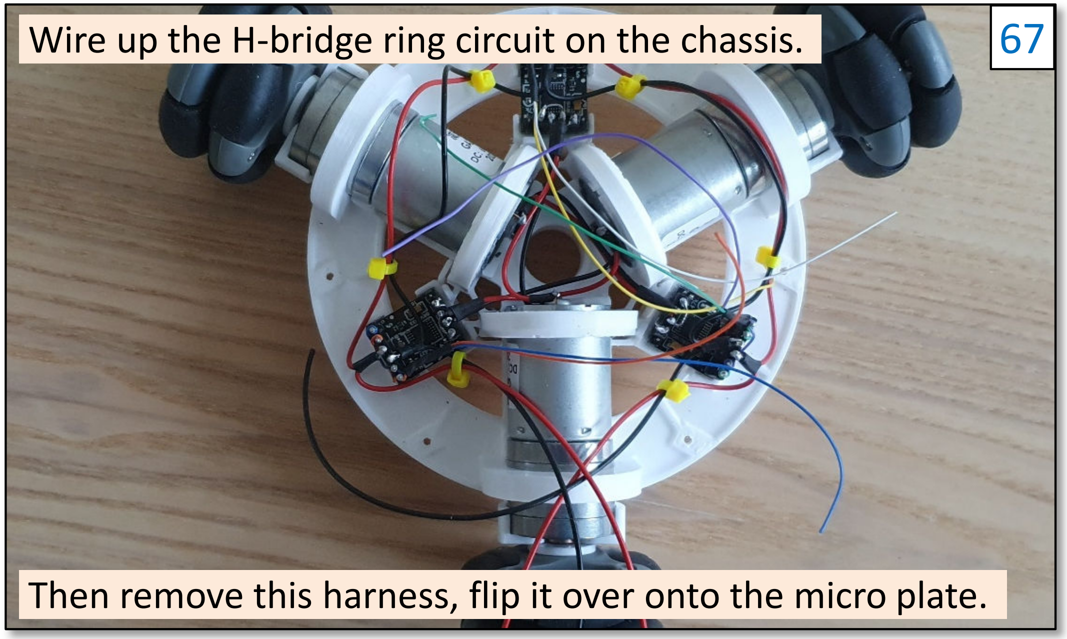
65 Enlarged view of image 64.



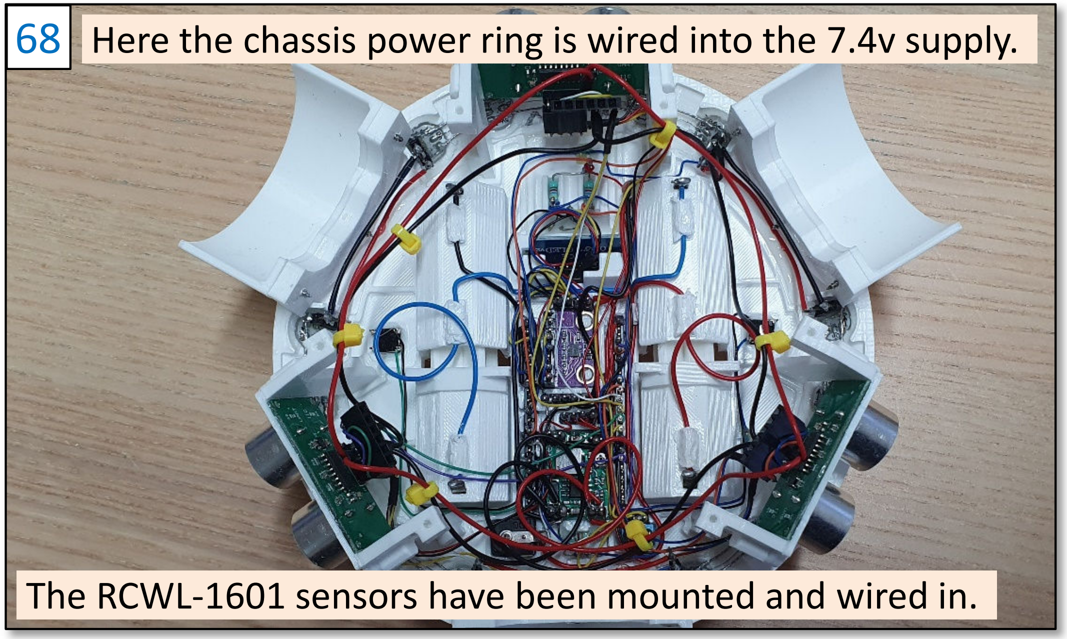
Recorded Build Sequence


66 Make up the three RCWL-1601 connector leads.

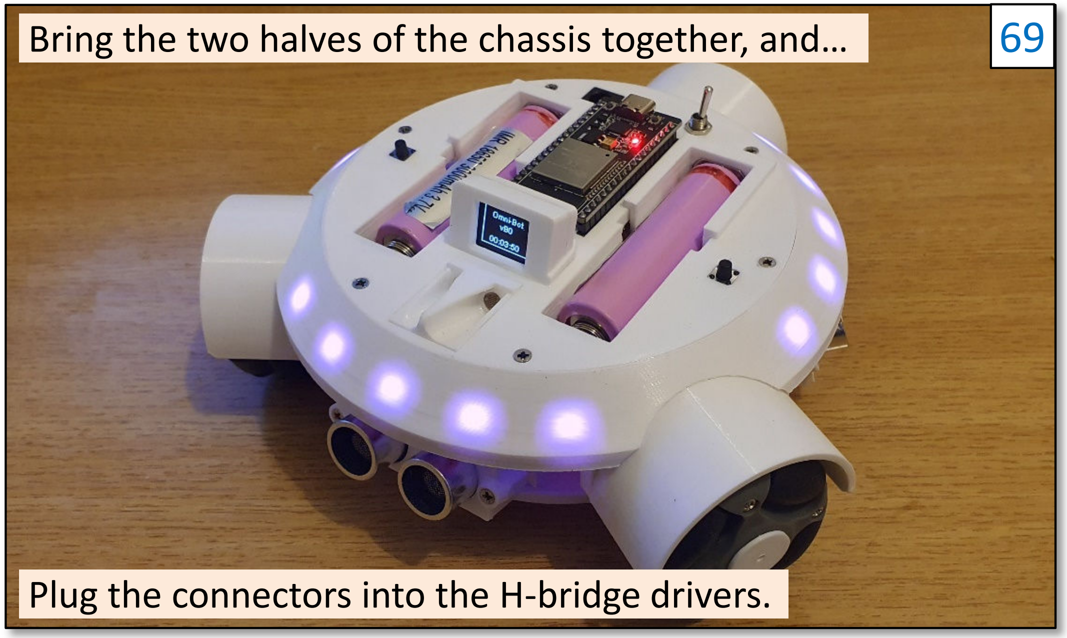
XXX


67 Wire up the H-bridge ring circuit on the chassis.

Then remove this harness, flip it over onto the micro plate.

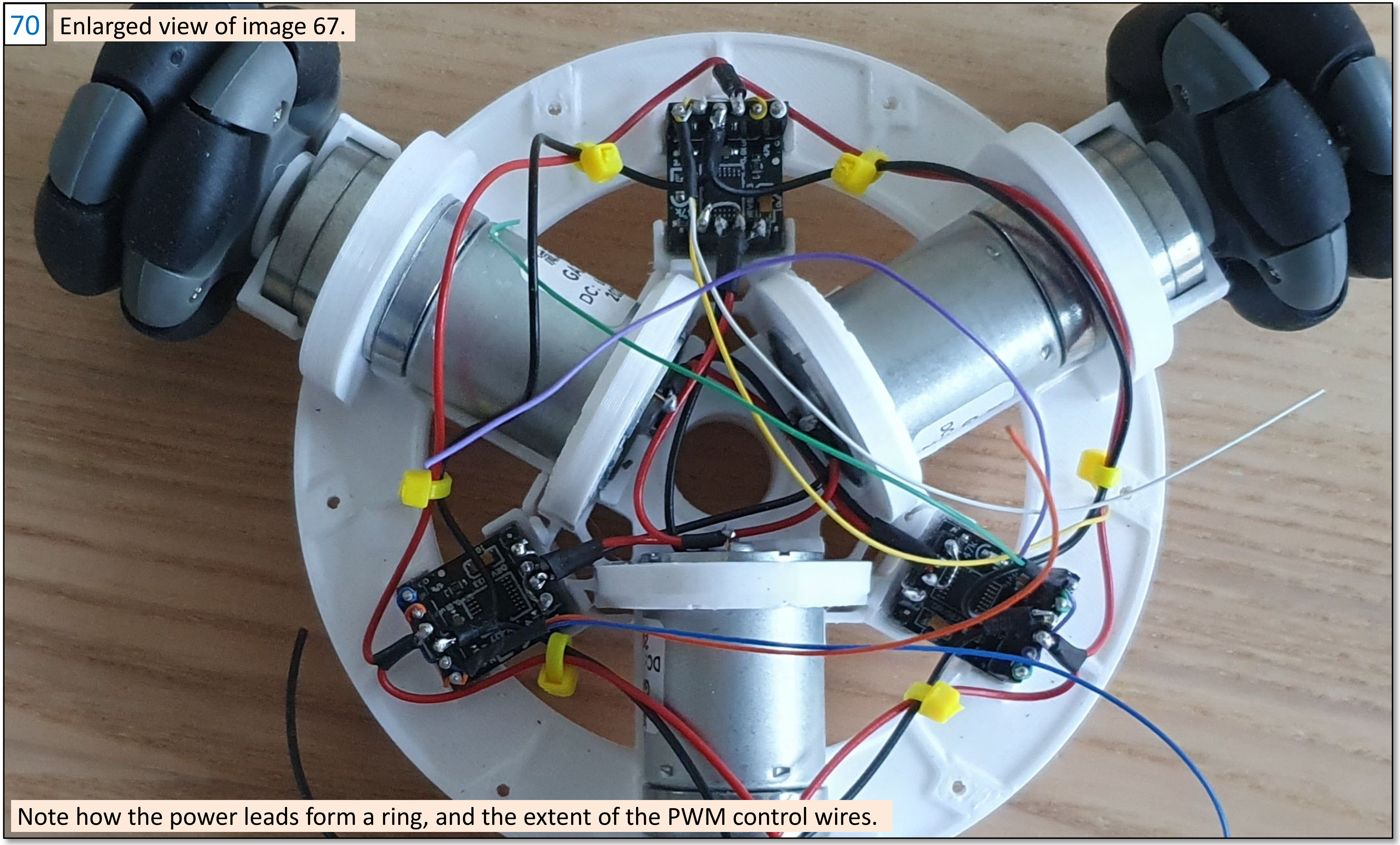

68 Here the chassis power ring is wired into the 7.4v supply.

The RCWL-1601 sensors have been mounted and wired in.


69 Bring the two halves of the chassis together, and…

Plug the connectors into the H-bridge drivers.

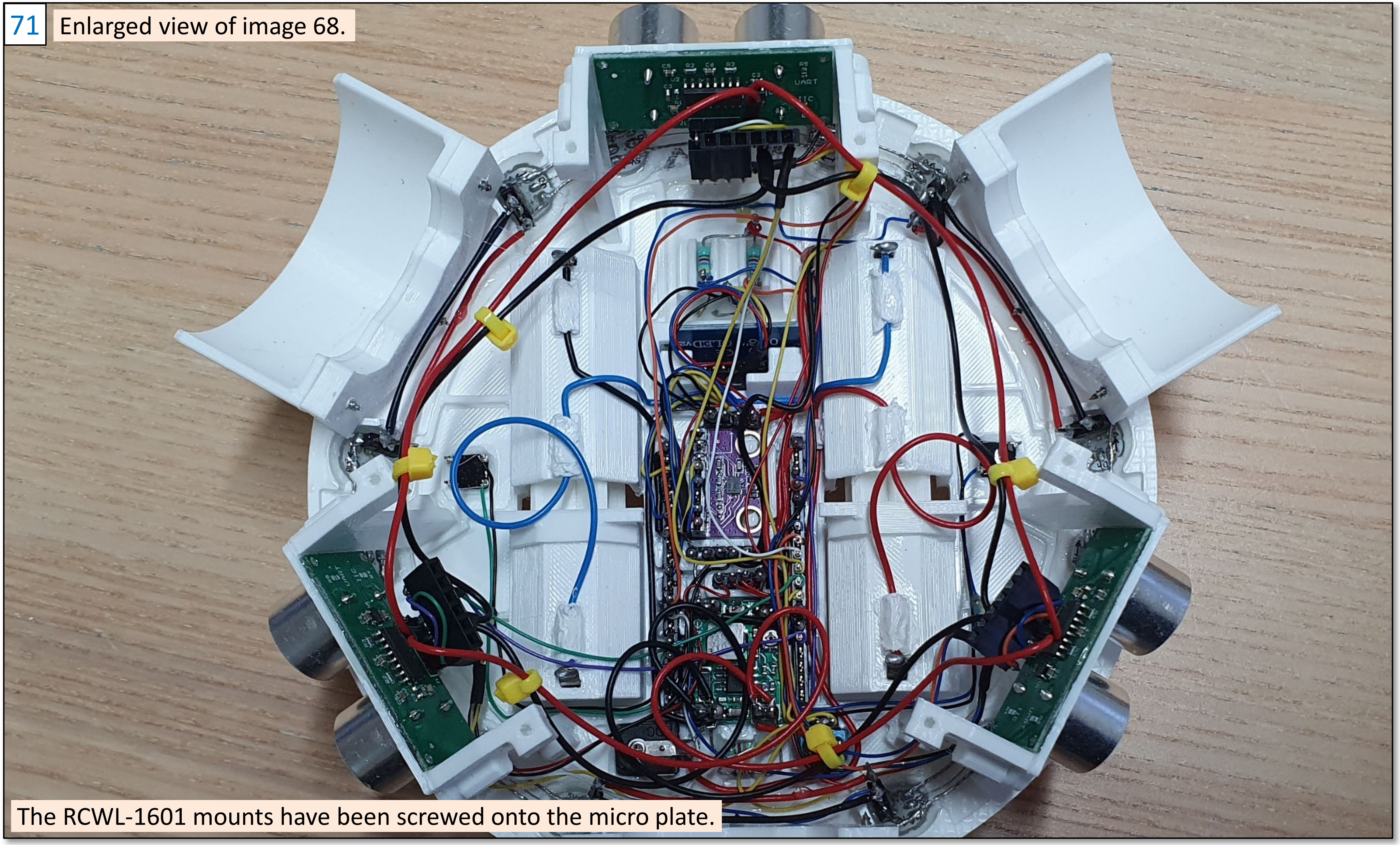
70 Enlarged view of image 67.



Note how the power leads form a ring, and the extent of the PWM control wires.

Recorded Build Sequence

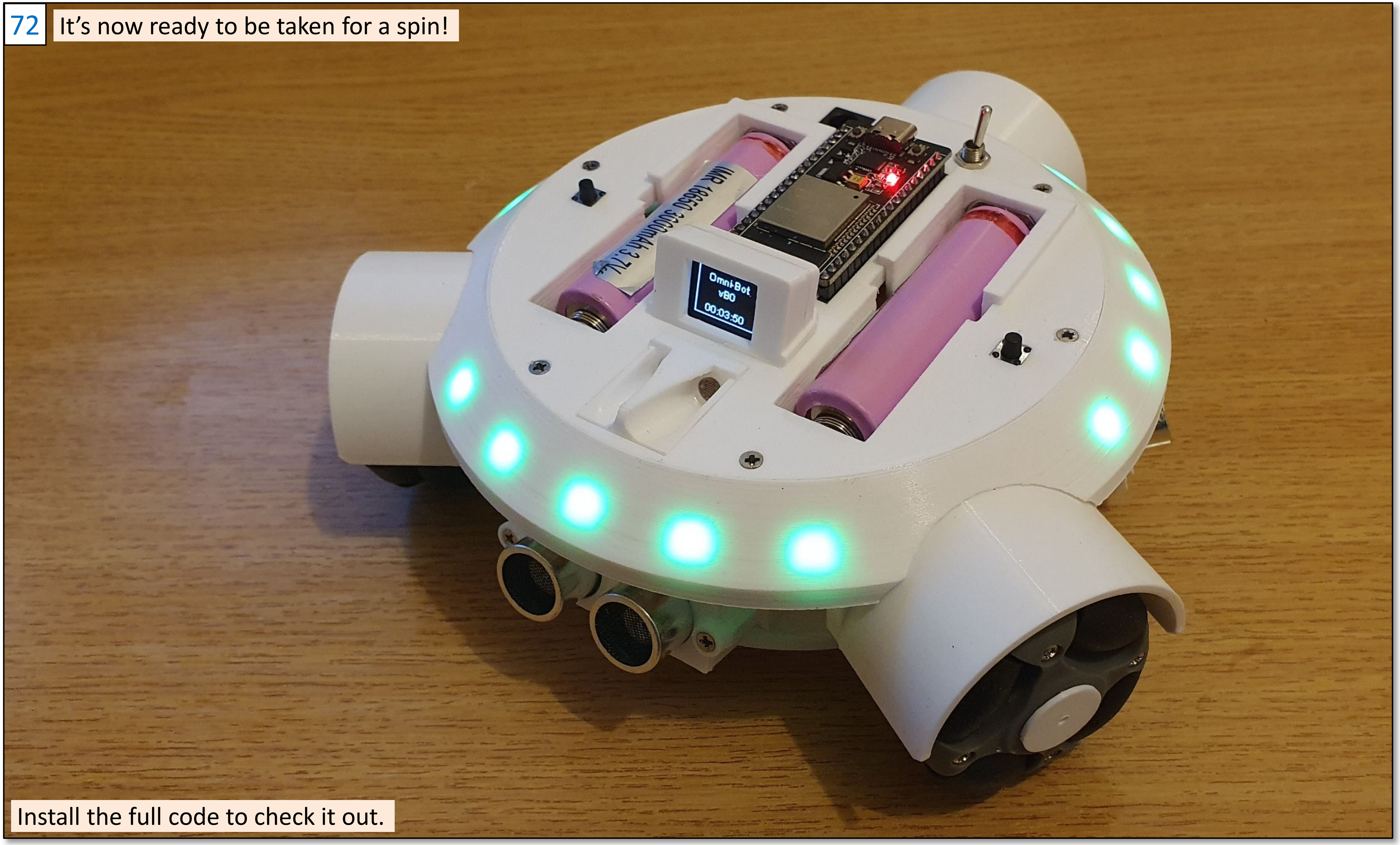
71 Enlarged view of image 68.



The RCWL-1601 mounts have been screwed onto the micro plate.

Recorded Build Sequence

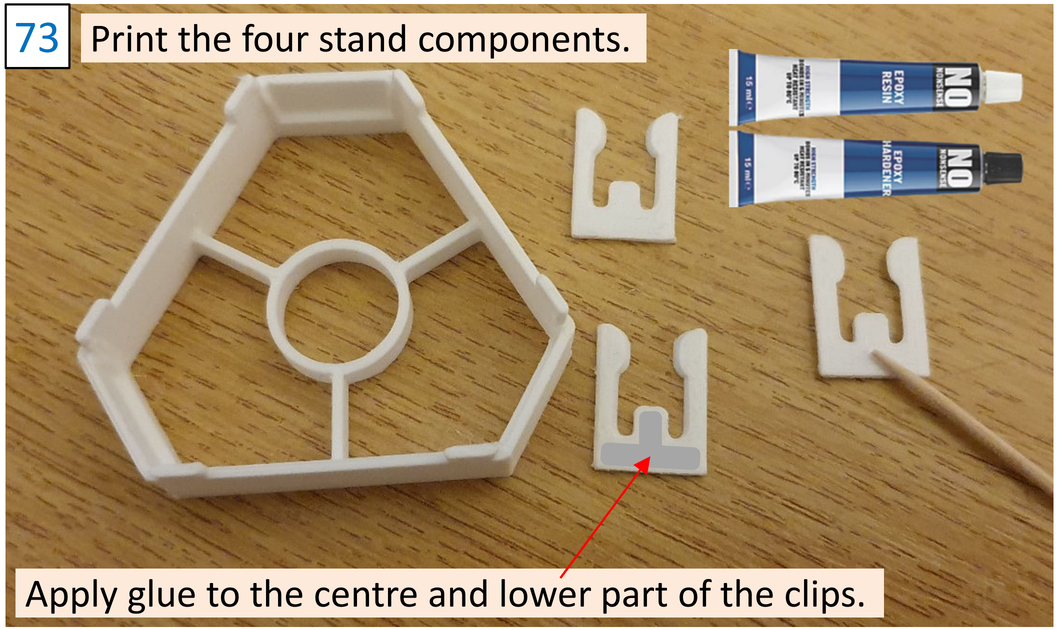
72 It’s now ready to be taken for a spin!



Install the full code to check it out.

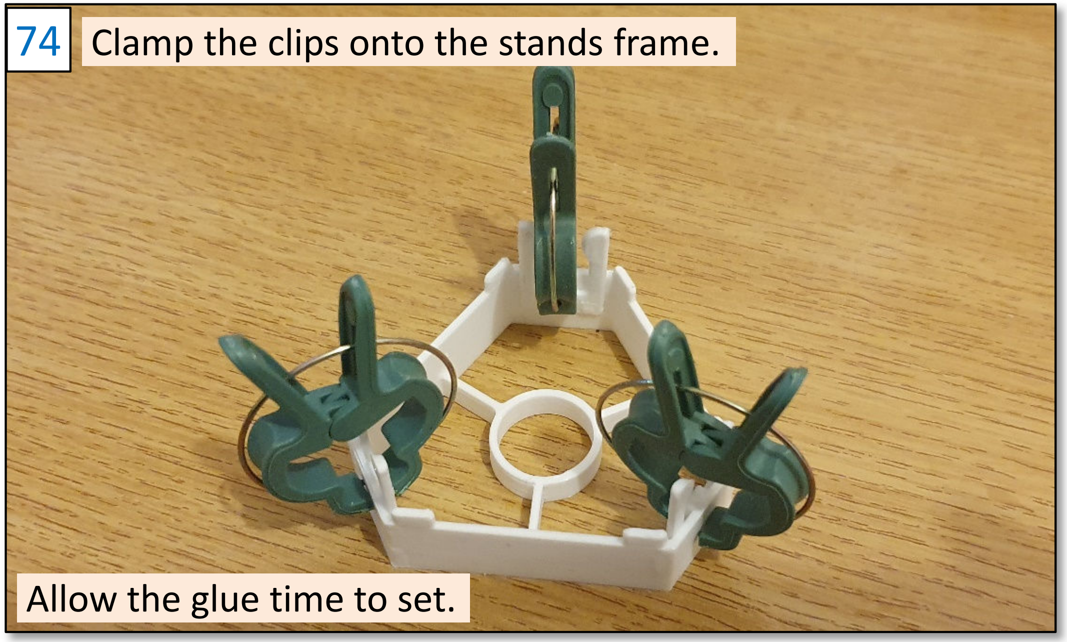
Recorded Build Sequence

73 Print the four stand components.



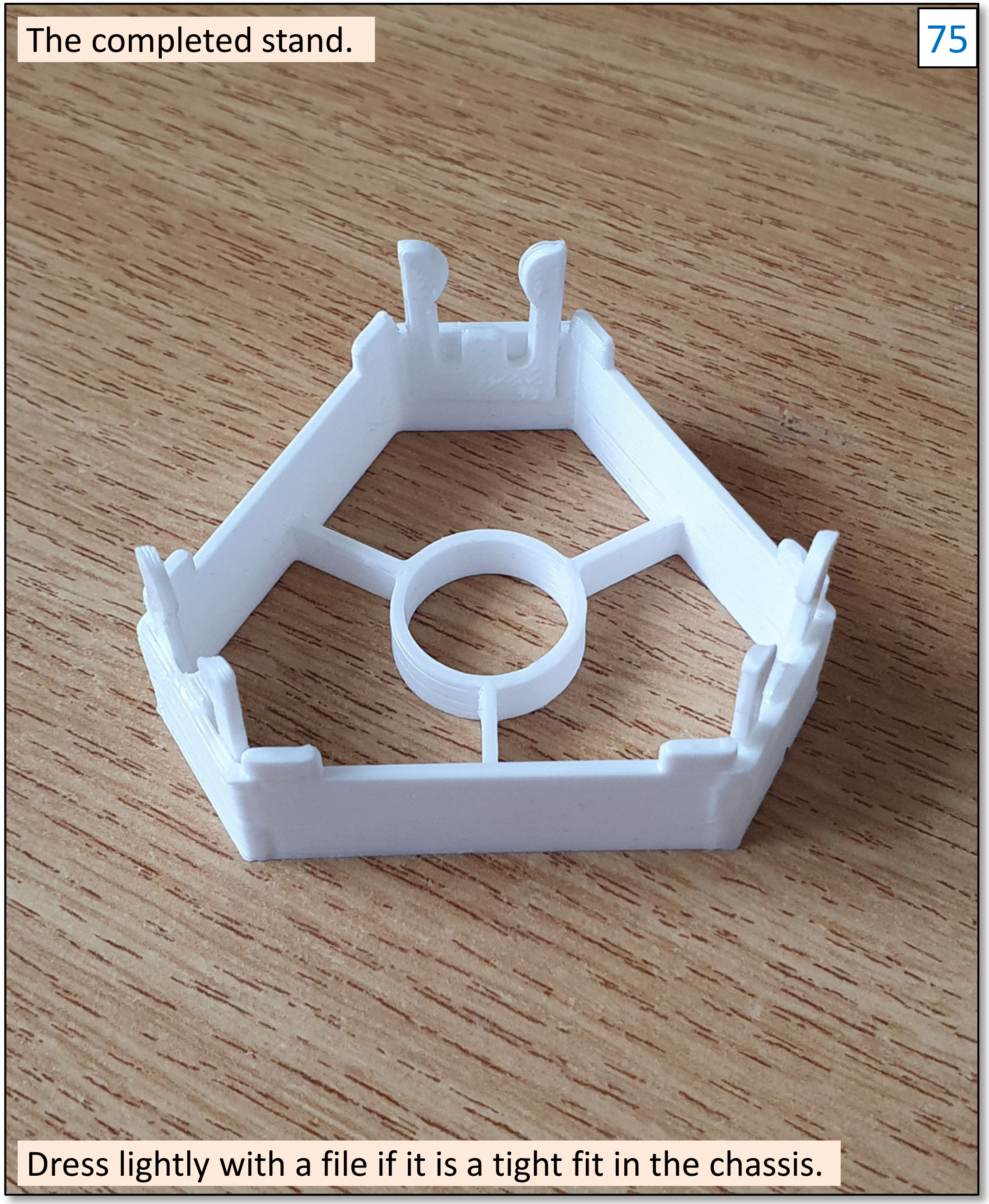
Apply glue to the centre and lower part of the clips.

74 Clamp the clips onto the stands frame.



Allow the glue time to set.

75 The completed stand.



Dress lightly with a file if it is a tight fit in the chassis.

Dual H-bridge Wiring

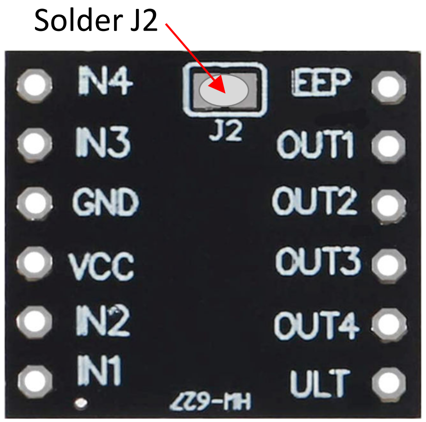
In this design I used one DRV8833 H-bridge dual channel controller for each motor, and combined each channels 2A max current capability, and halved their MOS-FETs on-channel resistance. Giving a more efficient drive signal for each motor.

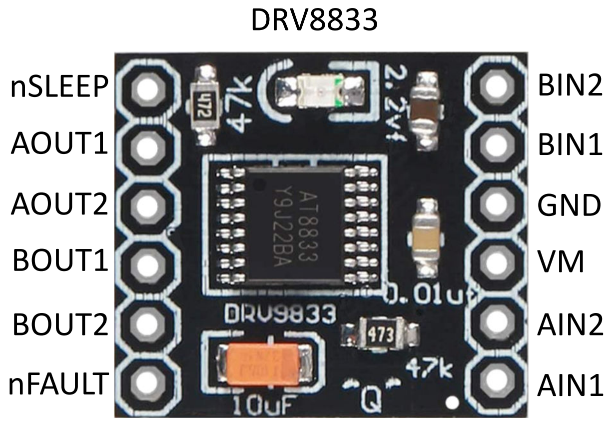
This is achieved by wiring the controllers as shown here. Both the input pins and output pins are connected together as shown. Make sure that you wire the controllers correctly, or you are likely to damage the controller, and draw excessive current from the batteries.

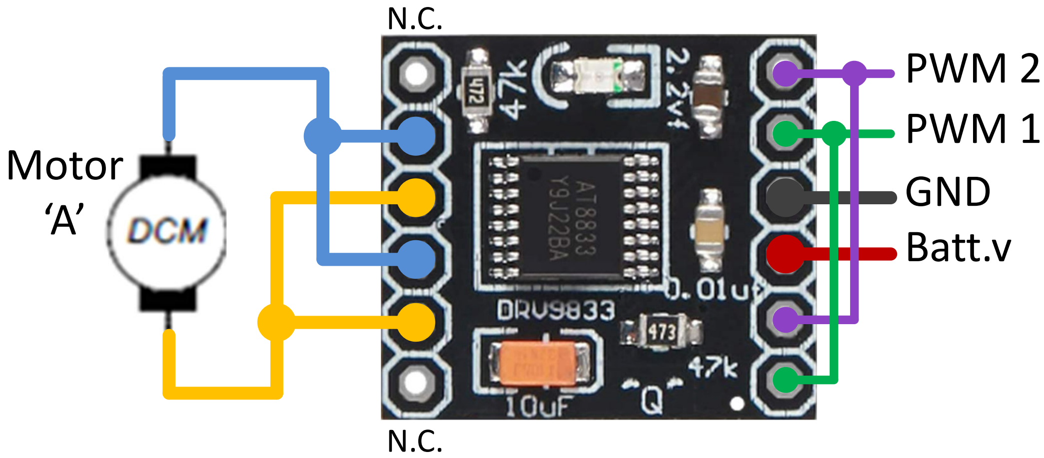
The nSLEEP and nFAULT pins are not connected; however, the nSLEEP pin is pulled HIGH by an on-board 47kΩ resistor when you bridge J2 with a blob of solder, as shown. Do this **first**, before soldering in the pin strips to the boards.

You will find more information on the DRV8833 driver on a later slide. This should enable you to understand how the code producing the PWM signals in the ESP32, controls the motion of the DC motor. For each direction of travel, one PWM input pin is held HIGH, whilst PWM signals are applied to the other.

As the supply voltage affects motor speed and torque, the code monitors the battery voltage and adjusts the applied PWM to give a maximum average voltage of 7.6v.

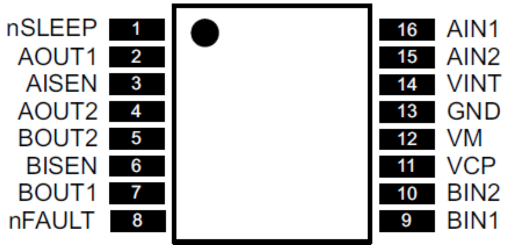


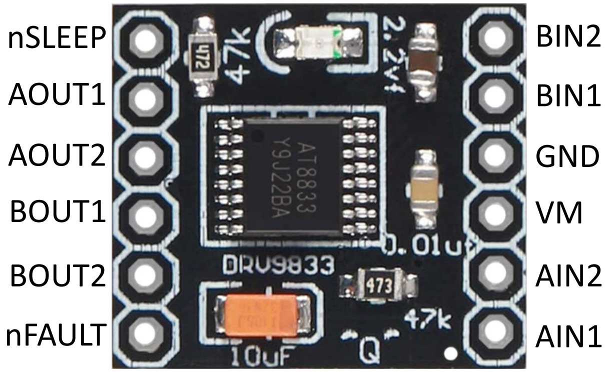




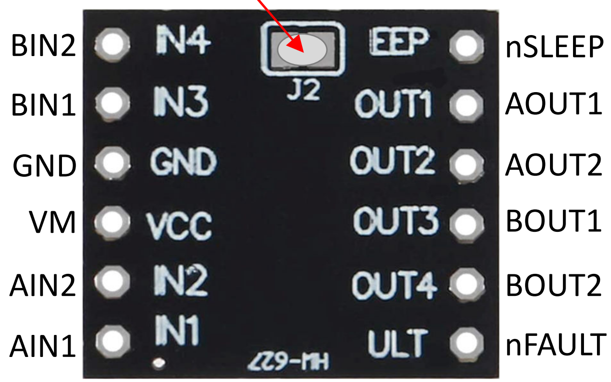
CAUTION – This design does not include any form of current limiting protection (like a fuse). Incorrect wiring can result in large currents, which can cause damage and even the risk of a **fire**. Check and double check your wiring before inserting the batteries.

DRV8833 H-bridge Driver

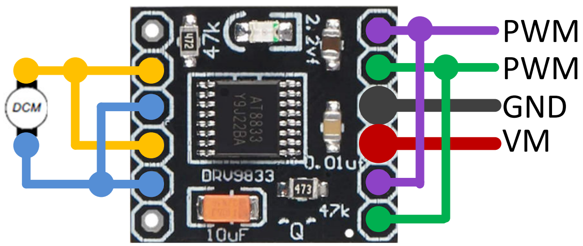




Solder J2



Combined H-bridge.



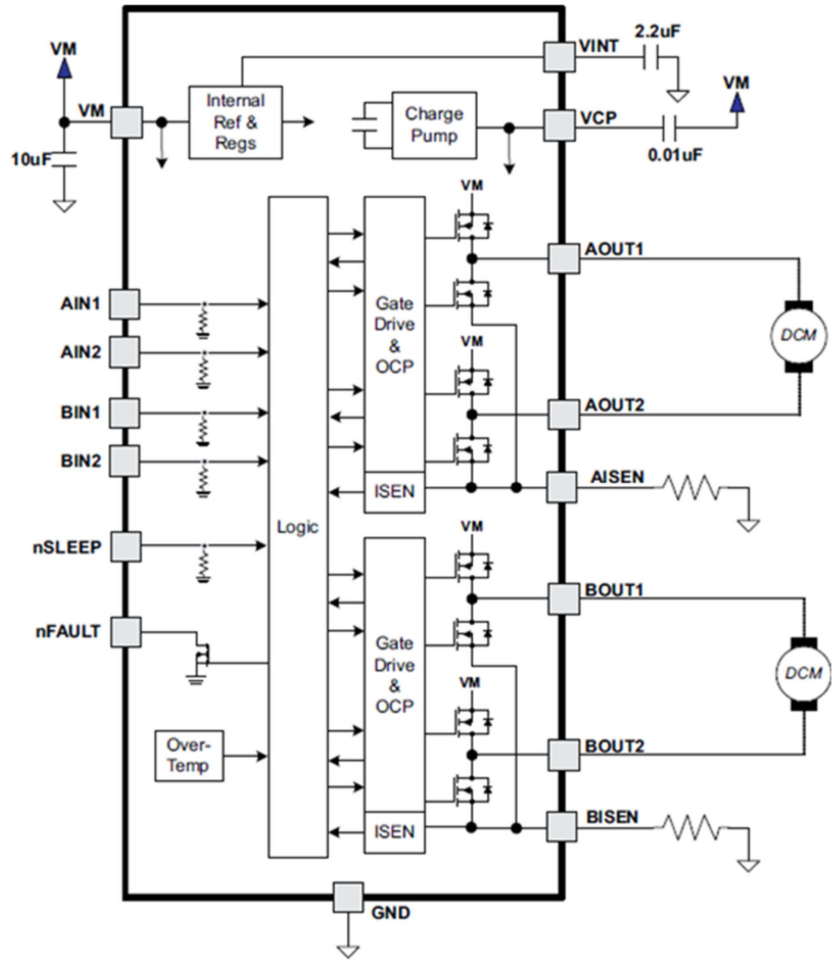
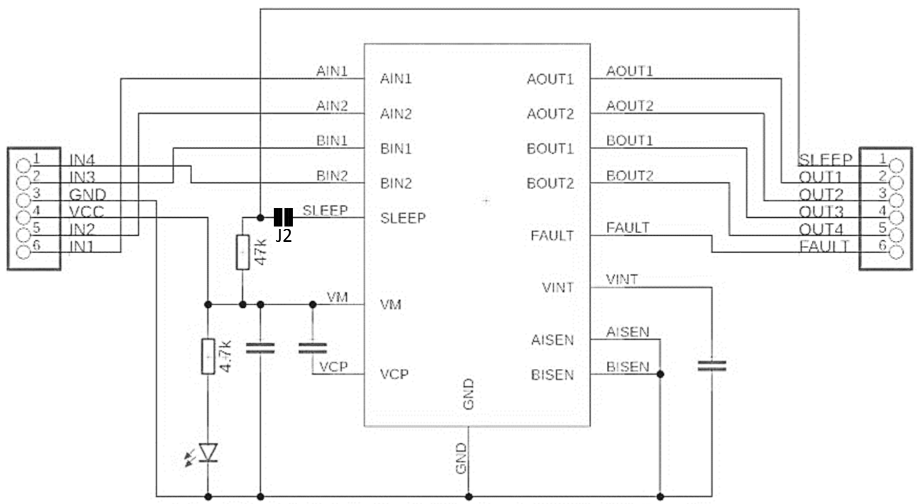


Table 1. H-Bridge Logic

xIN1	xIN2	xOUT1	xOUT2	FUNCTION
0	0	Z	Z	Coast/fast decay
0	1	L	H	Reverse
1	0	H	L	Forward
1	1	L	L	Brake/slow decay

Table 2. PWM Control of Motor Speed

xIN1	xIN2	FUNCTION
PWM	0	Forward PWM, fast decay
1	PWM	Forward PWM, slow decay
0	PWM	Reverse PWM, fast decay
PWM	1	Reverse PWM, slow decay



Motor Control Testing

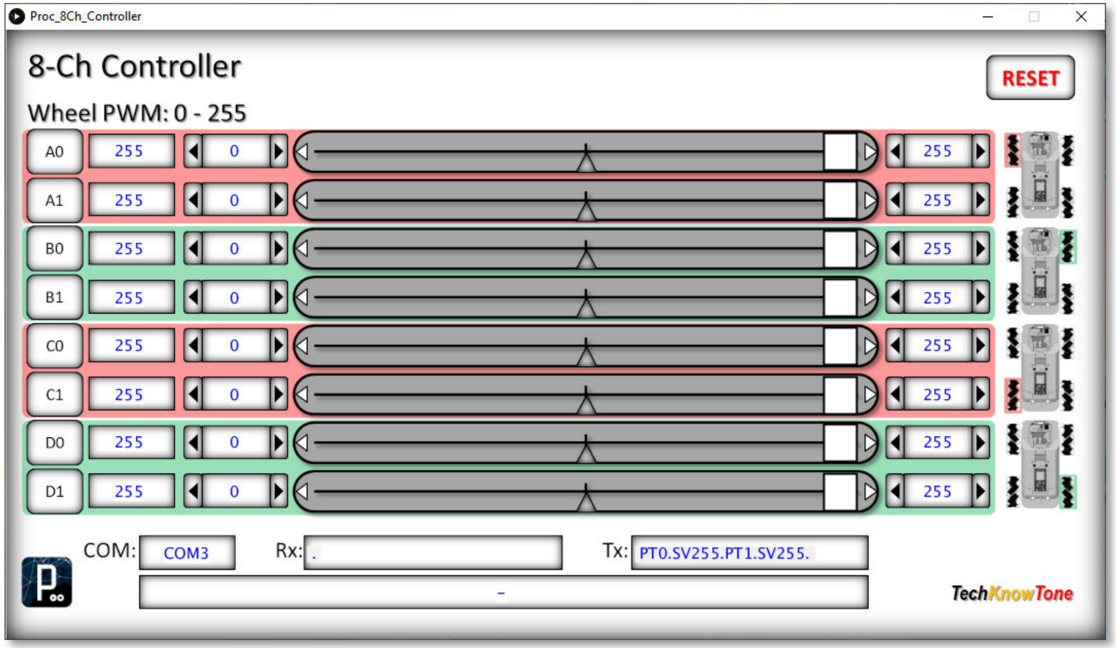
The Omni-Bot code included commands that can be sent over the serial interface, either directly from the Arduino IDE or via a custom Windows app. The 8-Channel Controller app uses the serial interface to send these commands.

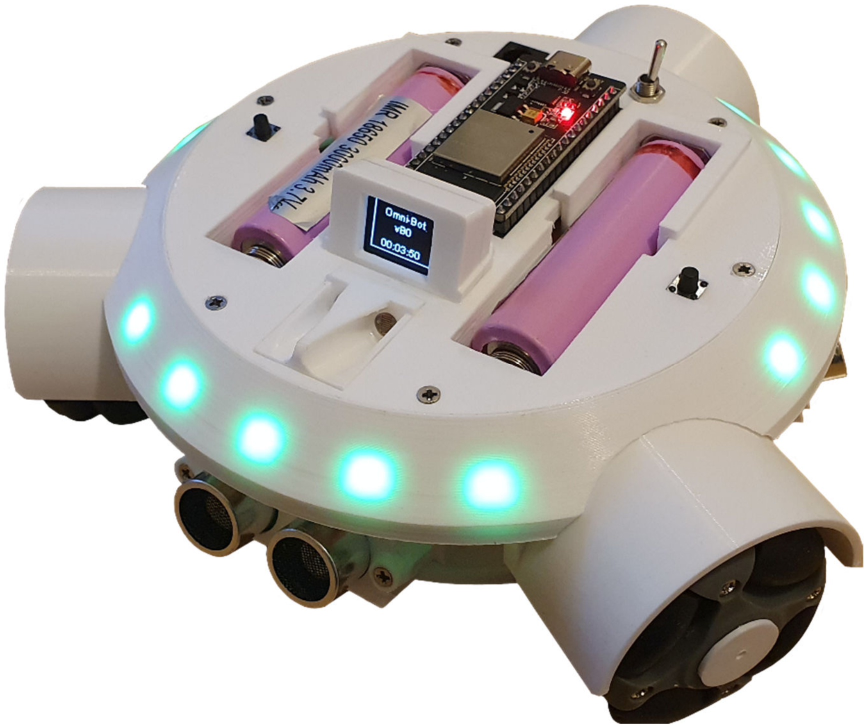
The DRV8833 H-bridge dual channel drivers, one for each motor, each receive two PWM control signals from the micro. Hence the 8-Channel controller groups these channels into four pairs; A0 & A1, B0 & B1, C0 & C1 and D0 & D1. In this application we only require the first three pairs and ignore the D channels.

The app presents sliders for each of the PWM channels, ranging from 0 to 255 in value. Simply click on a channel reference (like A0) to enable/disable a channel pair. Then move the sliders to send commands to the OMNI-Bot. The wheel motors should respond to these changes, and turn in an appropriate direction.

Note that the controller logic assumes one channel will be held at either LOW (0) or HIGH (255), whilst the other channel is in the range 0 – 255. The app slider logic therefore does not allow simultaneous use of both channels. So, one slider must be at its extremes, whilst the other is somewhere in the middle of its range.

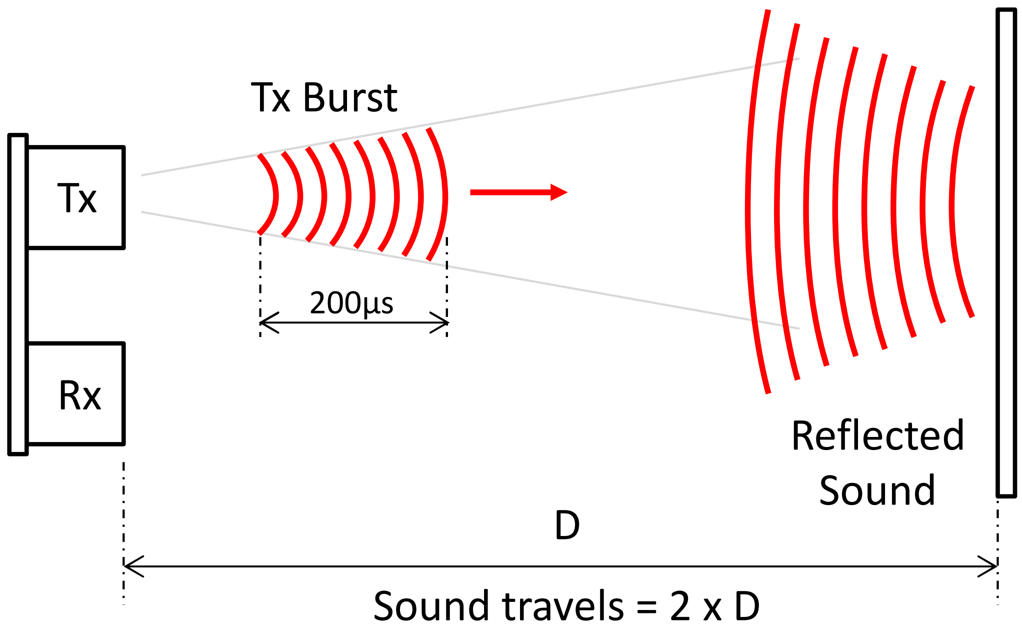
You can experiment with values, to look at things like direction of motor rotation, and minimum PWM start values.



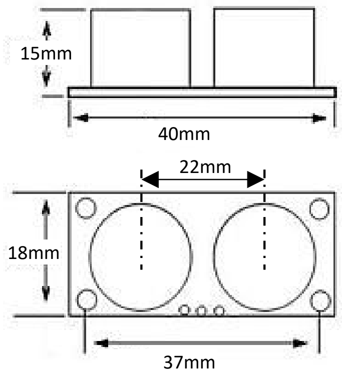


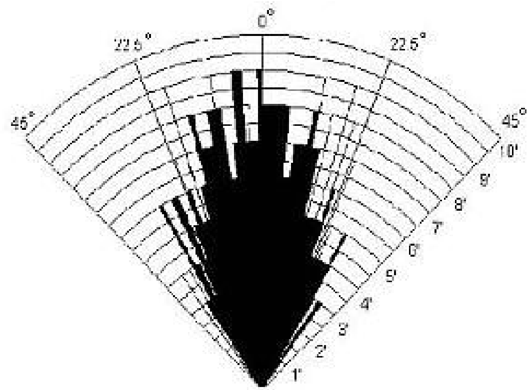
Sensor Features

- Low cost < £2.00
- Low power 2.2mA static current
- 8 cycle 40 kHz sound burst
- Detection range 2cm – 450cm
- Sensor angle 15 degrees
- High precision +/- 3mm
- Easy to use?



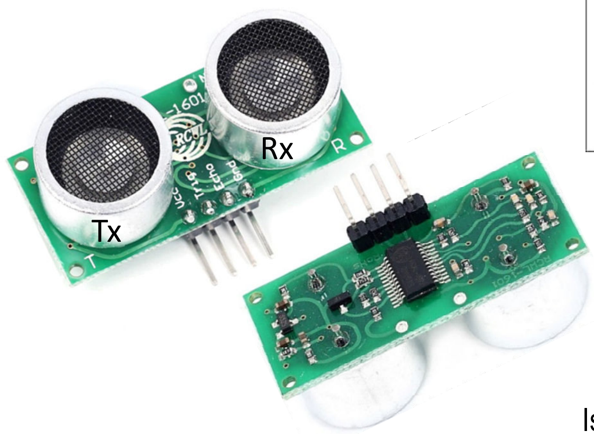
- Provide 10µs HIGH trigger pulse
- Measure width of returned ‘Echo’ pulse

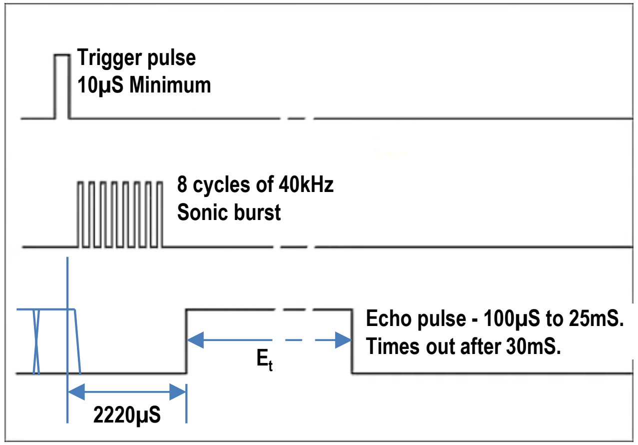




Practical test of performance,
Best in 30 degree angle

Speed of sound:	340 m/s
Distance	= speed x Time / 2
Distance in cm	= echo pulse width in µS / 58
Distance in inches	= echo pulse width in µS / 148
Range:	2cm min – 4.5m max (t = 13ms).

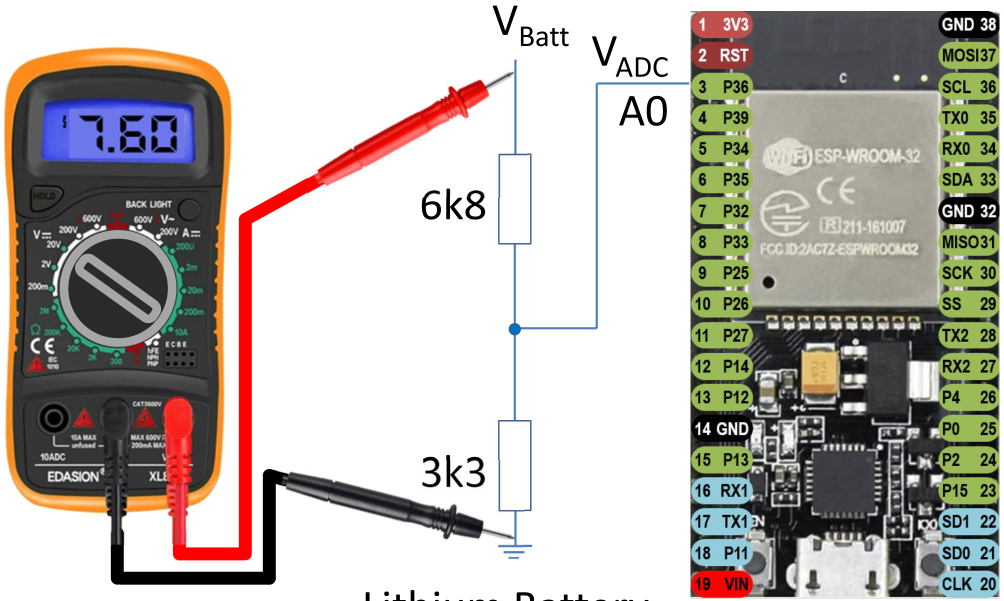




Echo pin state is undefined prior to triggering.
Code looks for positive edge, to time E_t.
Practical measurements up to 400mm take 5ms.

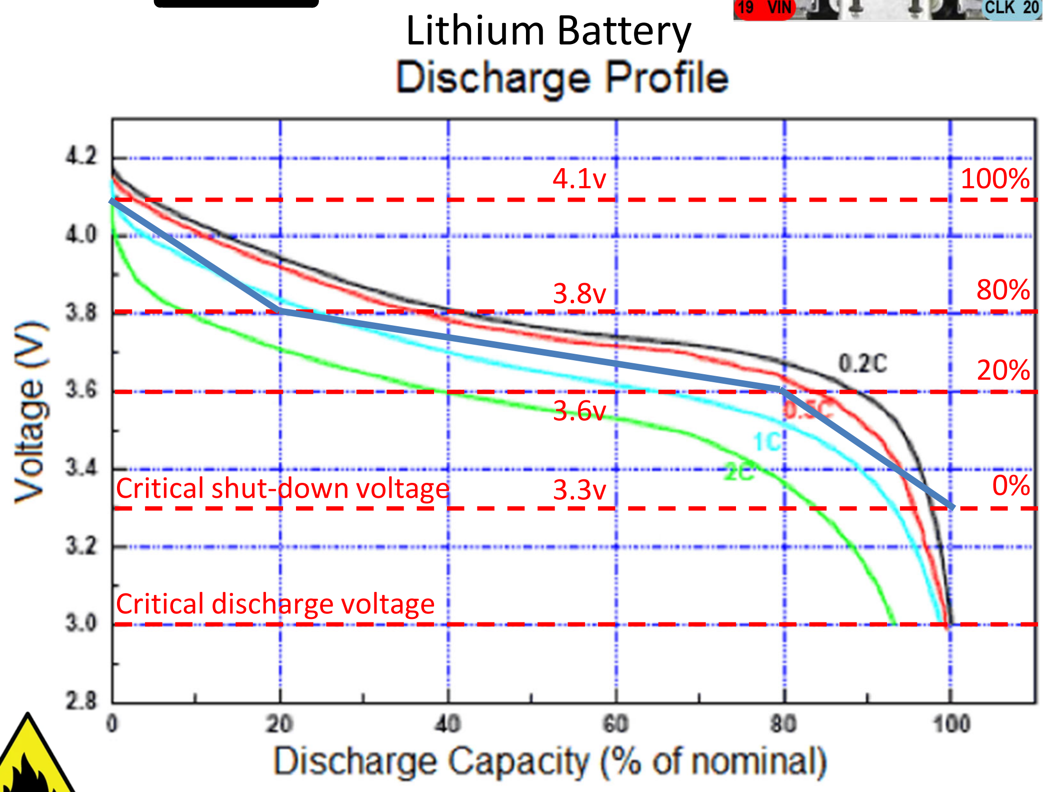
Battery Voltage Calibration

See Lithium discharge curve obtained from the internet. In this analysis the lipo battery consists of two identical batteries connected in series.
Assume fully charged 8.2v battery max voltage is V_{BM} >= 8.4v max (charging)
Set battery warning point at V_{BW} = 7.2v (2 x 3.6v)
Set battery critical point at V_{BC} = 6.6v (2 x3.3v)



The ESP32 is powered via a 5v voltage regulator, connected to the V_{in} pin, but the 6k8 supply sampling resistor is connected to source V_{Batt}.
For ESP32 V_{ADC} == 4095 on 12-bit converter (4095 max).
If we use a 6k8 resistor feeding A0 and a 3k3 resistor to GND, we get a conversion factor of 10.1v == 4095, or 2.47mV/bit, or 405.4 bit/v
Using a Multimeter and a variable DC supply, I determined the following V_{ADC} values for corresponding threshold voltages:

MAX. O.C	V_{OC} = 8.4v, gave A0 = 3226 On V_{ADC} (2 x 4.2v)
MAX: (100%)	V_M = 8.2v, gave A0 = 3136 on V_{ADC} (2 x 4.1v)
HIGH: (80%)	V_H = 7.6v, gave A0 = 2872 on V_{ADC} (2 x 3.8v)
WARNING: (20%)	V_{BW} = 7.2v, gives A0 = 2710 on V_{ADC} (2 x 3.6v)
CRITICAL: (0%)	V_{BC} = 6.6v, gives A0 = 2460 on V_{ADC} (2 x 3.3v)



Discharge: 3.0V cutoff at room temperature.

The code will sample the battery voltage on power-up to ensure it is sufficient, then at every 40ms interval, calculating an average (1/50) to remove noise. It also detects no battery as USB mode.

In the code I have assumed a discharge curve ranging from 8.2v (100%) to 6.6v (0%) capacity, using the overlay lines shown. The rate of discharge is monitored and used to predict the life of the battery in use.

Note: If connected to USB port with internal battery switched OFF the ADC will read a value 5 volts (A0 = 1919) or less. So, if the micro starts with such a low reading it knows that it is on USB power.