
# PiZ-Moto+

**(Pronounced "Pie-Zee Moto Plus")**

## Assembly Instructions (1.0)

**Developed for the Cotswold Raspberry Jam (@CotswoldJam)**

Latest instructions available at:

https://github.com/astro-designs/piz-moto

**Introduction**

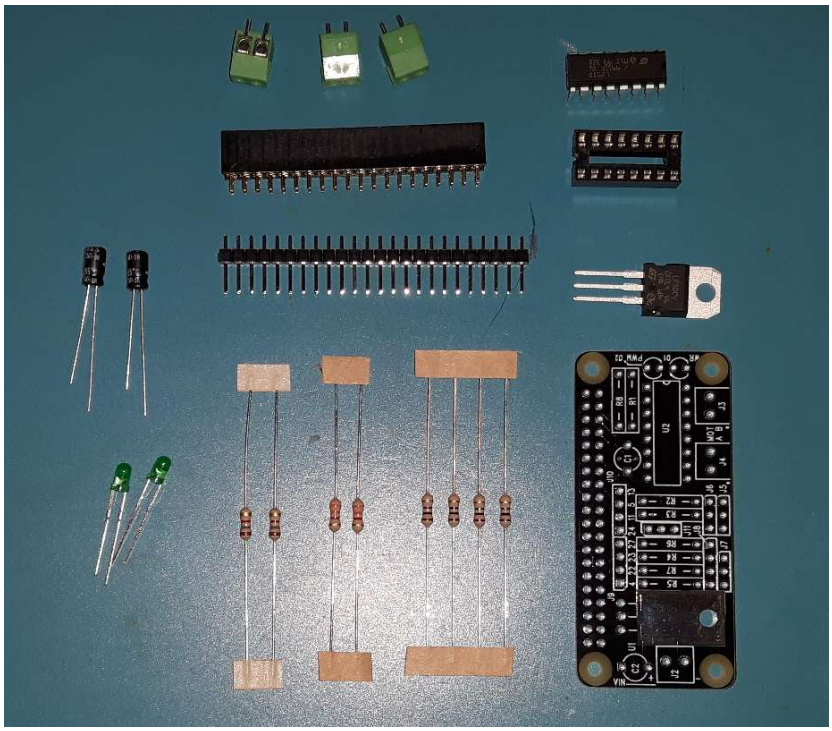
The PiZ-Moto kit is designed for use with the Raspberry Pi 'Zero'. It is based around the Texas Instruments L293DNE quad half-bridge driver which can be used to control up to two motors in two directions. The L293DNE was chosen because it's pretty much the only low-cost motor driver you can find that's 'through-hole' not surface-mount making it ideal for a self-assembly kit like this. We wanted to make this motor driver a little different from others, after all there are quite a few available. So what makes this one different? Firstly, it includes a built-in 5-volt regulator that enables the power source that's used to power the motors to also power the Raspberry Pi Zero. Secondly, it includes interfaces for an infra-red reflecting sensor, an ultra-sonic range sensor and a couple of 3-pin interfaces that support LEDs, low-power radio-control style servos or general purpose I/O. Plus, we've also brought out 8 other spare GPIO pins from the Raspberry Pi's GPIO header to an 8-pin header. For the 5-volt power supply we chose a low-cost leaded low drop-out linear regulator. Being low drop-out means that the motor power supply voltage only needs to be approximately 0.5V greater than the 5V supply needed by the Raspberry Pi. A 4-cell AA battery pack which produces 6V, makes an ideal power source for this motor driver kit. With the motor driver, 5-volt regulator and the extra simple but versatile interfaces on the board, the PiZ-Moto makes it easy to build an intelligent line-following, object avoiding 2-wheel robot using the Raspberry Pi Zero. The final feature of this particular design is that it's fully compatible with the CamJam EduKit #3 so if you've already got software that works with the EduKit #3 then it'll work with the PiZ-Moto too.

And another thing, in this latest version, we've also added a 3-pin power supply header that, together with the 8-pin spare GPIO header, it allows you to stack a second, slightly modified PiZ-Moto+ board on top of the first to double the number of motors that can be driven.

**Contents**

In your kit, you should find:

1) PiZ-Moto+ PCB (x1)
2) 100 Ohm Resistors (x4)
3) 1.8k Ohm Resistors (x2)
4) 3.3k Ohm Resistors (x2)
5) BA50BCT (or similar) Low-dropout Linear Regulator (x1)
6) L293DNE Quad Half-Bridge Driver I.C. (x1)
7) I.C. Socket (x1)
8) 2-way Screw Terminal (x3)
9) 3mm green LED (x2)
10) 24-way Pin Header (x1)
11) 22uF 16V Aluminium Electrolytic Capacitors (x2)
12) 40-way Socket (x1)



**Contents of the PiZ-Moto+ Kit**

**Tools Required**

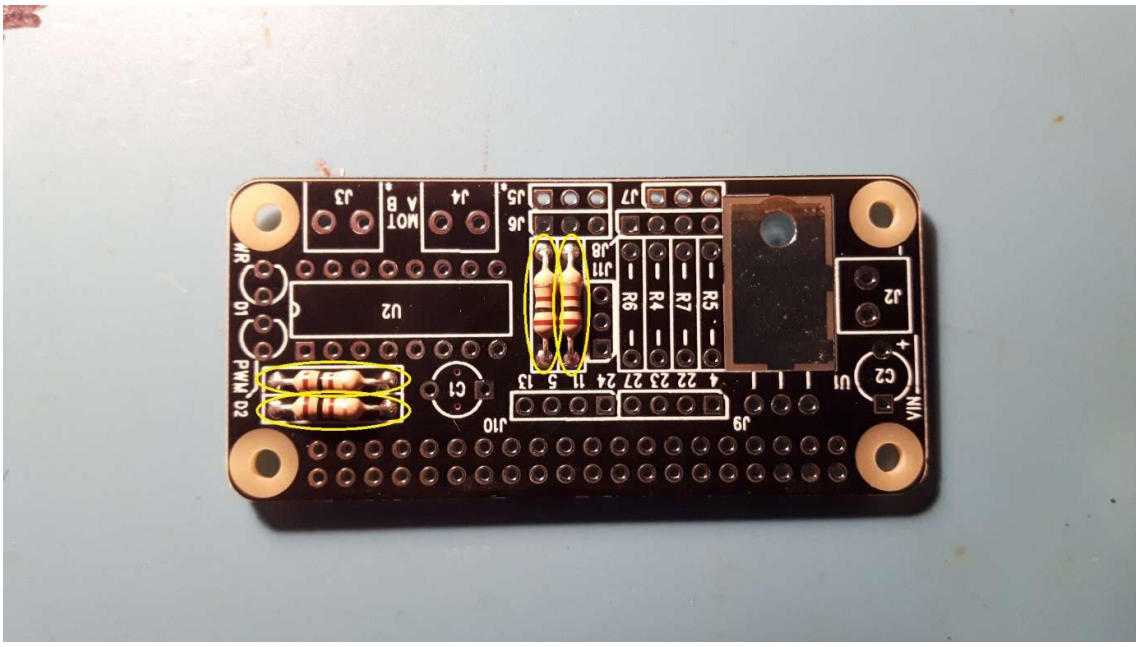
1) A reasonably decent soldering iron
2) Solder wire
3) A pair of snips
4) A pair of snipe-nosed Pliers
5) Optionally, a solder pult de-soldering tool or solder wick in case you need to remove any components or clean up any solder.

**Assembly Instructions**

Right, let's get stuck in…

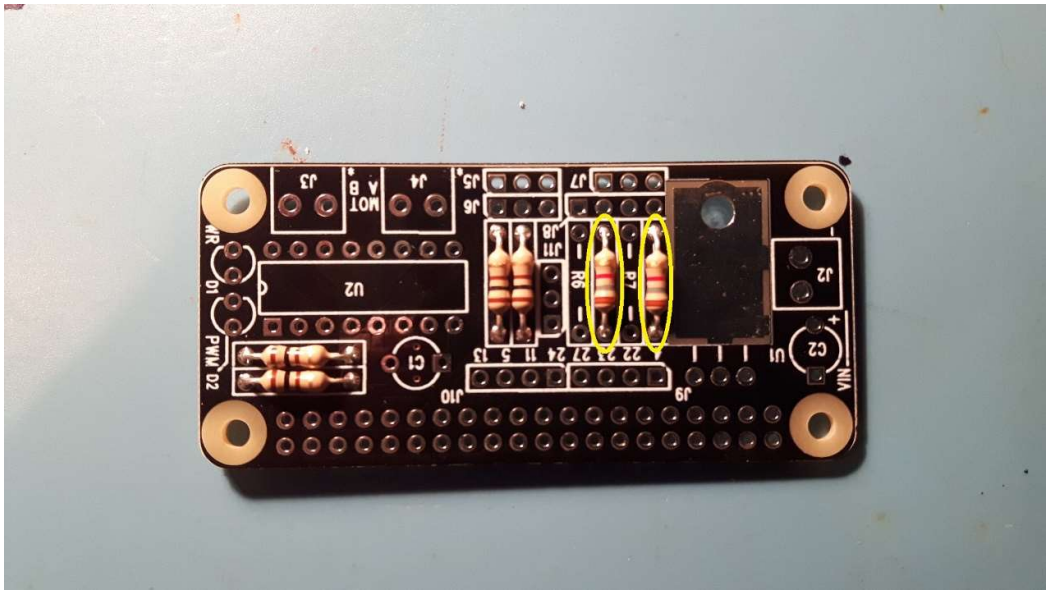
1) It's always best to start with the smallest components so firstly we'll fit the resistors, then gradually move through the taller components. There are three different values of resistors used so let's fit these in turn.

    a. Firstly, fit the four x 100 ohm resistors.
    These are colour-coded BROWN-BLACK-BROWN [GOLD]

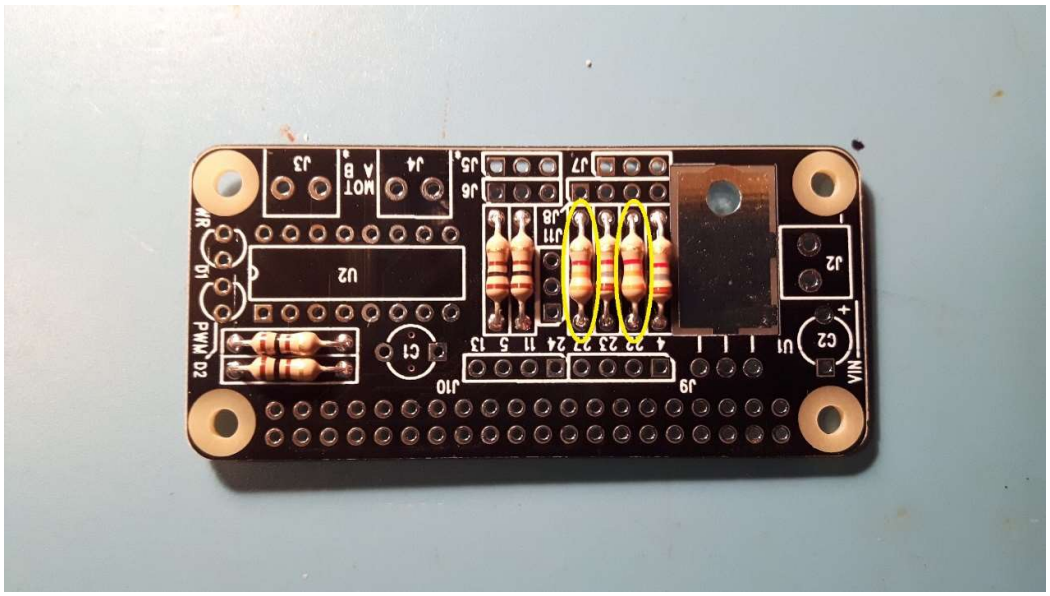



Oh and just to make it look nice and neat, you might want to arrange the gold bands at the same end, or is that just me being finicky?

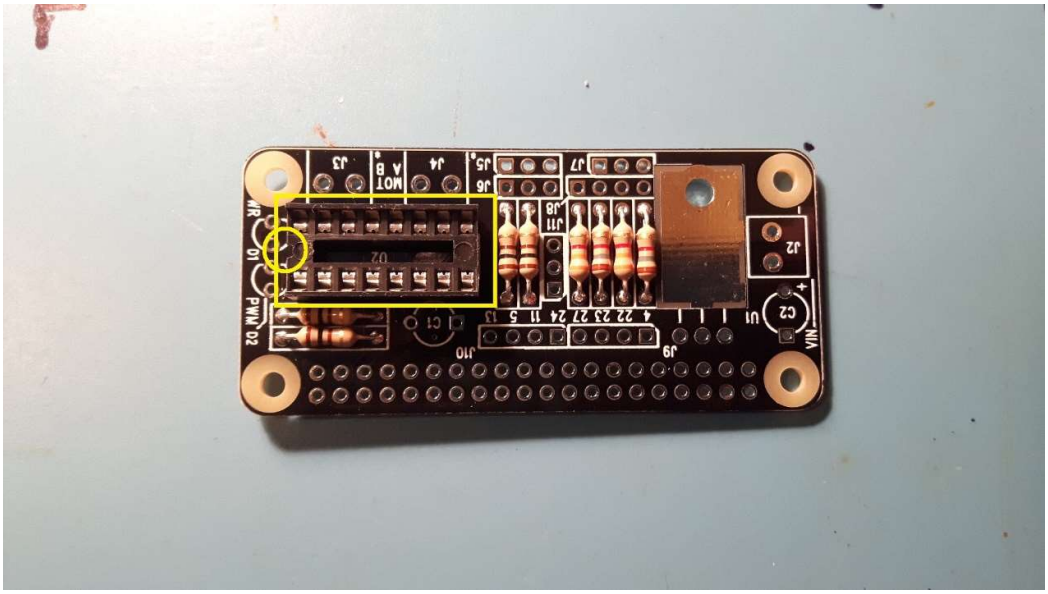
b. Next, fit the two 1.8k ohm resistors.
These are colour-coded BROWN-GREY-RED [GOLD]



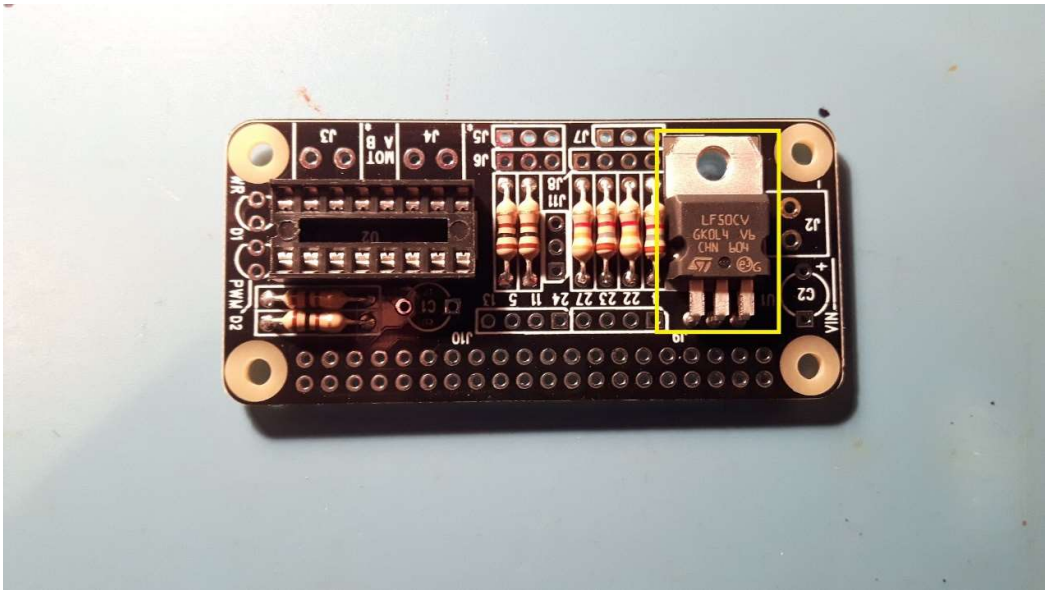
c. Next, fit the two 3.3k ohm resistors.
These are colour-coded ORANGE-ORANGE-RED [GOLD]




2) Next, fit the I.C. socket. Be sure to get the notch at the right end (see circle in yellow), while it makes no electrical difference it does help you to avoid plugging in the I.C. the wrong way around. Note: Don't plug the I.C. in just yet, we'll do that later.

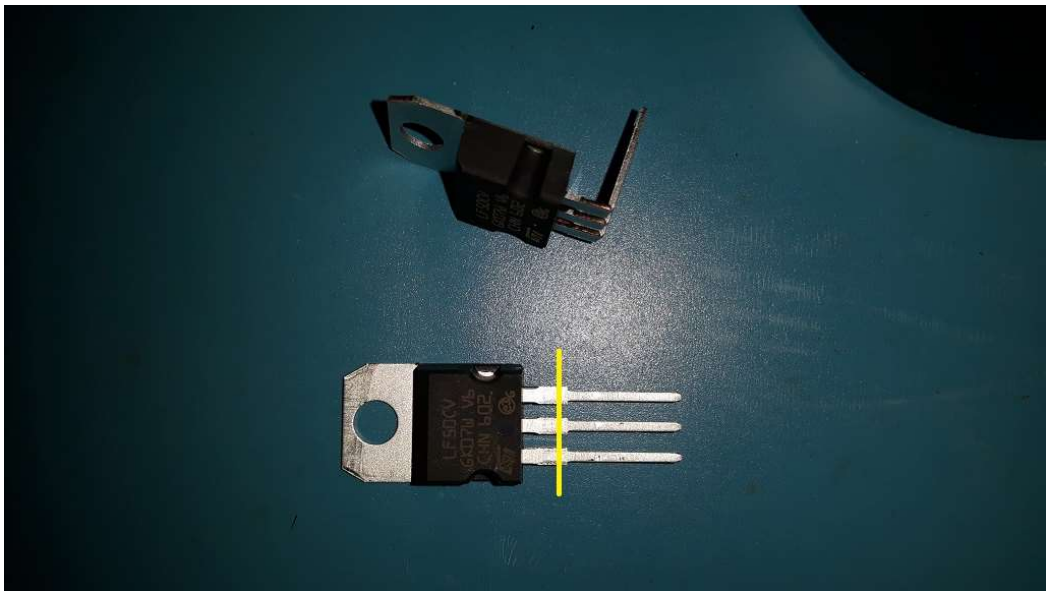


3) Next it's the turn of the linear regulator:

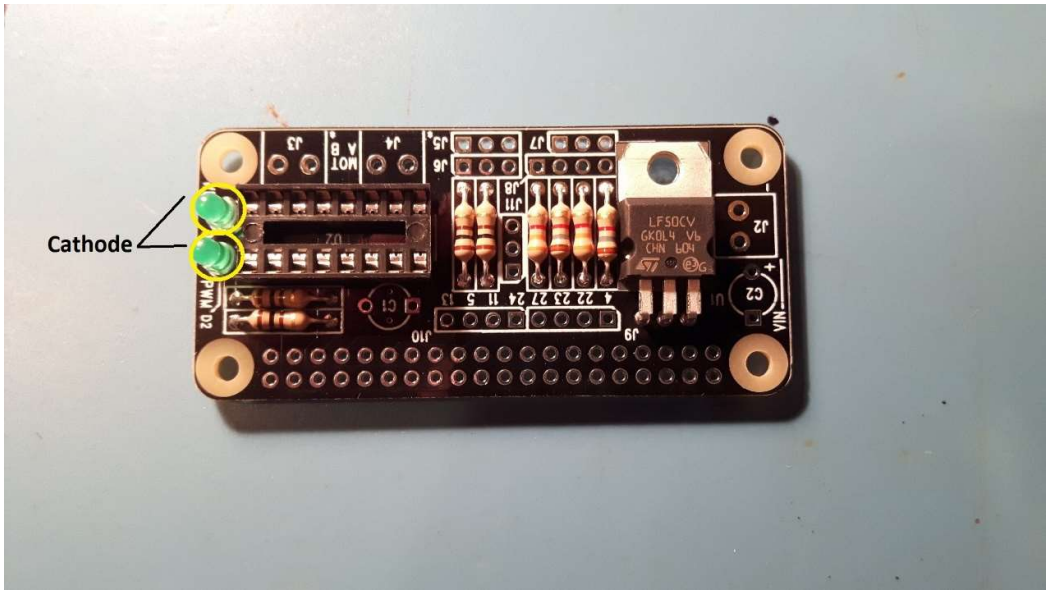



The linear regulator is supplied without the leads bent into position so these must be carefully bent. To get the mounting hole in the regulator to line up correctly, the leads must be bent just a fraction above the shoulder, where the leads get slightly wider (see picture below).

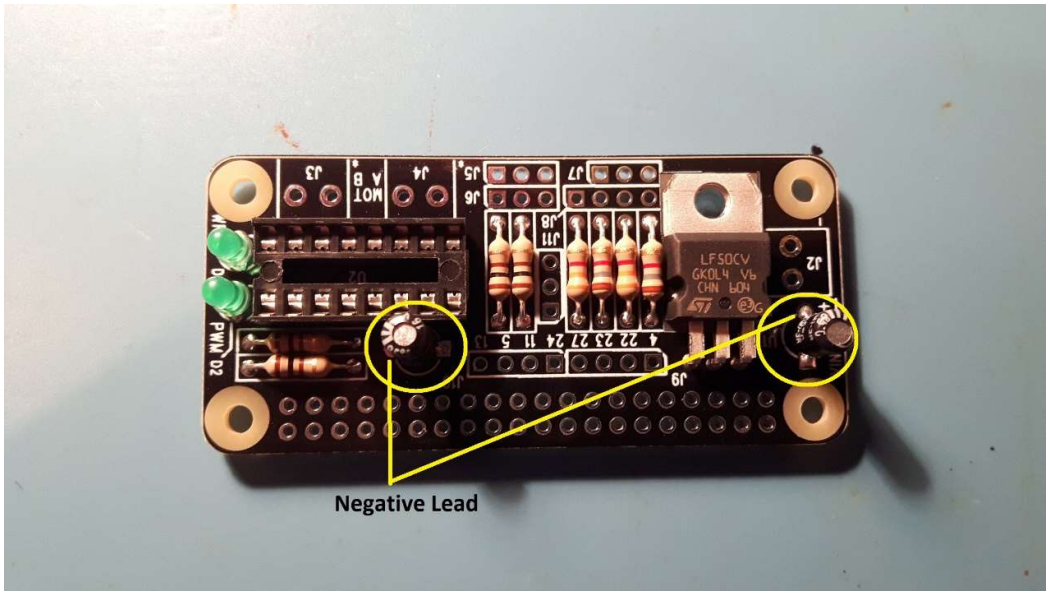
Note – if you prefer to power the motors separately from the Raspberry Pi which does have the advantage that you can then run the Motors at up to 36V, then simply don't fit the linear regulator. See the note in the FAQ section on powering the motors separately from the Raspberry Pi.

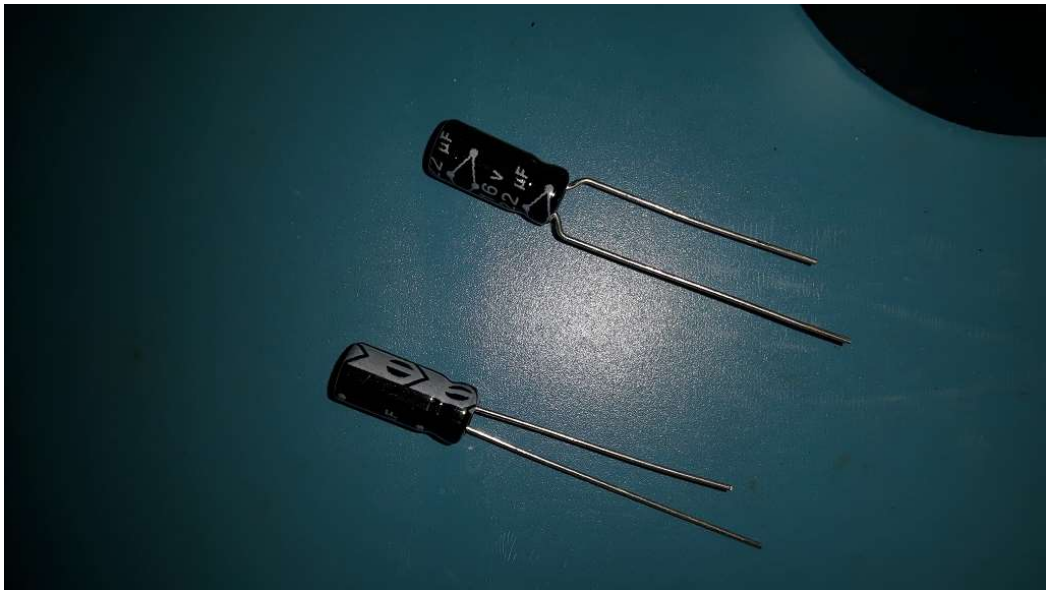


4) Fit the LEDs:
Ensure that the LEDs are fitted the correct way around. The ‘cathode’ terminal is indicated by the shorter lead. The circle outline for the LED has a flat section to mark the ‘cathode’, also marked with the yellow arrow in the picture below. Note: If the leads have been cropped and you want to identify the ‘cathode’, take a close look at the inside of the LED, the cathode is always the larger of two electrodes.

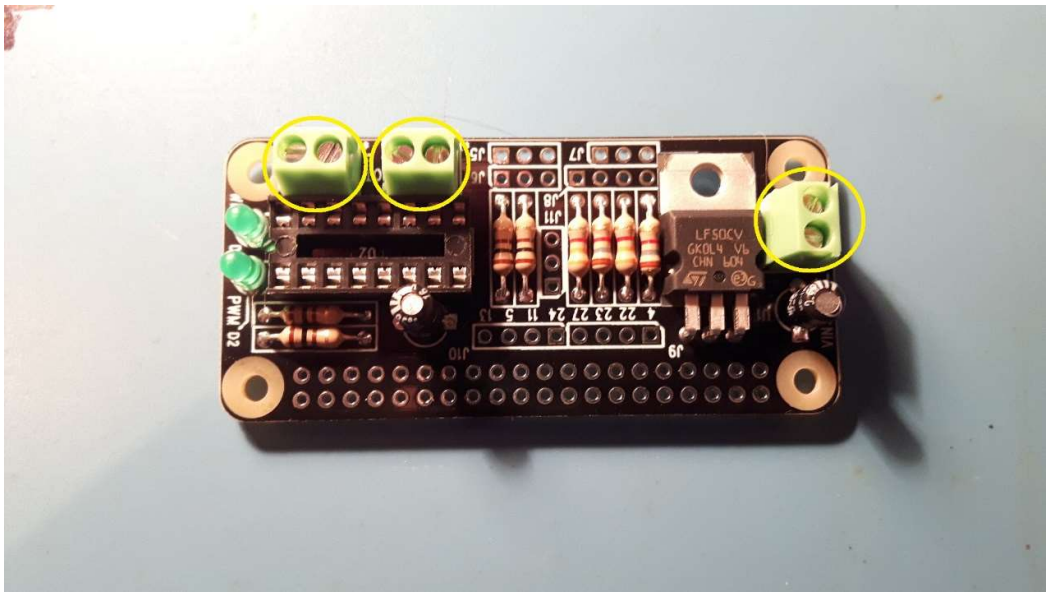


5) Next, fit the two capacitors. Firstly, these need to have the pins ‘formed’ slightly to match the separation of the holes on the board. The pins should be separated by about the same amount as the diameter of the capacitor. See the additional picture below illustrating how they should look. Secondly, it’s important to note that these are polarised capacitors so they MUST be fitted the correct way around. The negative terminal is marked on the capacitor with a stripe and “-“ symbols. The arrows on the picture below point to where the negative terminal should go. One of the terminals for the capacitor shape on the PCB has a square pad, this corresponds to the positive lead on the capacitor. Please note that the capacitors supplied in the kit may be slightly different to the ones shown in the bottom picture below.

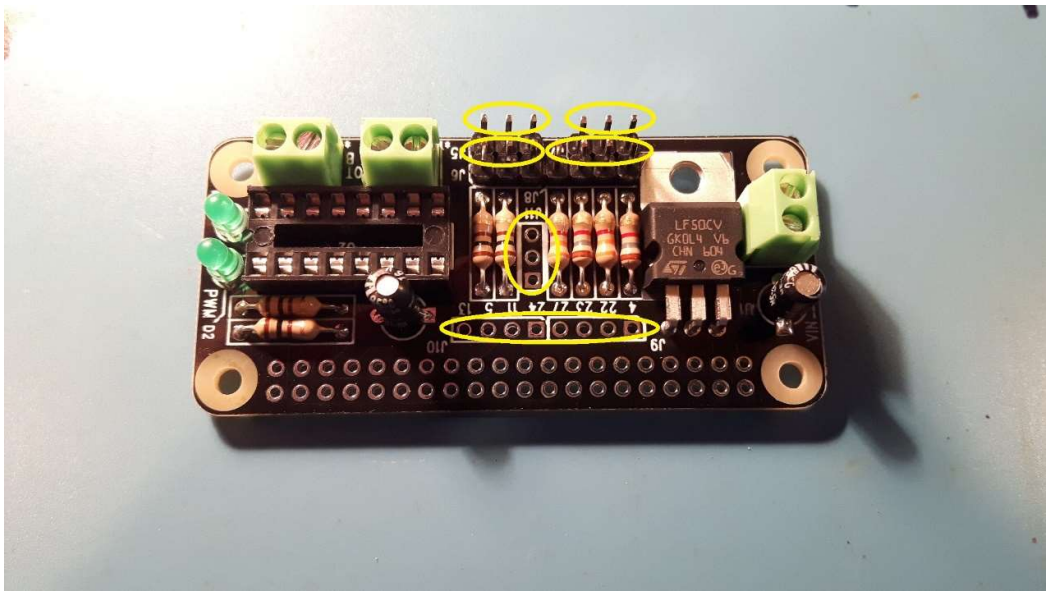


6) Next, fit the three 2-way screw-terminals:
   Try to keep them flat against the surface of the board as you solder them. It also helps when connecting the motor leads if the holes in the side for connecting the wires face outwards (yes, I did put one in the wrong way around once…).

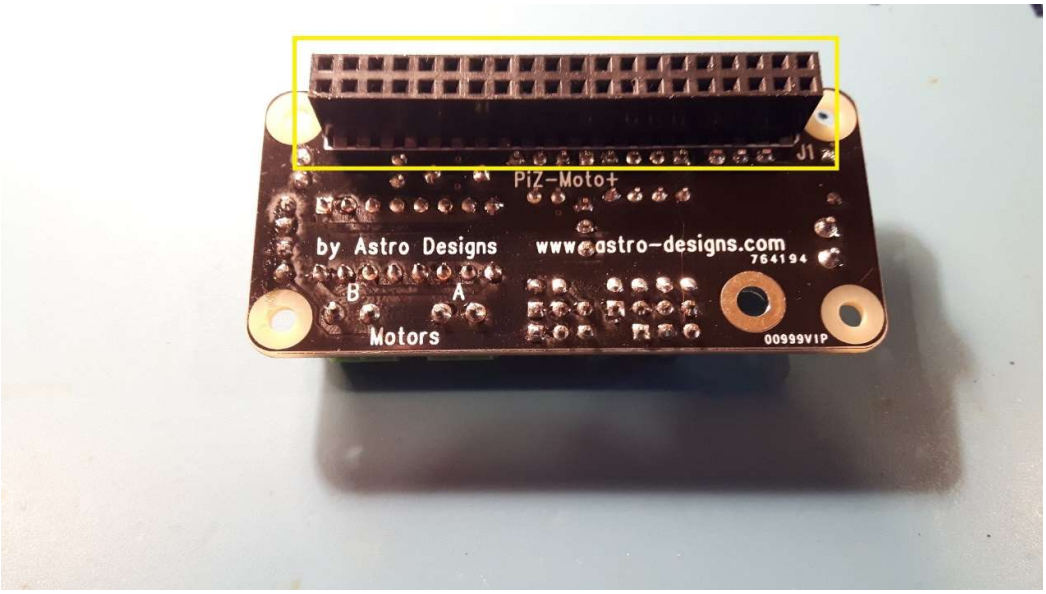


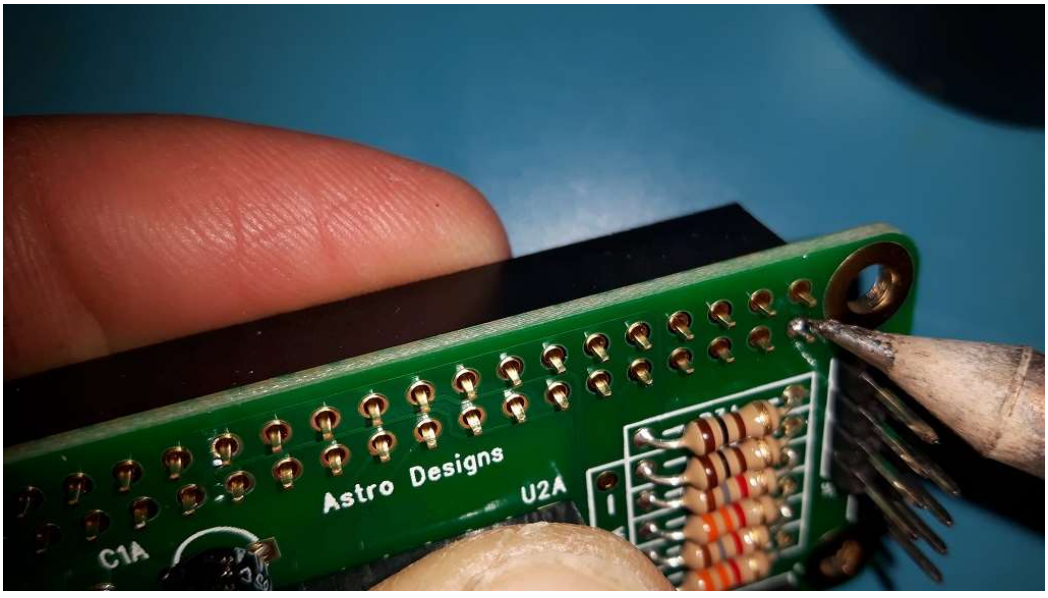
7) Next, fit the pin-headers. Four of these are three-pin, one is four-pin and the last one is 8-pin. These need to be carefully broken off the 24-pin strip provided.



8) Next, fit the 40-way socket. Note this needs to be assembled onto the opposite side of the board with the pins coming through to the top side.

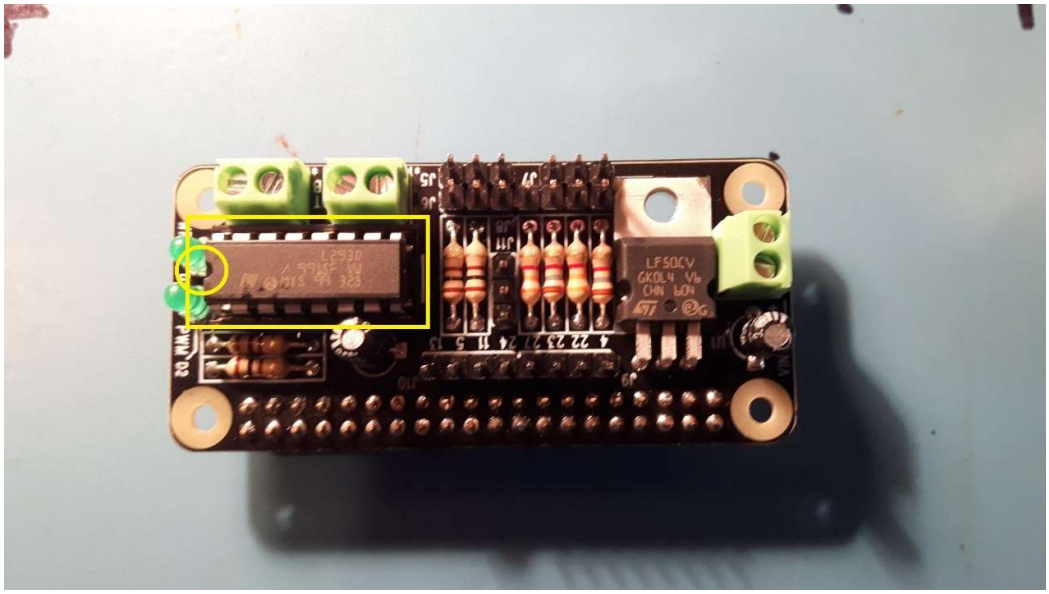
   It's best to solder just two pins in opposite corners to start with, then while pressing the connector into the board, re-melt the solder on those two pins just to ensure the connector is properly flush with the board. Once this is done then the rest of the pins can then be soldered.



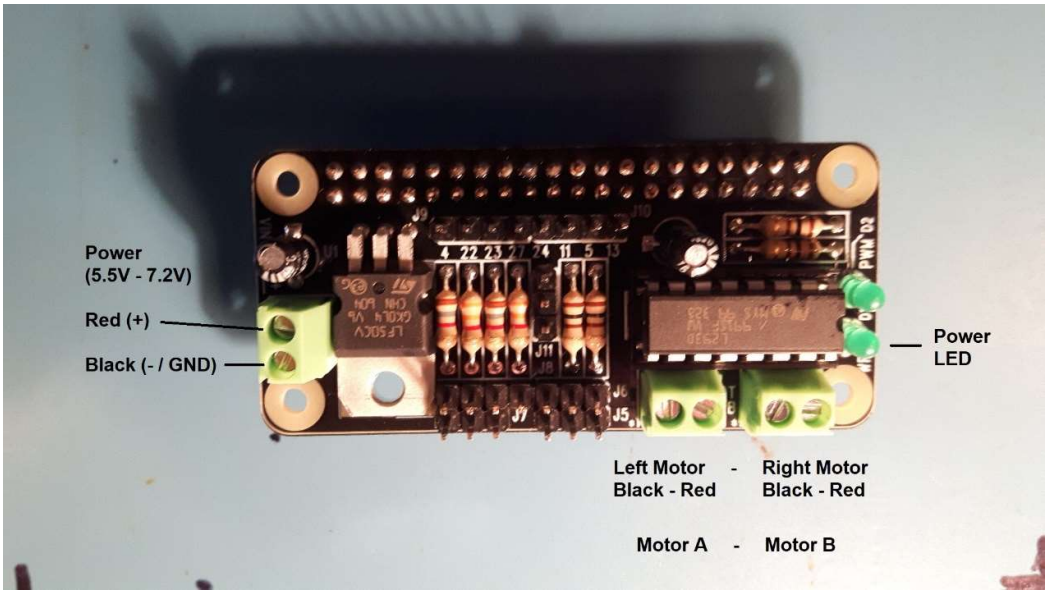



9) That's it! Ok, you should now do a quick check to ensure that all components are fitted and are fitted correctly. Check:
    a. All components are fitted;
    b. All pins are soldered cleanly;
    c. All long-leads have been snipped-off;
    d. The I.C. socket is fitted the correct way around;
    e. Capacitors are fitted the correct way around;
    f. The LEDs are fitted the correct way around;

10) Finally, plug the L293DNE motor driver I.C. into the I.C. socket. Make sure it's fitted the correct way around, the notched end should appear next to the LEDs.



**Connecting the Motors**




Each pair of screw terminals above connects to a single motor. In theory you could connect two motors in parallel to use four motors, with two motors on a single screw terminal but start with one motor per screw terminal pair. The motor driver is rated to 600mA per motor. The connections shown above are completely compatible with the CamJam EduKit #3.

**Connecting the Power Supply or Battery:**

It's important to get the polarity correct when connecting the battery. Getting this wrong could damage the board. The ground / black / 0V wire from the battery should connect to the battery screw terminal that's closest to the corner of the board (marked in yellow as "-" in the picture below). The positive / red wire from the battery should connect to the other screw terminal, marked below in yellow as "+".

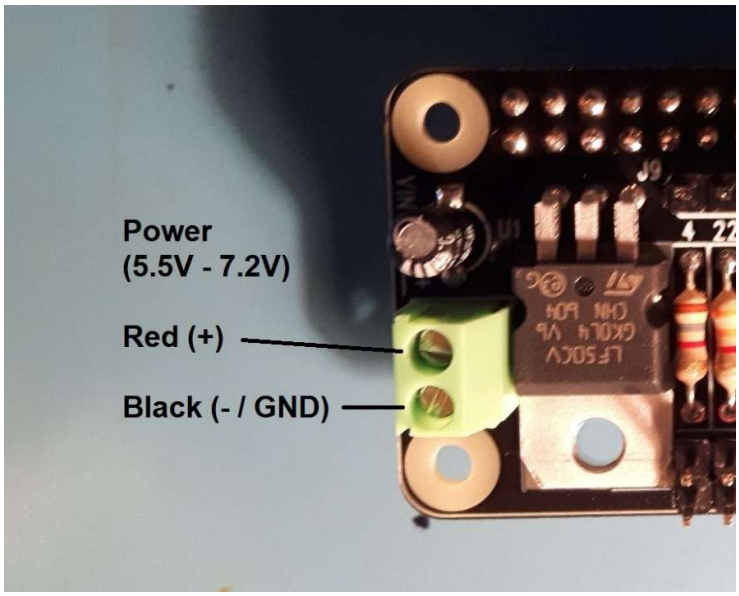
It's best to have a switch in line with your battery so that it's easy to turn things off. Make sure the switch is off when connecting the battery to the board.

**Important note**: The PiZ-Moto is designed to be powered through the screw-terminal power connector. The PiZ-Moto uses an on-board 5V regulator to provide a 5V regulated power supply to the Raspberry Pi Zero through the Pi's +5V & GND pins on the Pi's GPIO connector. Please do not power the Pi through the Pi's USB power connector while connected to the PiZ-Moto. While this is unlikely to result in any damage, you will not be able to provide power to the motors using the Pi's USB power connector so it's recommended that you don't use that connector when using the PiZ-Moto.

It's designed to run off four "AA" cells which should provide approximately 6V. But it will work on as little as 5.5V. The recommended upper limit is 7.2V although it will tolerate more. How much it can tolerate comes down to how much current is needed by the Pi Zero. If you connect a USB WiFi dongle or a camera then I wouldn't recommend anything more than 6V. Also check the temperature of the regulator (U1A) as this gets hot if the Pi Zero draws more current.

It's really best suited to minimal applications where the loading on the Pi Zero is very little. A Bluetooth dongle or Wireless Keyboard Dongle consumes very little power so these are ideal.

One final note on the battery side of things, you don't need to (& shouldn't) power the Pi Zero separately. The Pi Zero is powered from the regulator on the PiZ-Moto.

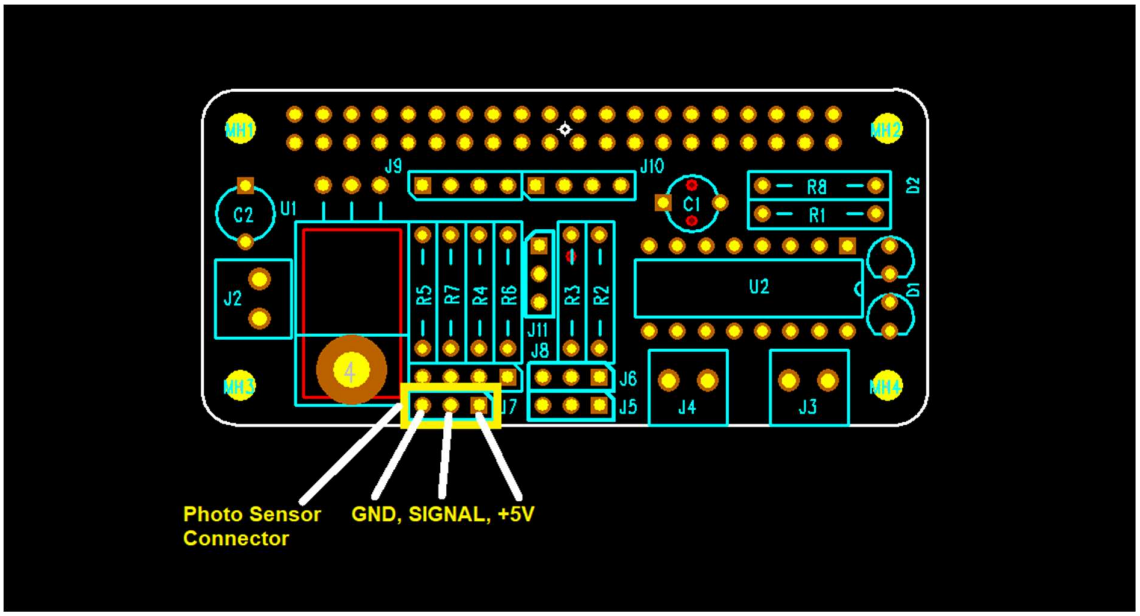

Power
(5.5V - 7.2V)
Red (+)
Black (- / GND)

**Connecting the Photo Sensor**

The Photo Sensor header is designed to connect directly to a photo-reflector / line sensor like the one found in the CamJam EduKit #3. Simply connect the three pins on the PiZ-Moto Photo Sensor Header to the three pins on the photo-reflector / line sensor, making sure you get the pins connected to the right terminals on the photo-reflector board and you'll be ready to test.

Please note that the ordering of the pins on the PiZ-Moto board are different to the order of the pins on the line sensor found in the EduKit #3 so please take care when connecting the sensor.

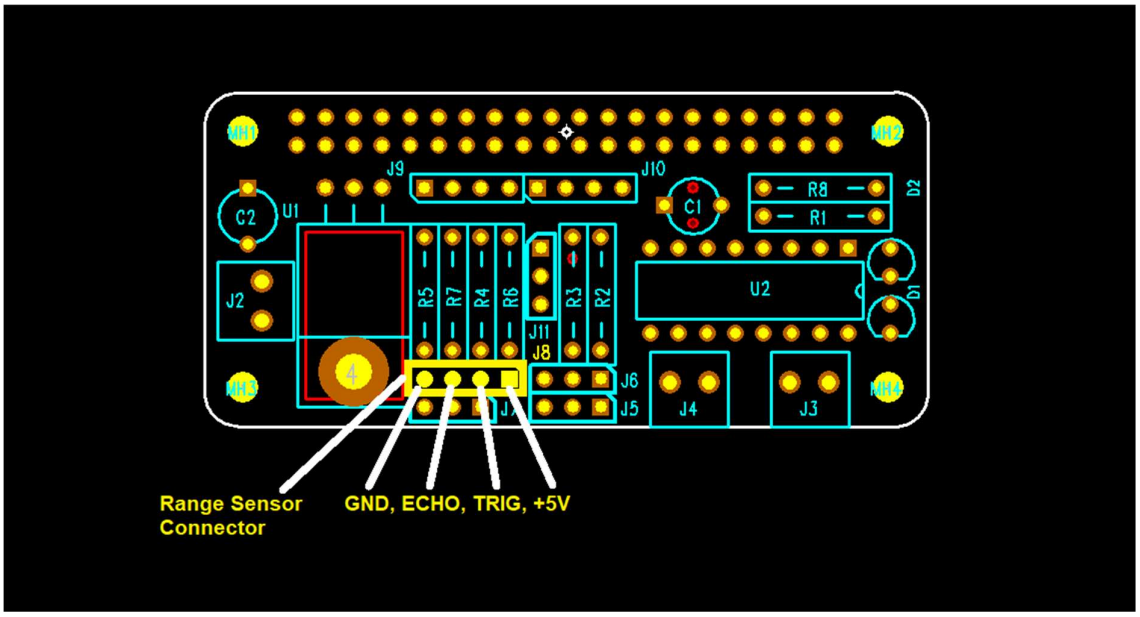
The signal connection on the PiZ-Moto Photo Sensor Header can also be used as a general-purpose 5V input and is connected to the Pi Zero's GPIO-25 pin via a 5V to 3.3V level shifting network.



**Connecting the Range Sensor**

The Range Sensor header is designed to connect directly to an SR-04 range sensor, as found in the CamJam EduKit #3. The SR-04 is also available separately from several suppliers. Simply connect the four pins on the PiZ-Moto Range Sensor Header to the four pins on the SR-04, making sure you get the pins connected to the right terminals on the SR-04 and you'll be ready to test.

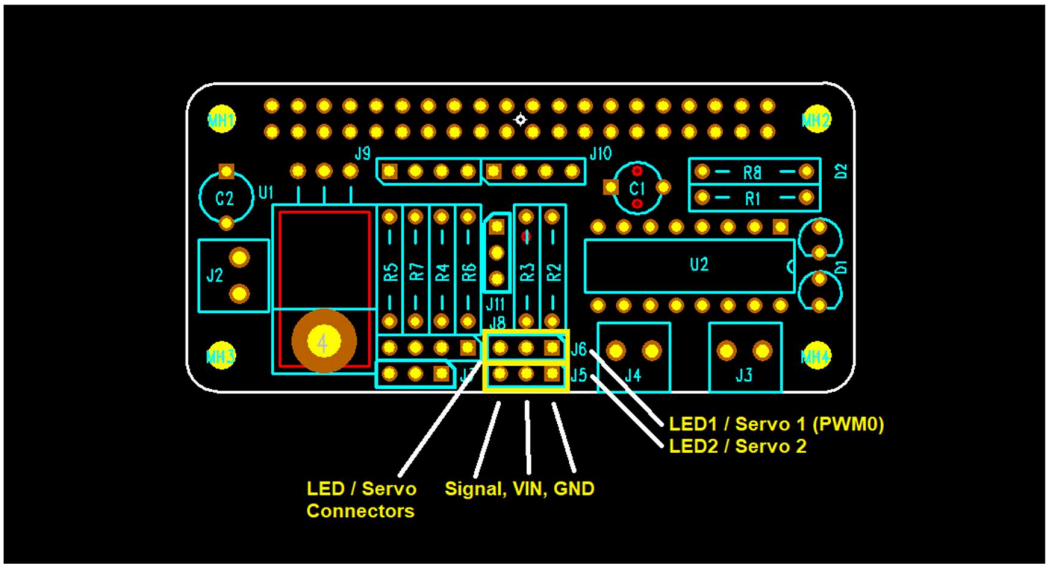
The 'Echo' connection on the PiZ-Moto Photo Sensor Header can also be used as a general-purpose 5V input and is connected to the Pi Zero's GPIO-18 pin via a 5V to 3.3V level shifting network. The 'Trig' connection can be used as a general-purpose 3.3V input or output and is connected directly to the Pi Zero's GPIO-17 pin.



**Connecting the LEDs or Servos**

The LED header is really a couple of general purpose I/O pins, each paired with it's own 'GND' pin. The two LED pins have a 100 ohm resistor in series with it's connection the header on the Pi Zero. This both protects the I/O on the Pi Zero as well as allowing an LED to be connected directly across the two pins with no need for an extra resistor in line with the LED. In our examples here, we just connect a couple of LEDs directly but you could use these pins as inputs or outputs.

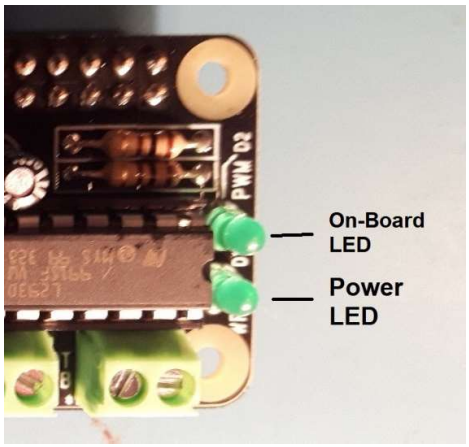
The LED1 / Servo1 Signal connection on the LED Header is connected to the Pi Zero's GPIO-16 pin via a 100 ohm resistor. The LED2 / Servo2 Signal connection is connected to the Pi Zero's GPIO-19 pin via a 100 ohm resistor. Note that GPIO-16 can be configured as PWM0 on the Raspberry Pi which means that a servo on this connector can be driven by a 'Hardware PWM' signal instead of a software based PWM signal. This means the servo position will be more accurate, less 'noisy' and it also means that this connector can be configured to drive Neopixel RGB LED strings.



**The On-board LED**

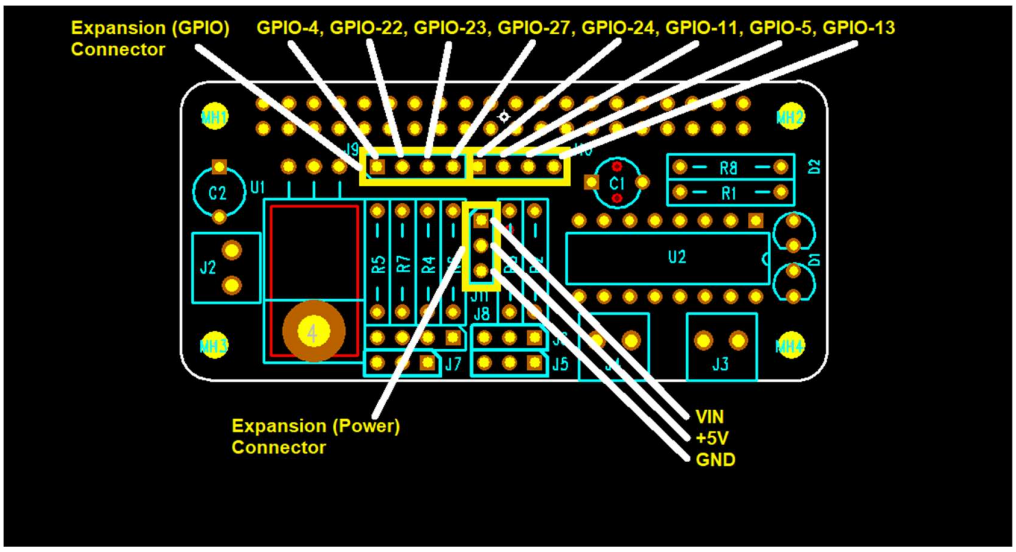
We've added an on-board LED to this later version of the PiZ-Moto because we found that this is particularly useful to use as a status indicator when running the PiZ-Moto+ & Raspberry Pi in headless mode. The on-board LED is located next to the Power LED.

The on-board LED is connected to and can be controlled using GPIO-6 (BCM) on the Raspberry Pi



**Expansion**

There are additional pin headers on the PiZ-Moto+ to provide access to power and other GPIO pins on the Raspberry Pi. This provides limited support to allow you to connect other functions to the Raspberry Pi with a PiZ-Moto+ fitted to it.




**Testing**

1) Firstly, without connecting the PiZ-Moto+ to the Raspberry Pi, connect the power supply or battery and switch it on (if there's a switch), making sure that you get the polarity of the power supply correct!
2) Check that the Power LED comes on. If it doesn't turn it off immediately and double-check the polarity of the power supply;
3) If you have a volt-meter, measure the voltage on the "+5V" terminals of the 40-way socket. These should be very close to 5.0V. You can also measure this across the Photo Sensor connector or Range Sensor connector – see the pictures earlier on in this document to identify which pin is GND and which pin is +5V;
4) Disconnect the power supply or battery and check that the LED goes off;
5) Connect the PiZ-Moto to a Raspberry Pi Zero. Take care to ensure that the connector lines up correctly and is not offset by (say) one pin. (I've done that once and now have a PiZero that's minus one GPIO output ☹)
6) The Python script PiZ-Moto+_test.py is the best way to test the board. If you connect the battery, motors, LEDs, line sensor & proximity sensor and run the script, it'll quickly exercise all functions on the PiZ-Moto allowing you to check it all works. You'll need a Pi Zero with HDMI display & USB keyboard connected.
    a. Plug a small DC brushed motor into each motor terminal;
    b. Connect the 3-pin photo sensor header to a photo-reflector / line detector;
    c. Connect the 4-pin range sensor header to a SR-04 ultrasonic range sensor;
    d. Connect the LED / Servo headers to a couple of LEDs (Connect 'Signal' to the Anode or long lead and 'GND' to Cathode or short lead)

You can also use the worksheets that can be downloaded for the CamJam EduKit #3. These are an excellent way to test and learn more about the functions provided by the PiZ-Moto+ and we highly recommend the EduKit3 as a kit for your Raspberry Pi if you want to get a kick-start in understanding how to interface motors & sensors to your Raspberry Pi.

7) There are examples and a Python Package for the PiZ-Moto on GitHub. To install the package and examples:
    a. Log on to the pi as usual, opening a terminal window if you're in the Pixel GUI and navigate to the /users/pi/home folder;
    b. Clone the PiZ-Moto GitHub folder into piz-moto

        *git clone https://github.com/astro-designs/piz-moto.git*

    c. The examples can be found in the examples folder within the piz-moto folder;
    d. The piz-moto 'package' can be loaded into any Python code running in the home folder using import piz-moto;

8) Connect the battery again and switch it on.
9) Run the Python script [piz-moto_test.py], following the instructions on-screen or over a terminal;

If you encounter any problems, maybe a motor only works one way, one or more of the sensors doesn't work, one or both LEDs doesn't work, then double check all soldering, re-flowing any joints that you're not sure about.

**Prerequisites of the example code**

If you're going to use the Bluetooth based "Wii-Mote" controller then you'll need to install the CWiid module so that Python can talk to the Wiimote.

Install cwiid using

sudo apt-get install python-cwiid

**Examples**

1) The Python script PiZ-Moto_Wii.py is designed to control a 2-motor wheeled robot with line-following and object avoidance functions. It's also primarily designed to work with a BlueTooth based "Wiimote" controller.

   www.astro-designs.com/downloads/piz-moto/examples/piz-moto_wii.py

2) The Python script PiZ-Moto_ kb.py is designed to control a 2-motor wheeled robot with line-following and object avoidance functions. This version is primarily designed to work with a wireless USB keyboard.

   Note: the PiZiMoto_kb.py script isn't ready yet but we'll hope to get it ready early July 2016.

   www.astro-designs.com/downloads/piz-moto/examples/piz-moto_kb.py

3) The board is designed to be 100% compatible with the CamJam EduKit #3 so the worksheets designed for the EduKit #3, and the sensors that come as part of that kit, will all run on the PiZ-Moto. We highly recommend getting hold of the EduKit #3 if you want to get going playing with motors with the Raspberry Pi.

**Frequently Asked Questions**

Q) What's the recommended power supply?

A) It was designed to run from 6V. It can handle more and has been tested at 9V, but only with minimum loading on the 5V supply rail. And it can take less, as little as 5.5V. But the recommended supply is 6V. Four x AA cells make an ideal power supply.

Q) Can it use rechargeable AA cells?

A) Rechargeable AA cells are slightly lower voltage than Alkaline AA cells. Rechargeable AA cells (NiMH) are 1.2V So four cells make 4.8V. This is under the minimum 5.5V supply so these will not work. However, it will work well with five (6.0V) or six (7.2V) NiMH rechargeable AA cells.

Q) Can it support a WiFi dongle on the Raspberry Pi?

A) Yes. They can be a little power-hungry and can cause the 5V regulator to run quite hot but it can work. We would recommend that you run off a power supply voltage of no more that 6V when using WiFi to help keep the temperature of the 5V regulator down. We also recommend screwing the 5V regulator to the PCB using an M3 screw. No need to insulate the screw as the tab of the regulator is already electrically connected to the copper pad under the regulator. Add a little heatsink compound if you have some to improve the thermal connection to the PCB. The PCB will then help sink some of the heat generated in the regulator.

Q) Will it work with the Raspberry Pi A, B, B+, 2 or 3?

A) It's designed for the Pi Zero, because the Pi Zero has a lower power consumption than other versions of the Raspberry Pi. It's not designed to supply much current on the 5V power rail, that's how we get away with using an easy to assemble & low-cost 5V regulator. However, depending on what you connect to the 5V supply, yes it can possibly work with other versions of the Raspberry Pi. The 5V regulator is rated to supply up to 1A so the total current draw on the 5V supply, including the Pi, any USB dongles and anything loading the 5V pins on the Pi's GPIO connector must be less than 1A. However, since the 5V regulator is a 'linear regulator' it's not as efficient as a 'switching regulator' and it can get quite hot, especially when the supply voltage is greater than 6V. If you're

going to load up the 5V supply rail with current-thirsty USB dongles (e.g. WiFi) or plug heavy loads into the GPIO connector then you should try to keep the supply voltage to around 6V. Also, keep an eye on the temperature of the regulator. If it gets hot then you probably need a heatsink to help keep the temperature down or you need to reduce the load on the 5V supply.

Q) How much can I connect to the 5V supply.

A) Please see the previous answers related to WiFi dongles and using it with the Raspberry Pi 2 or 3.

Q) Can I power the Raspberry Pi from the 5V USB power or data connectors?

A) It is not recommended but it does work. If you're connecting a USB power supply to the Pi Zero while it's plugged into the PiZ-Moto, disconnect the power supply to the PiZ-Moto.

Q) Can I use a separate power supply for the motors and a separate USB power supply for the Raspberry Pi?

A) If you want to do this, it's advised to not fit the Linear Regulator. If the Linear Regulator is not fitted then it is perfectly safe to use a separate power supply for the motors using the Power Connector on the PiZ-Moto+ and power the Raspberry Pi from the USB power connector on the Raspberry Pi as normal. You can then power the motors at up to 36V but do not connect a servo to the servo connector if using a supply larger than 7.2V as you will very likely destroy the servo.

Q) Can I run the camera on the Pi Zero when it's powered from the PiZ-Moto?

A) We've not tested this at the time of writing this document but we believe this is probably ok. The current consumption of the camera is around 200 – 400mA (TBC) which should be fine. We'd recommend using a power supply voltage of no more than a 6V supply for this. If you try using the camera or any extra load on the 5V supply, keep an eye on the temperature of the regulator.

Q) Which motor terminal is for the left motor & which terminal is for the right motor?

A) It really depends on how your software is written but if your using our example code or the code from the EduKit #3 then J3 should connect to the right motor, J4 should connect to the left motor.

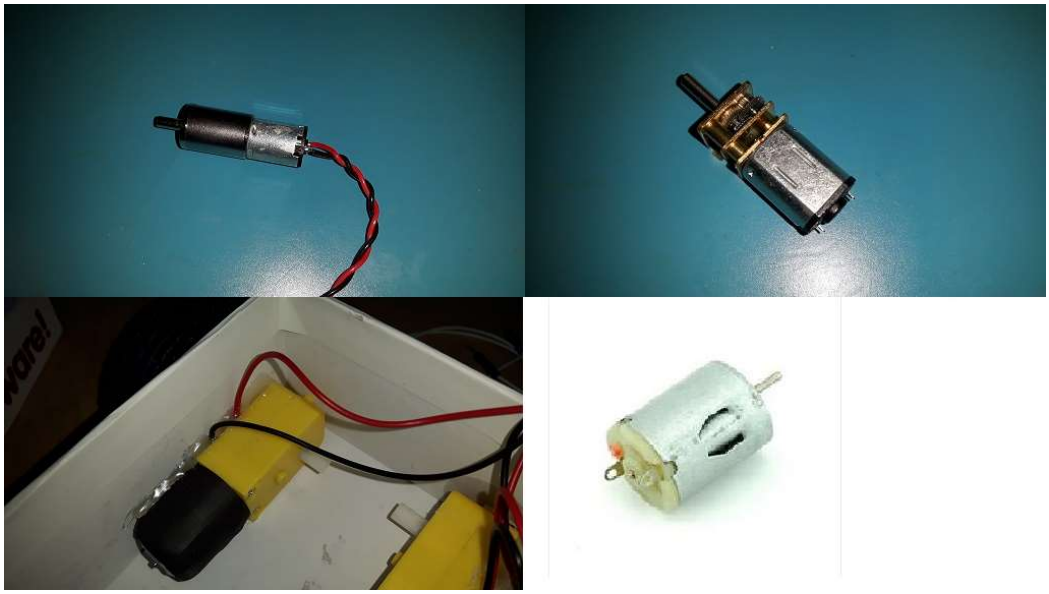
Q) Which way round should I connect the red & black leads from the motors?

A) This really depends on how your motors are wired. The motors from the CamJam EduKit 3 seem to work fine when connected as described in the picture in the earlier section on Connecting The

Motors. If you find the motors run the wrong way, simply swap the red / black wires to the motor that's running the wrong way.

Q) What kind of motors can I use?

A) Small, low-current DC brushed motors. Here are a few examples:



Have fun with your Astro Designs PiZ-Moto+

@AstroDesignsLtd

www.astro-designs.com

**Acknowledgements**

"Raspberry Pi" is a trademark of The Raspberry Pi Foundation.

The "CamJam EduKit #3" was created by the organisers of the Cambridge Raspberry Jam and is distributed by ThePiHut. Find them at: https://thepihut.com/