

Open Source Rover: Electrical Assembly Instructions

Authors: Michael Cox, Eric Junkins, Olivia Lofaro



Jet Propulsion Laboratory
California Institute of Technology

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology. ©2018 California Institute of Technology. Government sponsorship acknowledged

Contents

1 Overview	3
1.1 Tools Necessary	5
1.2 Component Descriptions	6
1.2.1 Battery	6
1.2.2 Voltage Regulators	6
1.2.3 Raspberry Pi 3	6
1.2.4 Printed Circuit Boards	7
1.2.5 LED Matrix	7
1.2.6 Motors	8
1.2.7 Encoders	8
2 Powering Components	9
2.1 Connectors, Switch, and Volt Meter	9
2.2 Volt Meter and Voltage Regulator	12
2.3 Powering the Raspberry Pi	15
2.4 Serial Communication	17
3 Connecting Motors/Encoders	22
3.1 Drive Motors	24
3.2 Corner Motors/Encoders	26
4 Connecting the LED Matrix	31

1 Overview

First, it is important to examine the components we will use in the rover and understand how both data and power travel through the system. Figure 1 shows how each of the electrical components and boards receives power, and Figure 2 shows how data are sent from each of the electrical systems and what communication protocol it uses.

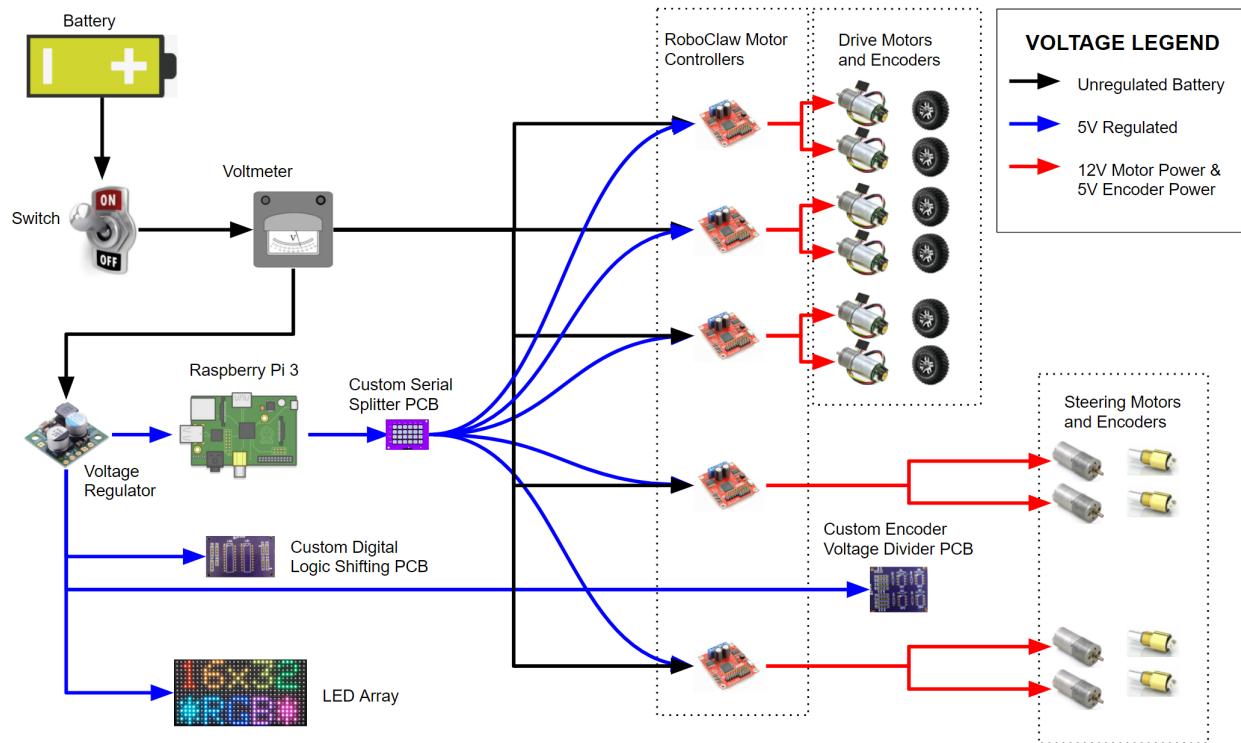


Figure 1: Electrical Power Overview

One important thing to note from the above is that there are multiple different voltages in the system. There is the unregulated voltage coming from the battery, 5V regulated coming from the voltage regulator, and the 12V motor voltage which is sent from the motor controllers to the motors.

1 OVERVIEW

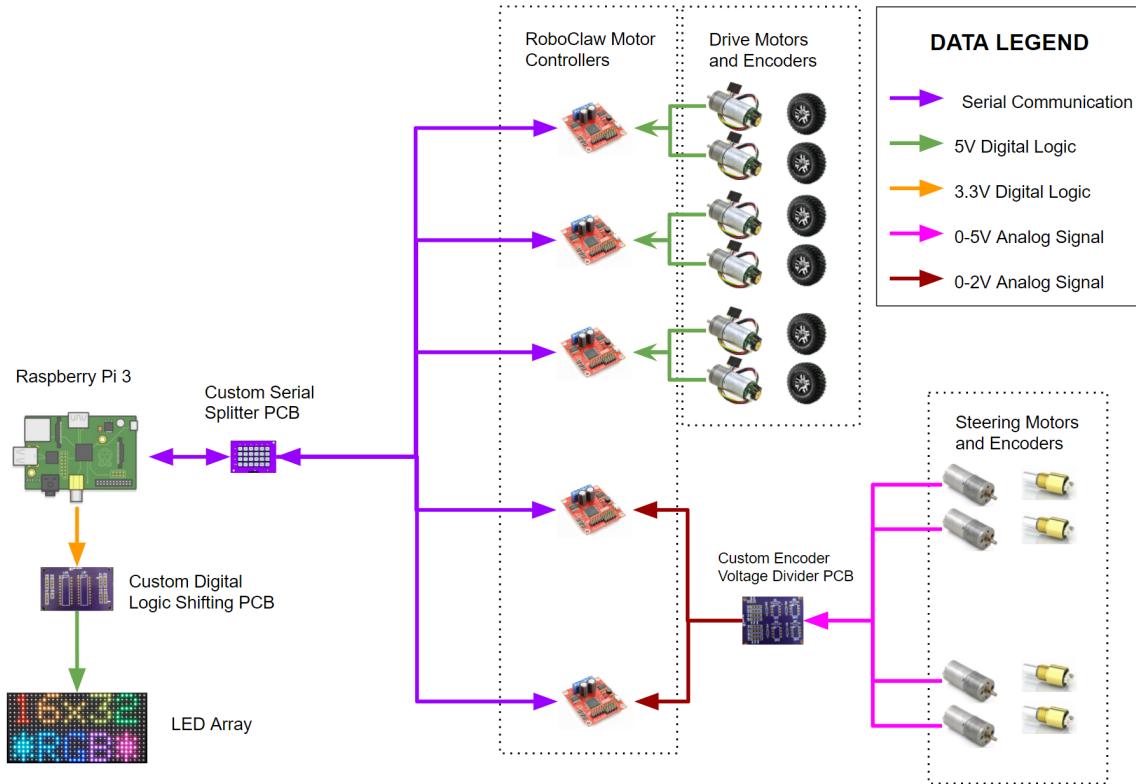


Figure 2: Electrical Data Overview

For the data transmission, we use both 3.3V and 5V digital logic systems, analog signals (of varying max voltages), and serial communication between devices. Information on these protocols can be found at:

- https://en.wikipedia.org/wiki/Serial_communication
- <https://learn.sparkfun.com/tutorials/analog-vs-digital>

Part	Description
Battery	Provides power to the system. Has an unregulated voltage range of around 11.5V - 16.75V depending on its charge level
Switch	Mechanical disconnect of the electrical power to the rest of the robot
Volt Meter	Monitors and displays information such as current draw, voltage level, and power consumption of the robot
Voltage Regulator	Takes in the unregulated battery voltage and outputs a steady regulated voltage of 5 volts for consumption by electronics.
Raspberry Pi 3	Acts as the brains and processing power of the robot
LED Matrix	Gives the robot a display and a personality
Motor Controllers	Sends voltage signals to the motors telling the motors the direction and speed at which to spin. Reads encoder data
Encoders	Monitors the position of the motor to give information about speed and position
Custom PCBs	Single purpose boards custom-made for this project. Each of these will be explained later on in this document.

1.1 Tools Necessary

Tools Necessary

Item	Image	Item	Image
Heat Shrink Gun		Solder	
Digital Multimeter		Solder Wick	
Solder Iron		Power Supply (optional)	

Figure 3

This project assumes you have the tools shown in Figure 3. These will all be used in the construction and testing/debugging of this project. Most of these can be purchased inexpensively through DigiKey, Amazon, or other retailers.

1.2 Component Descriptions

The following are brief descriptions and basic notes about some of the components we use in this project. For more detailed information on each component, consult the data sheet for that component.

1.2.1 Battery

The battery is used to power all the electrical components and the motors in our rover. The biggest driving factor for battery selection was safety of the battery chemistry while still being able to drive the current and voltage necessary for operation of all the electronics (we'll go into these requirements more later). The recommended battery was chosen because of its Poly Switch and PCB protection. These limit the amount of current that can be pulled from the battery, as well as give protection against things like electrical shorts and overcharging/fully discharging the battery.

1.2.2 Voltage Regulators

As our battery outputs unregulated voltage based on its charge level, voltage regulators are added to give steady DC power to the electronic components. There is a 5V regulator that runs the raspberry pi and the LED matrix, and a 12V regulator that supplies power to the motors which run on 12 volts.

1.2.3 Raspberry Pi 3

We selected the Raspberry Pi 3 Model B as the "brain" of the rover. We chose the Pi because it allows users at many levels of programming knowledge to gain familiarity in basic

programming concepts without a complicated learning curve. It is also widely used in the maker community, so modifications and additions on the base rover should be easy. The Pi 3 also comes with WiFi and bluetooth modules built into the board and we therefore don't need additional dongles to communicate over WiFi or bluetooth. This retains all 4 USB ports for addons and modifications to the project.

1.2.4 Printed Circuit Boards

In order to simplify the electronics of the rover, we designed a few single purpose Printed Circuit Boards (PCBs). In total, there are three boards:

Board	Description/Purpose
Voltage Divider	Distributes the +5V and GND to power the absolute encoders. Also takes the 0-5V signal from the encoders and transforms it to a 0-2V signal that the controllers expect
Logic Shifter	Takes the 3.3V digital signals from the raspberry pi and converts them to a 5V digital signal output
Serial Splitter	Takes the TX/RX serial read/write lines and the +5V and GND power pins from the Raspberry Pi and distributes them to each motor controller.

1.2.5 LED Matrix

The LED Matrix is a 16x32 pixel display that is controlled through the Raspberry Pi¹. In order to integrate this display into our rover, we followed along with an Adafruit project (but added our own custom PCB which isn't in this tutorial). This tutorial can be found at: <https://learn.adafruit.com/connecting-a-16x32-rgb-led-matrix-panel-to-a-raspberry-pi/overview>

¹This display uses a lot of the Pi's GPIO pins. If you need many GPIO pins for your own additions, you may want to swap this display out for another option.

1.2.6 Motors

Motors cause the actual movement of the robot. There are many types of motors; in this project we will be used brushed DC motors. More information on different types of motors and a link about how brushed DC motors work can be found at the following links:

- https://en.wikipedia.org/wiki/Brushed_DC_electric_motor
- https://en.wikipedia.org/wiki/Electric_motor

1.2.7 Encoders

Encoders give us readings on the position and speed of motors as they spin. There are two types of encoders in this project: absolute and quadrature. An absolute encoder is used to get the position of the motor on a fixed scale such that you always know the position of the motor even after power cycling the system. This type of encoder will usually have one output, which will either be a PWM digital signal or an analog signal. Each of these signals will have a specific reading corresponding to where it is on the fixed position scale. These are typically more expensive than their quadrature encoder counterparts, which will sense position only relative to the last time the power was cycled or the signal was reset. Quadrature encoders use two digital signals and look at the different between the signals to sense movement. More information about quadrature encoders can be found at:

- http://www.dynapar.com/Technology/Encoder_Basics/Quadrature_Encoder/

In this project, we use absolute encoders on the corner wheels to get absolute measurements on where the wheels are turned. We use quadrature encoders at each of the driving wheels to get speed measurements.

2 Powering Components

Here we will be setting up the connections to the electrical components. It is recommended that you try hooking up each of these components outside of the robot body first to test in a simpler environment and to make sure you fully understand how all the signals are passed around. This guide also assumes that you have already followed the PCB Testing guide and that you have fully tested and functioning PCBs.

When connecting any of the power lines it is important to consider how much current will be running through that wire, and select the appropriate wire gauge to handle that current. For more information on wire gauge selection you can visit:
https://www.bluesea.com/support/articles/Circuit_Protection/1437

During this project you will need to cut, strip, and solder wires to extend and split electrical connections. It is important to understand how to do this safely so that you are protecting your system from electrical shorts. Below are a few quick links on these skills, however it is crucial that you fully understand these processes before proceeding.

- <https://learn.sparkfun.com/tutorials/working-with-wire/how-to-strip-a-wire>
- <https://www.wikihow.com/Splice-Wire>
- <https://www.youtube.com/watch?v=Y8wjv6lj5KU>

****Note** Please read through this entire section and understand it fully before starting to connect devices and electronics together.**

2.1 Connectors, Switch, and Volt Meter

We'll start with the battery and connecting it through the switch and volt meter. Figure 4 shows the section of the power diagram we'll be working on, with components and connections that we make in this section marked in **RED**.

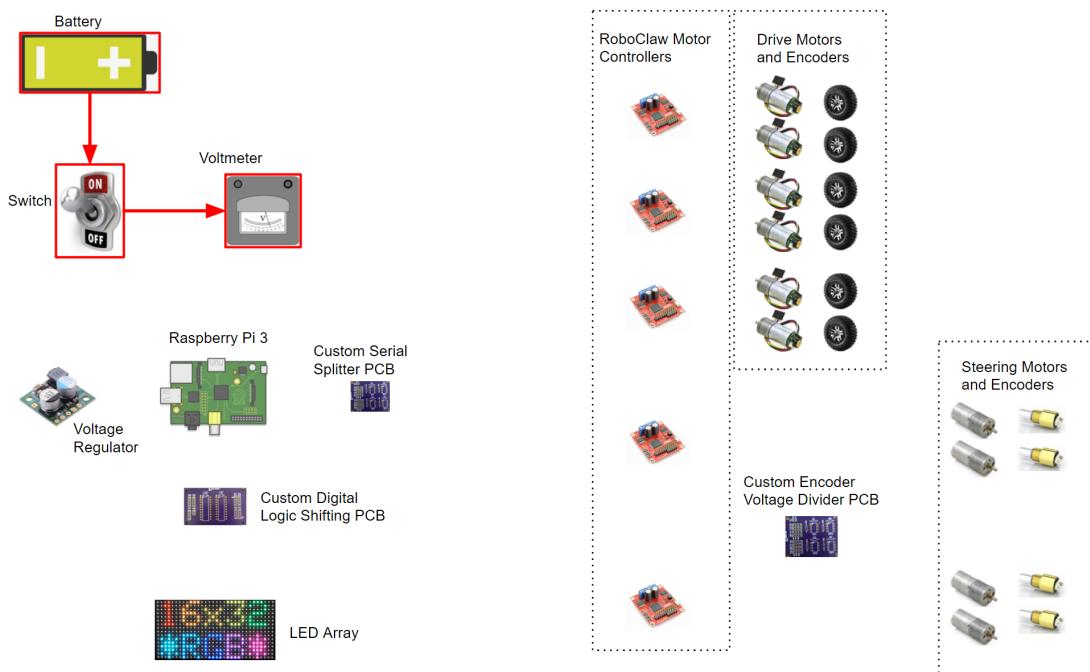


Figure 4: Battery section Overview

There are a couple of **very important** things to be aware of as you are making these connections, as with any project where you are working with batteries or electrical current:

THE BATTERY IS LIVE AT ALL TIMES. This means that at any point, if the two terminals of the battery come in contact they will cause an electrical short. This may trigger the protection circuit in the battery and prevent extensive damage, but we do not want to rely on that for safety. **Make sure to keep the terminals of the +V and GND separated at all times with no possibility of contact.**

The connectors will only fit together one way. That means you have to make sure your connections are correct the first time when soldering them in: make sure to verify this before connecting anything to the battery.

The switch we use is a Single Pole - Single Throw switch. This means that either the two terminals are connected together in one switch position or there is a physical disconnect when the switch is in the other position. Make sure to test the switch and know which are its ON and OFF positions previous to connecting it to anything.

It is extremely important to plug the battery in the correct direction into the volt meter as well, as plugging it in backwards (even for an instant) could damage the volt meter.

Using the Tamiya connectors, connect the battery to the switch and then the volt meter according to Figure 5. The Tamiya connectors allow you to disconnect the battery from the rest of the system so you can recharge the battery.

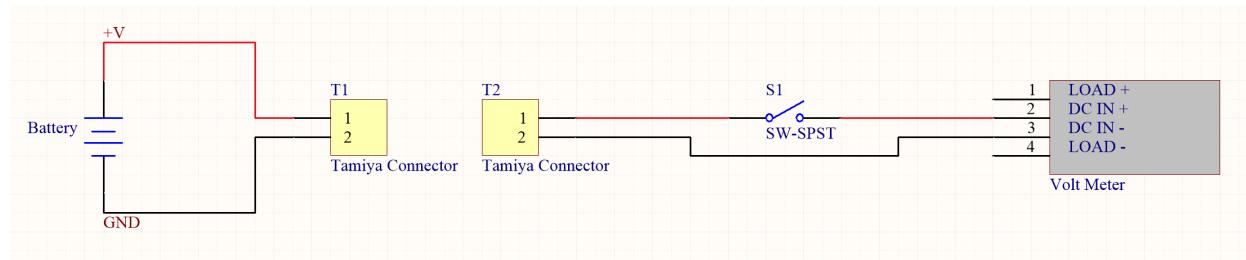


Figure 5: Battery connection through switch and volt meter

The battery will live in the body of the rover above the electronics plate, with the switch mounted to any of your choice of the 0.5in holes in the body, and the volt meter has a press fit cutout in the back panel. Figure 6 shows an example of how we installed the switch, and the volt meter/ Tamiya connectors. Note how the connectors are available readily (in our case, outside the rover). This allows us to unplug the battery and charge it without removing any of the rover's components.

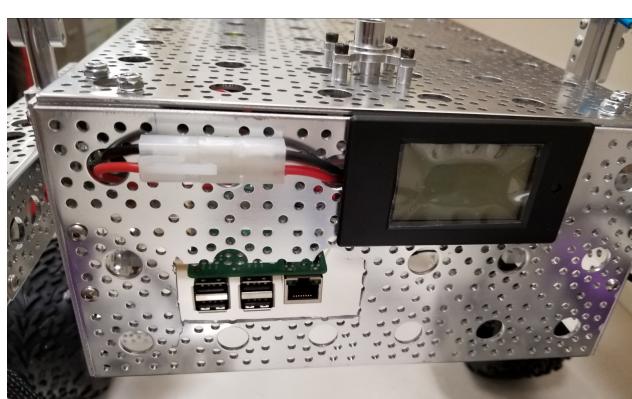
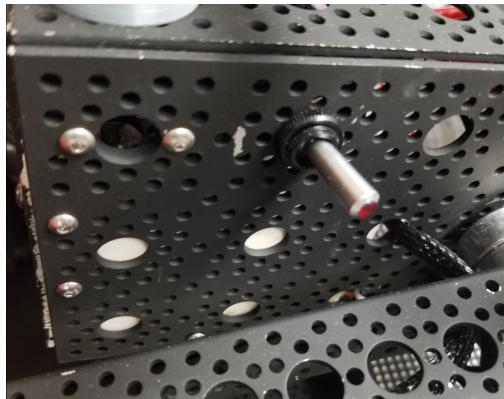


Figure 6: Installing the switch and voltmeter

2.2 Volt Meter and Voltage Regulator

Next, we connect the output side of the volt meter to the 5V voltage regulator and the motor controllers. Figure 7 shows the section of the power diagram we'll be working on, with components and connections that we make in this section marked in **RED**.

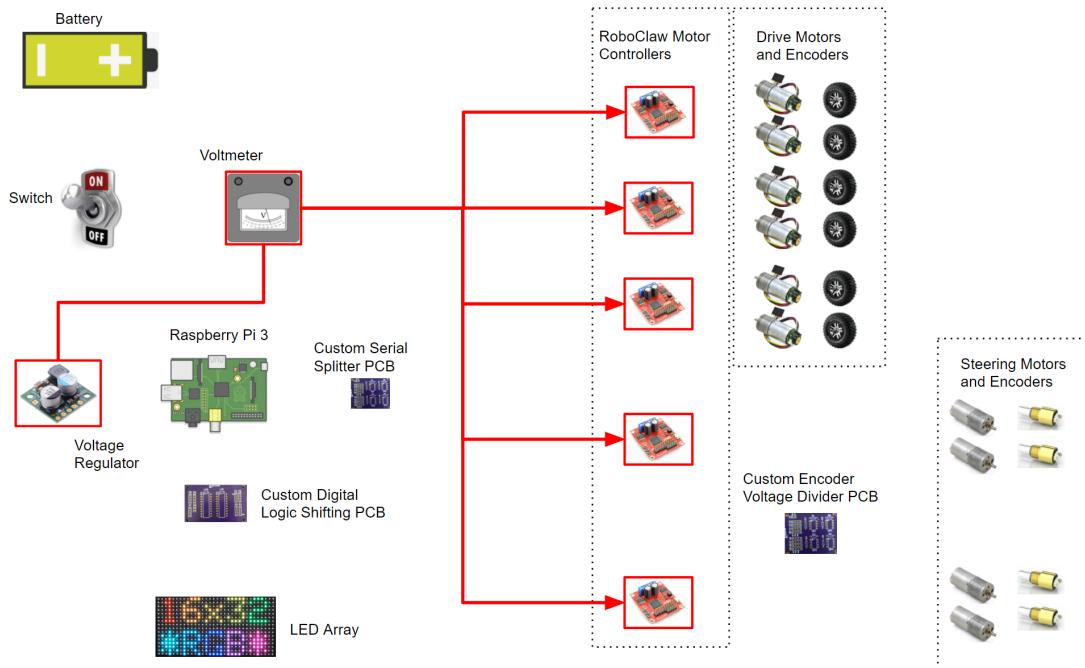


Figure 7: Regulator and Motor Controllers Overview

The "LOAD" terminals of the volt meter are the electrical load of the robot, which correspond to all the electronics running in the robot. Figure 8 shows the pin-out of how to connect the volt meter to the 5V regulator and the motor controllers.

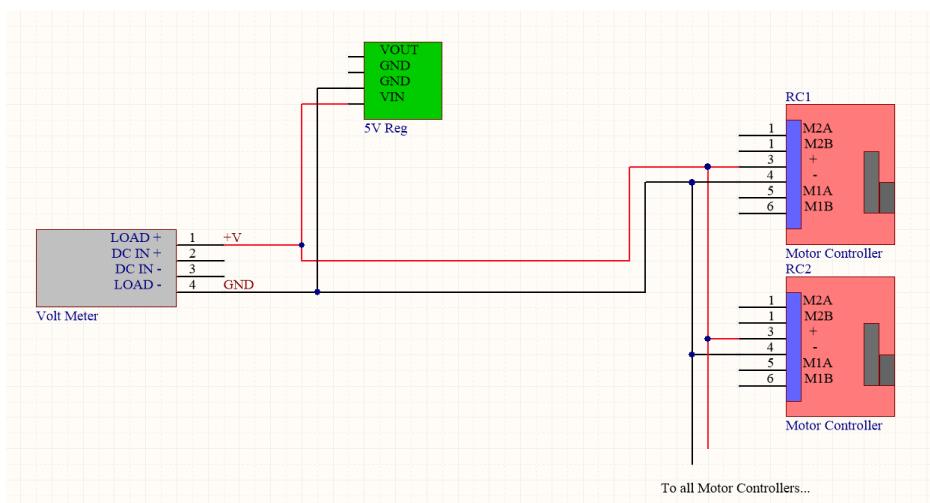


Figure 8: Volt Meter to Electronics connections

The LOAD+ pin should connect to VIN on the 5V Regulator, and + on each of the RoboClaw Motor Controllers, and the LOAD- pin should connect to GND on the Regulator and - on the RoboClaws. After the Volt Meter, you should break out the connections to the regulator and RoboClaws into two separate wires, each with sufficient wire gauge. The 5V regulator will be fine with 20AWG wire, but the power lines running to every motor controller should be 16AWG.



Pololu 5A Step-Down Voltage Regulator D24V50Fx, top view with labeled pinout.

Figure 9: Voltage Regulator Diagram

Figure 10 shows an example of one way we did the wire routing for power to the motor controllers. We can see that the power was split off before reaching any of the motor controllers, and instead each of them individually got a power and ground wire.

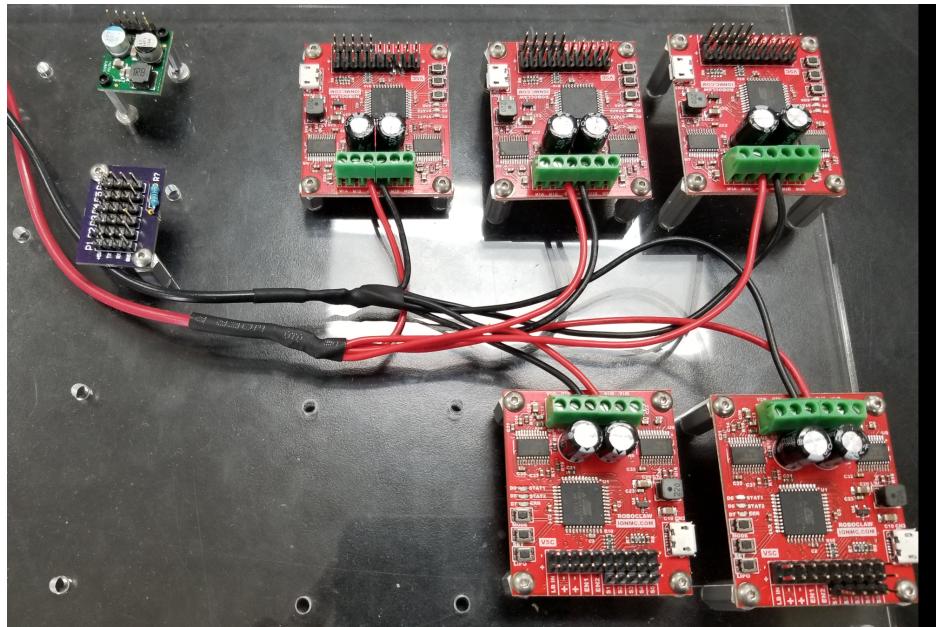


Figure 10: Powering the RoboClaws

Before connecting to the rest of the electronics, we will want to test the voltage coming out of the 5V regulator. To do this, use a Multi-meter to test the voltage across the VOUT and GND pins. You should see very close to 5V. **Be very careful not to short across the GND and VOUT pins while testing** as this could short the regulator and cause it to break.

2.3 Powering the Raspberry Pi

Next, we need to bring power to the Raspberry Pi. It requires a regulated 5V DC power source, so it will be powered from the 5V regulator. Figure 13 shows the section of the power diagram we'll be working on, with components and connections that we make in this section marked in **RED**.

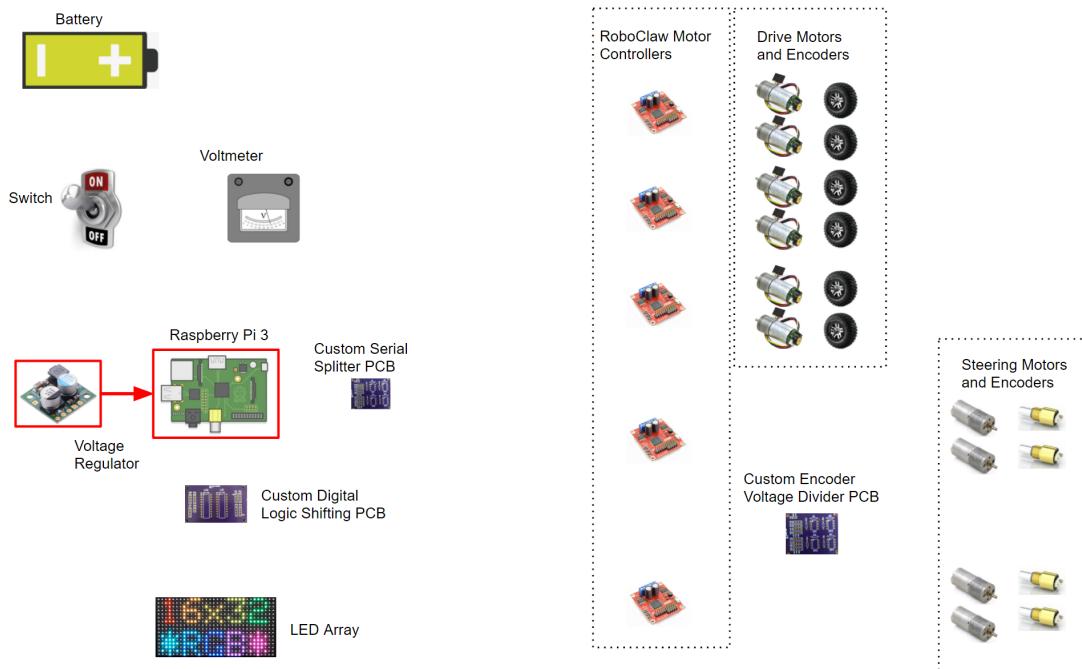


Figure 11: Powering the Raspberry Pi

Note: If you plan on using the LED matrix / face, you should split 5V power off of the 5V regulator before the Raspberry Pi, which should be done now. As the LED Matrix takes up a large number of GPIO pins which some might want for other devices, it's not a required part of the project and thus there is a separate document describing how to put it together.

To power the Raspberry Pi, we soldered the 5V and GND from the 5V regulator directly to the power pads on the bottom of the Raspberry Pi. The positive voltage goes to pad PP2 and negative/ground goes to pad PP5 in Figure 12. For more information and a video on doing this you can follow this link and go to around 3:30

- <https://www.youtube.com/watch?v=9LAtu39xpR8>.

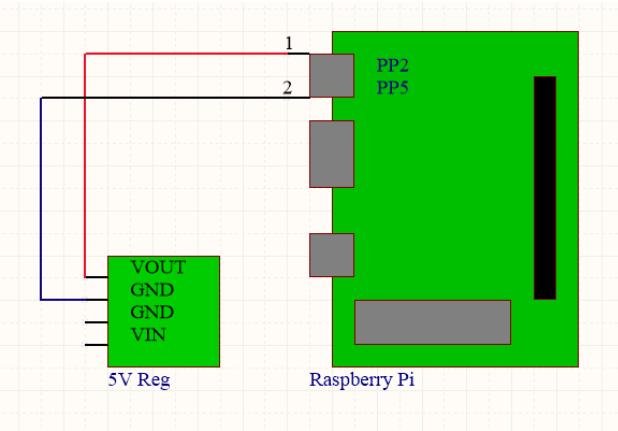
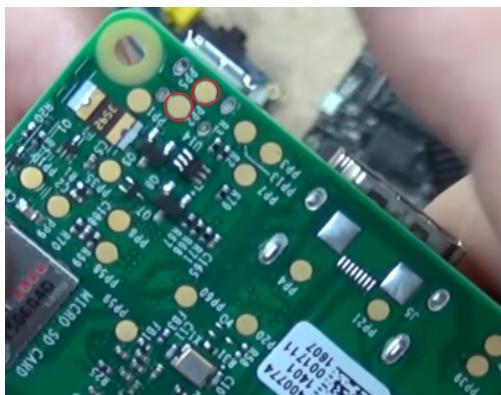


Figure 12: Raspberry Pi 3 soldering pads

2.4 Serial Communication

Connecting the Raspberry Pi to Serial Splitter Board: In order for the Raspberry Pi to talk to the motor controllers we will be using Serial communication. Figure 13 shows the section of the power diagram we'll be working on, with the power connections shown in **RED** and data lines in **BLUE**.

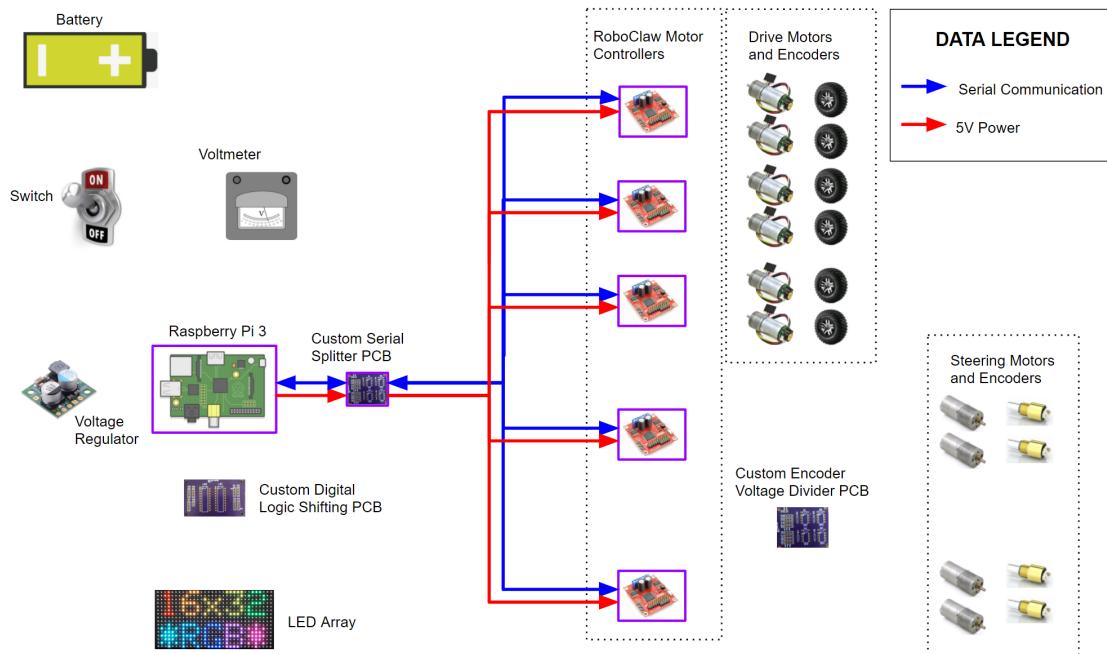


Figure 13: Serial Splitter Overview

Figure 14 shows the schematic and pin-out of how to connect the Pi, Serial Splitter board, and Motor Controllers. From the Raspberry Pi we use the UART TX and RX pins, which correspond to pins #8 and #10, or GPIO14 and GPIO15. In addition we will be passing 5V to the motor controllers for the logic battery controller which is used by the encoders and digital logic. For this use the 5V PWR rail which is pin #2 or #4 and any of the GND pins (we recommend pin #6). Raspberry Pi GPIO diagrams can be found on the following page in Figure 15.

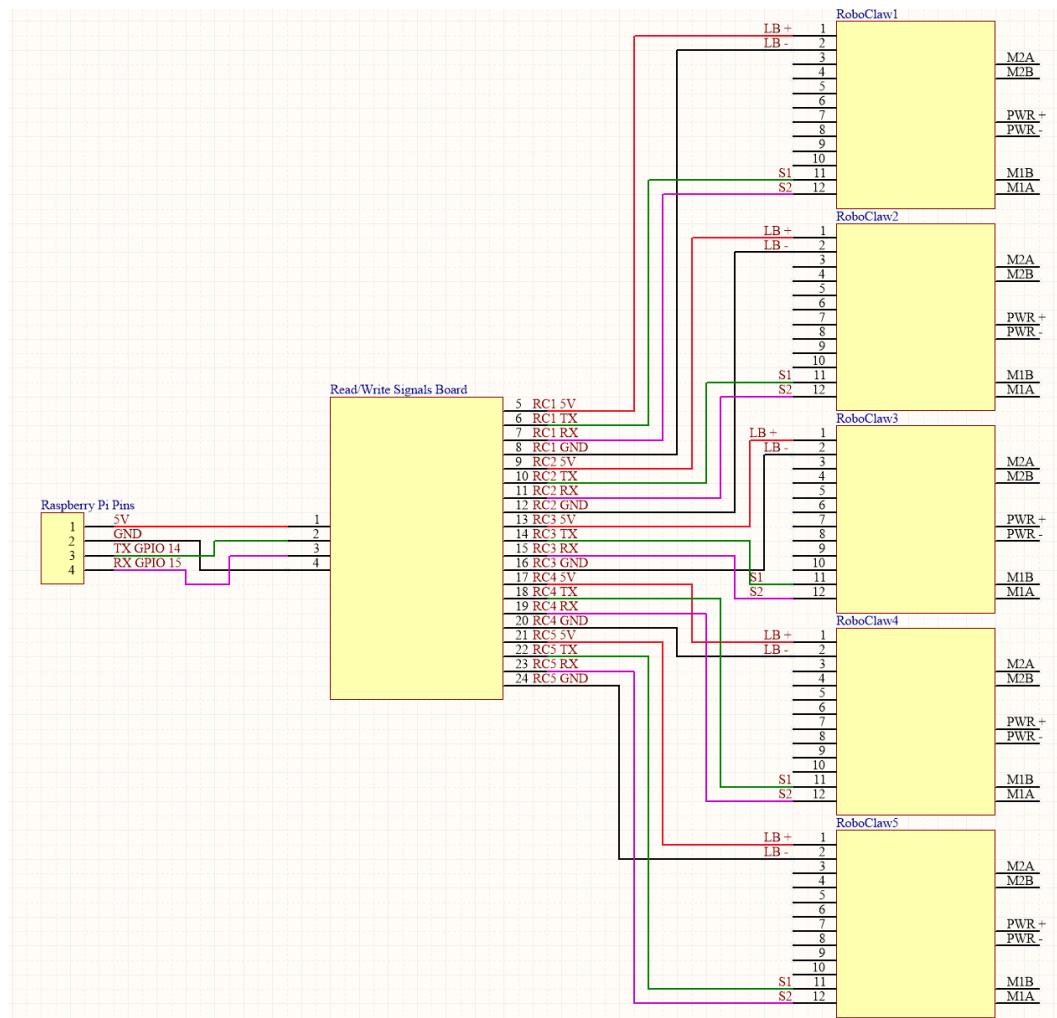


Figure 14: RPi connections to RoboClaws

Below shows the description of what each pin connects from and to on the Serial Splitter board. Use this in combination with Figure 14 and Figure 16 to make the serial connections

to the RoboClaw Motor Controllers.

Pin	Description
1	+5V from Raspberry Pi 5V DC pin #2 or #4
2	Serial Write line, TX from Raspberry Pi GPIO 14
3	Serial Read line, RX from Raspberry Pi GPIO 15
4	GND from Raspberry Pi GND pin #6
5,9,13,17,21	5V to RoboClaws LB + pin
6,10,14,18,22	Serial Write to RoboClaws S1 pin
7,11,15,19,23	Serial Read to RoboClaws S2 pin
8,12,16,20,24	GND to RoboClaws LB - pin

Please note that there is a difference between the pin numbers for the physical pins on the Raspberry Pi 3 and their corresponding GPIO pin numbers (for instance, physical pin #8 is GPIO 14). Consult Raspberry Pi forums and documentation if you need additional information.

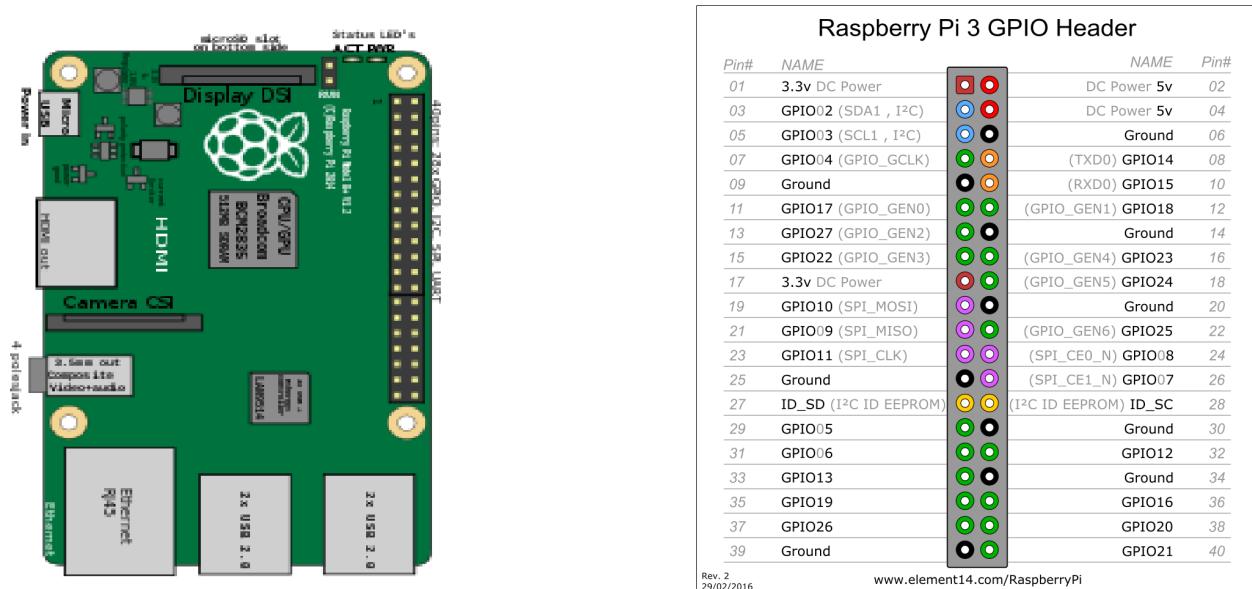


Figure 15: Raspberry Pi 3 pinout

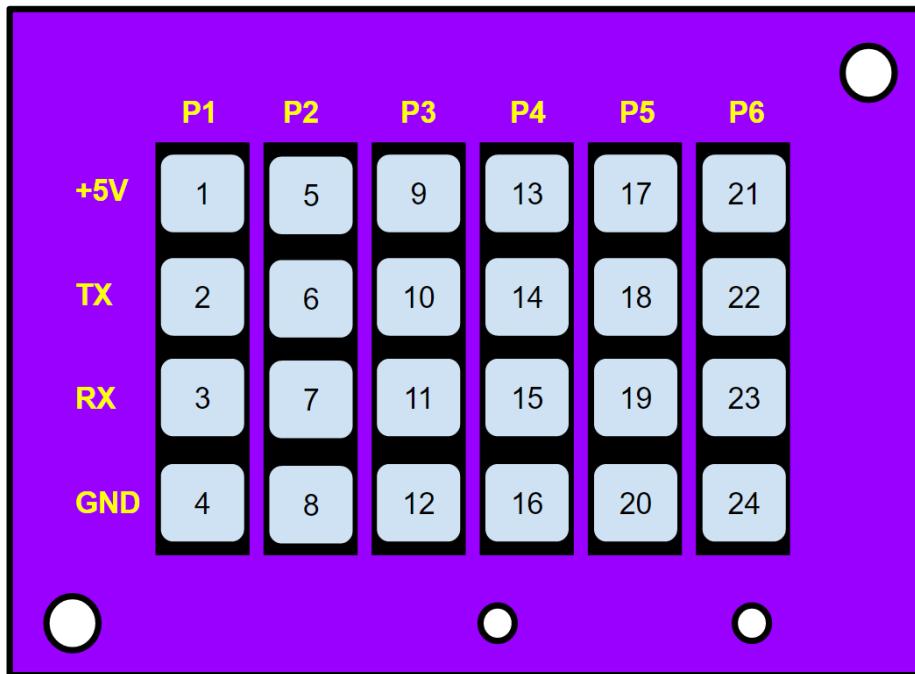
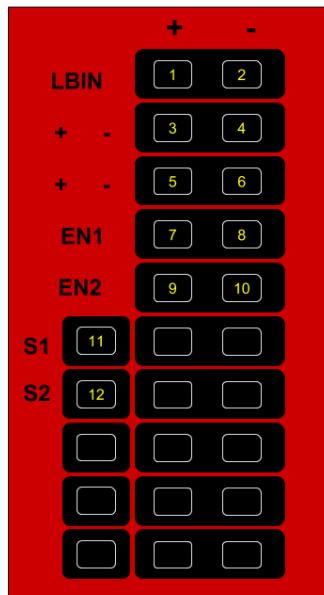


Figure 16: Read/Write Board



1. Logic Battery Power (+5V)
2. Logic Battery GND
3. Encoder 1 Power (+5V)
4. Encoder 1 GND
5. Encoder 2 Power (+5V)
6. Encoder 2 GND
7. Encoder 1 Signal (PWM signal A if quadrature encoder, 0-2V if absolute)
8. Encoder 1 Signal (PWM signal B if quadrature encoder, open if absolute)
9. Encoder 2 Signal (PWM signal A if quadrature encoder, 0-2V if absolute)
10. Encoder 2 Signal (PWM signal B if quadrature encoder, open if absolute)
11. Serial Read (TX line from Raspberry Pi)
12. Serial Write (RX line from Raspberry Pi)

Figure 17: RoboClaw pinout

Next, follow Figure 14 and the pinout Figures 16 and 17 to connect the signals to

each individual RoboClaw². For more information on the pins for the RoboClaws and their purpose refer to the RoboClaw Manual and data sheet.

Examples of the wiring and routing can be seen in Figure 18 and Figure 19

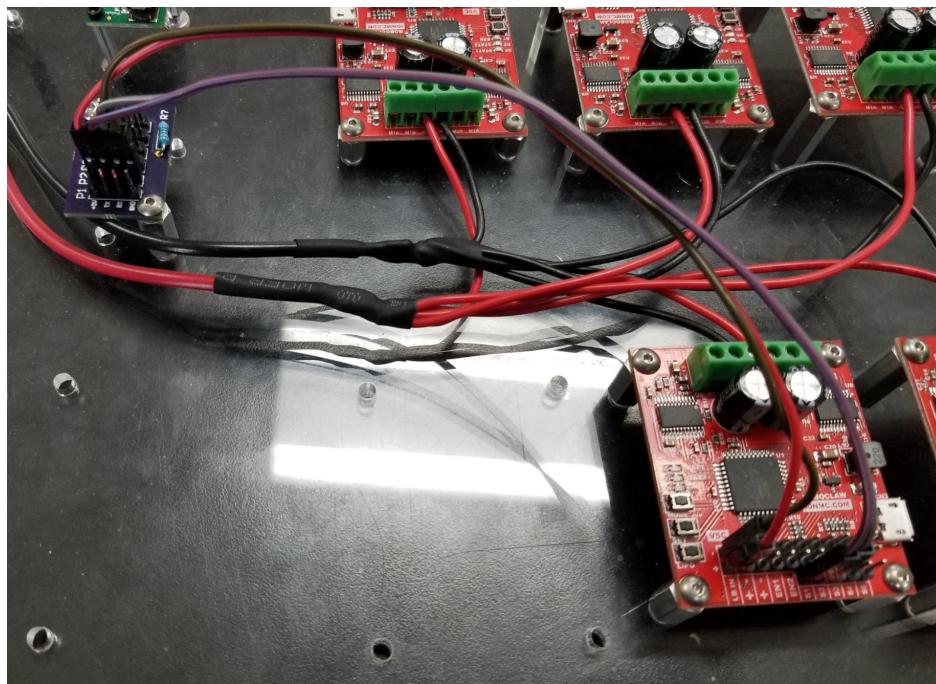


Figure 18: Serial Splitter to RoboClaws (shows 1 of 5 RoboClaws finished)

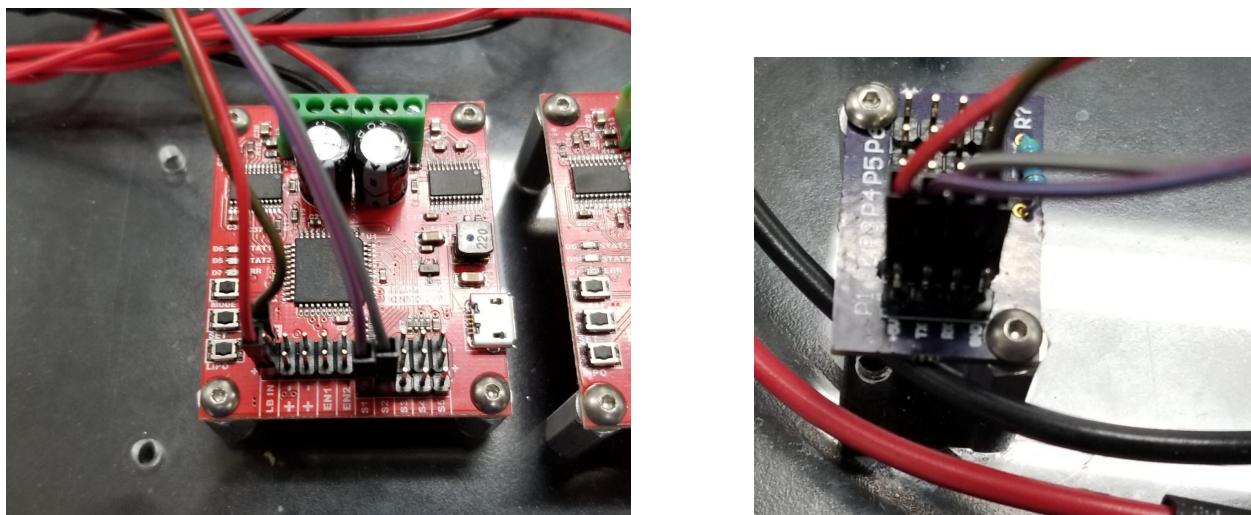


Figure 19: Serial signal routing

²The pin number in Figure 14 for the Serial Splitter Board corresponds to the pin numbers in Figure 16

3 Connecting Motors/Encoders

Next we can connect the motors and encoders to the RoboClaw motor controllers. For each of the 10 motors you will need to connect motor power as well as the encoders. This means that there will be close to 60 wires traveling between the motor controllers and the motors/encoders, so it is very important to try to keep your wires organized and bundled together well. Below is some information about strain relief and using wire braid in your electronics systems. See these links and Figures 20 and 21 for examples.

- <https://www.techopedia.com/definition/2301/strain-relief>
- <https://www.youtube.com/watch?v=FeCs98TSsYQ>

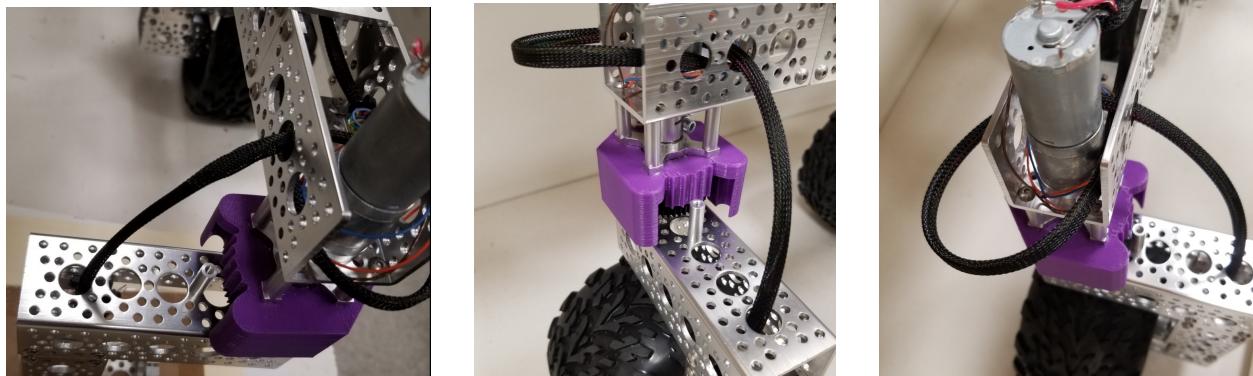


Figure 20

An example of how we routed our wiring is shown by Figure 20. It is important to give strain relief in the cable as you are routing it through, as well as to give extra wiring for the locations that can move. In addition, Figure 21 shows how we routed the wires through the rocker-bogie, and then from the rocker-bogie into the main body. Try to pick locations that reduce strain on the wires as the robot legs move when you are routing wires into the body.

3 CONNECTING MOTORS/ENCODERS



Figure 21: Serial signal routing

Section	Wires Needed	Length [inches]
A	2x 24AWG & 4x 30AWG	35
B	2x 24AWG & 4x 30AWG	28
C	2x 24AWG & 4x 30AWG	48
D	2x 24AWG & 3x 30AWG	20
E	2x 24AWG & 3x 30AWG	32

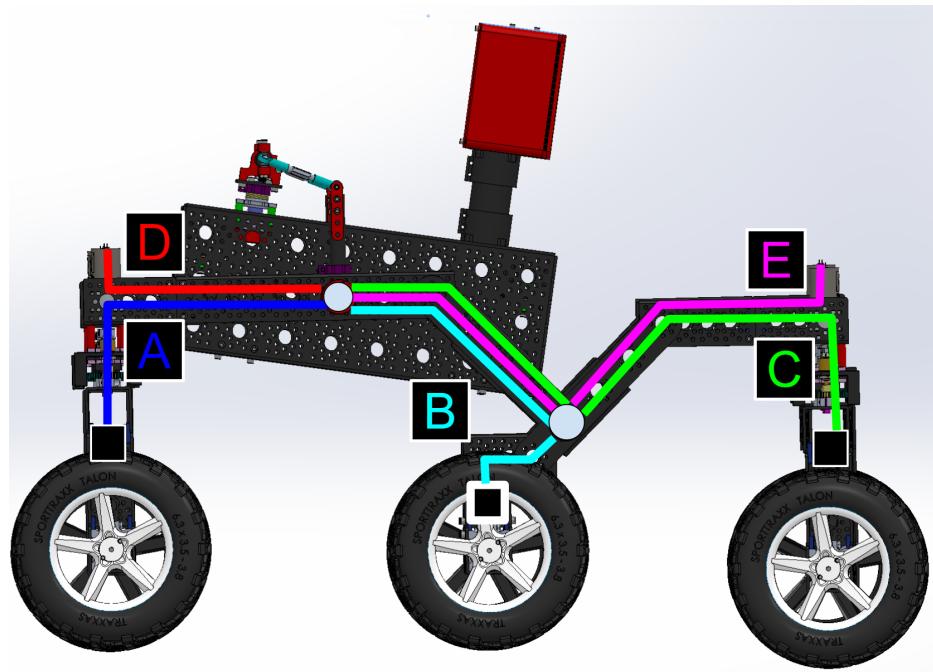


Figure 22: Wire lengths

3.1 Drive Motors

We will begin with the drive motors (the 6 motors that are directly connected to each drive wheel). Figure 23 shows the section of the power diagram we'll be working on, with lines with both encoder signals and motor power in **PURPLE**.

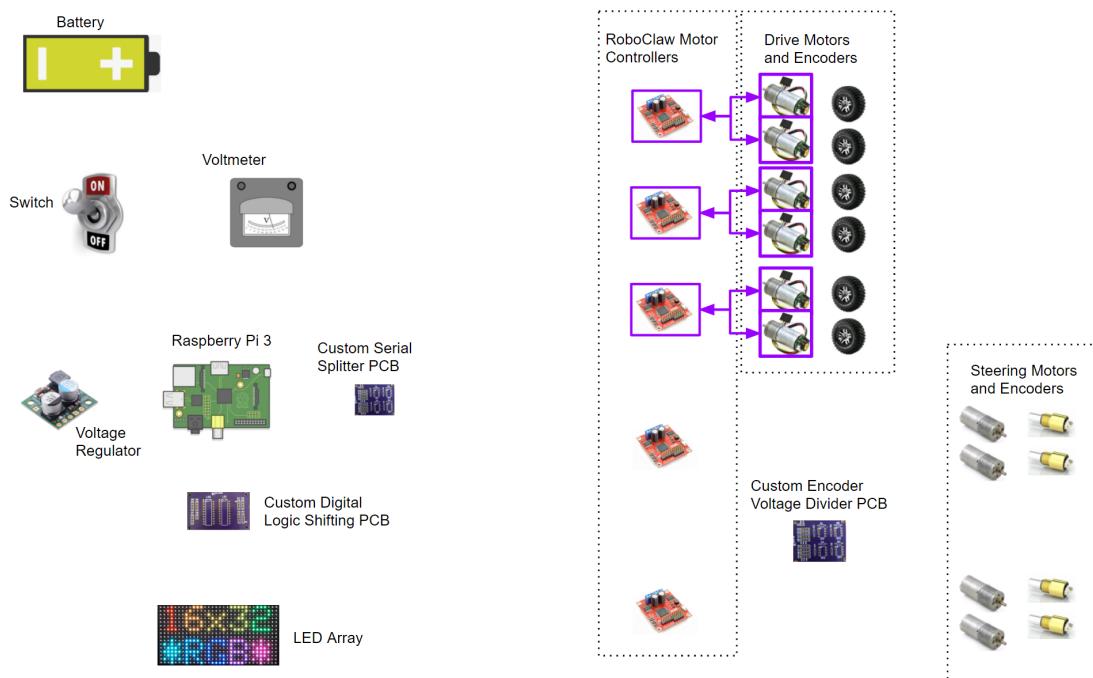


Figure 23: Motor and Encoder Overview

Make sure to look at which RoboClaw corresponds to which motors it is controlling. It is important that this ordering is preserved. Assign and mark each of your RoboClaws a number 1-5 and then follow Figure 24 when connecting each to its motors.

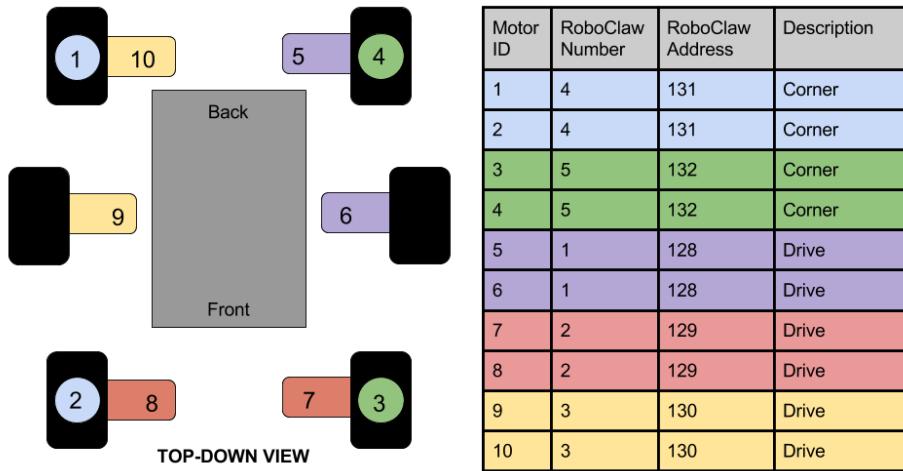


Figure 24: RoboClaw - Motor Assignments

Figure 25 shows what each wire from the motor represents and Figure 26 shows which of those wires should connect to which pin on the RoboClaw motor controllers. Note that the white wire is represented by pink in Figure 26.³

Color	Function
Red	motor power (connects to one motor terminal)
Black	motor power (connects to the other motor terminal)
Green	encoder GND
Blue	encoder Vcc (3.5 – 20 V)
Yellow	encoder A output
White	encoder B output

Figure 25: Motor pinout

³Figure 26 shows only a single motor controller and 2 motors, broken into 2 pictures to reduce clutter on the image and increase wire readability. You will need to replicate Figure 26 for all 5 RoboClaws and all 10 motors.

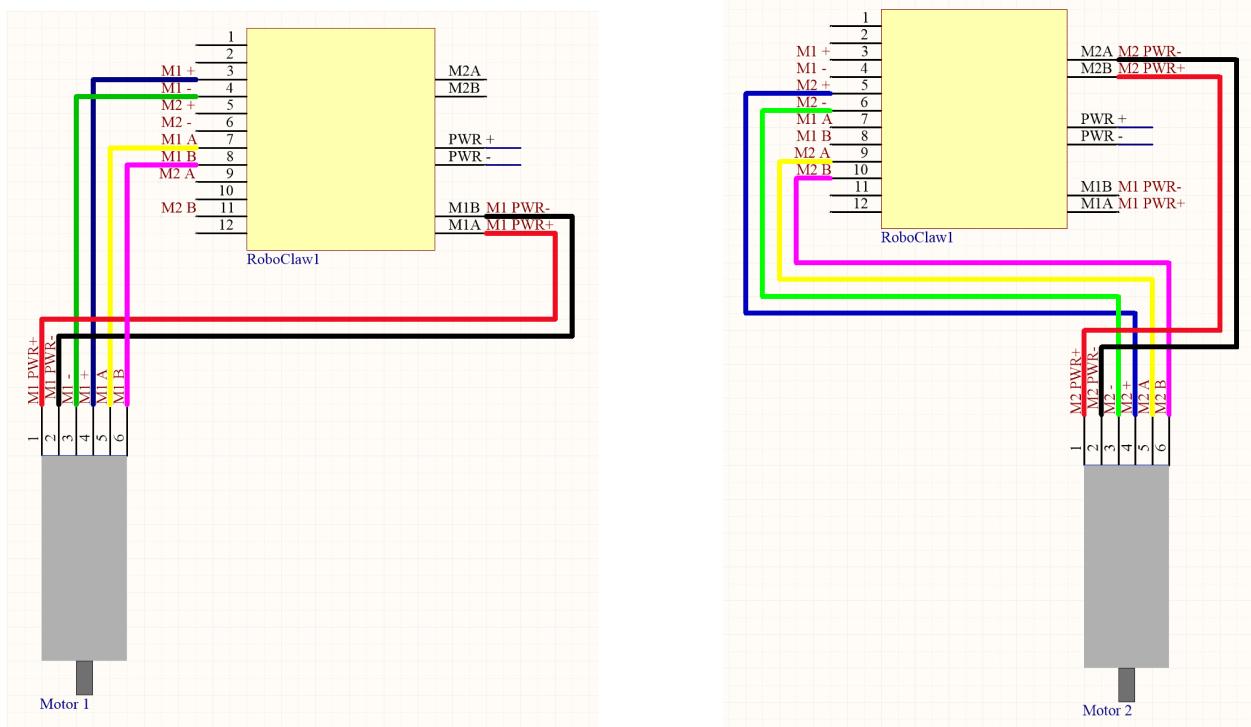


Figure 26: Connecting RoboClaws to Drive Motors

3.2 Corner Motors/Encoders

The big difference between the corner motors and drive motors is the use of absolute encoders. The absolute encoders are separate from the motor and so we have to do a little bit of work to get them to interface with the motor controllers. Figure 27 shows the section of the power diagram we'll be working on, with lines that power the motors in **RED** and encoder signals in **BLUE**.

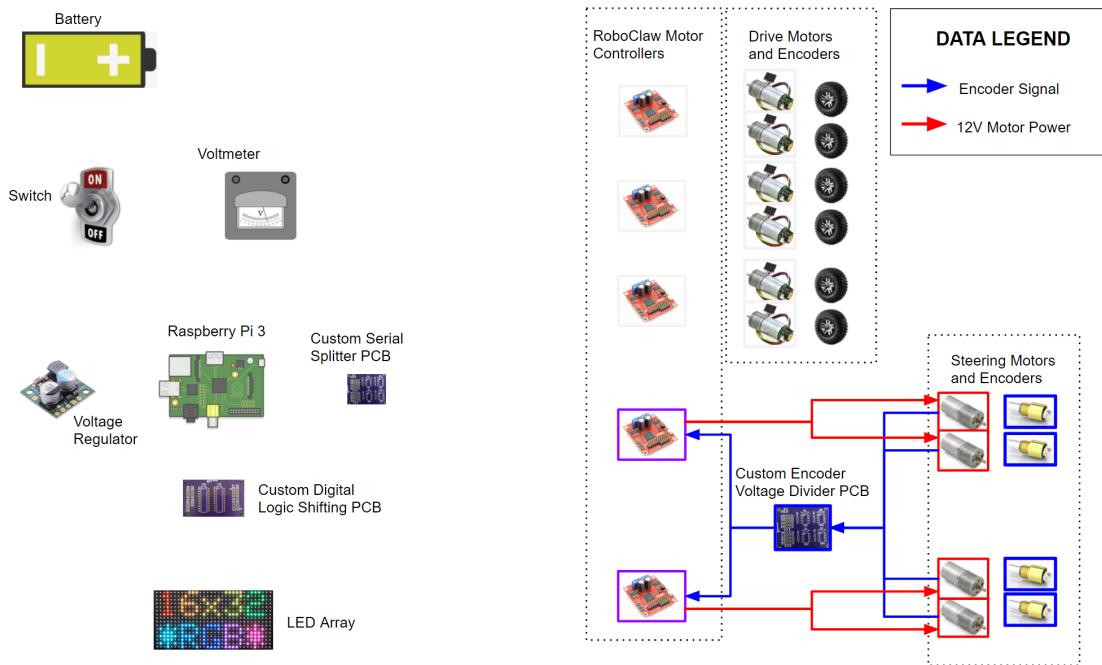


Figure 27: Corner Motor and Encoder Overview

The corner motors only have the two connections to the physical motor, the (+) and (-) going to the top of the motor. However, you must also hook up the absolute encoder. When plugged into the encoder, the 3-pin Micro connector will have 3 wires coming out of it, shown in Figure 28.

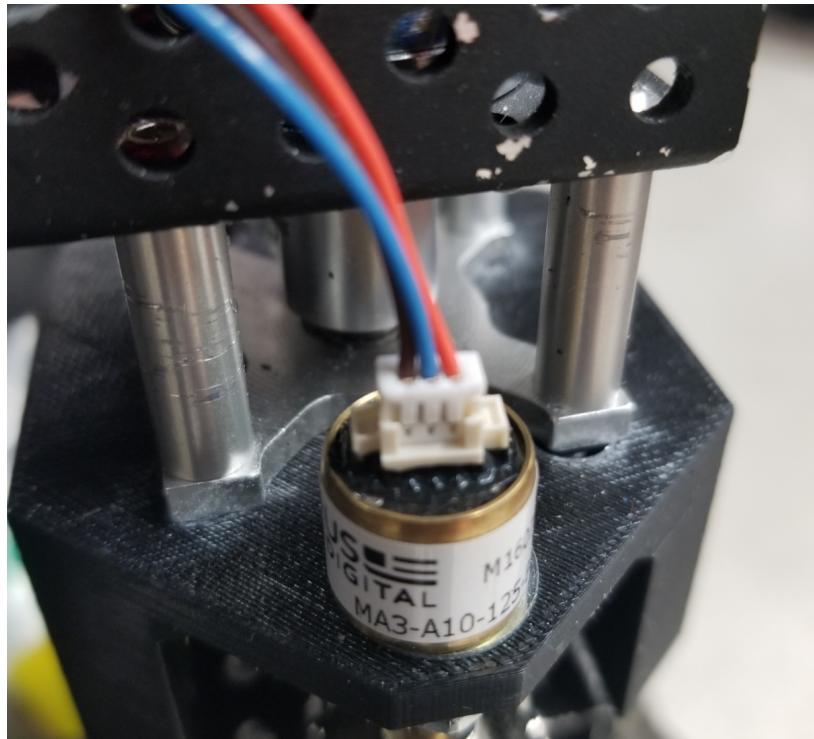


Figure 28: 3-pin Micro connector

Pin	Description	Color
1	+5V DC Power	Orange
2	0-5V Analog	Blue
3	Ground	Brown

The absolute encoder will be hooked up similarly to the encoders for the drive motor on the RoboClaw side, with the exception of the OUTPUT B signal which doesn't exist for absolute encoder types: it just needs to send the 10-bit analog signal to the RoboClaw. However, there is an issue where the encoder outputs signals on a 0-5V scale, but the RoboClaw expects an input of 0-2V for an absolute encoder value. To address this issue, we will use the Encoder Voltage Divider PCB, the pinout of which can be seen in Figure 29. Connect the corner motors, the Encoder Voltage Divider PCB, and the RoboClaws according to Figure 30.



Figure 29: Encoder Voltage Divider board pinout

Pin	Description
1,4,7,10	+5V Input to Encoders
2,5,8,11	0-5V Signal from Encoders based on position
3,6,9,12	GND for Encoders
13,16,19,22	+5V Power from each respective RoboClaw
14,17,20,23	0-2V Signal Output to RoboClaws, after voltage division on board
15,18,21,24	GND from each respective RoboClaw
25	+5V from Raspberry Pi power rail
26	GND from Raspberry Pi pin

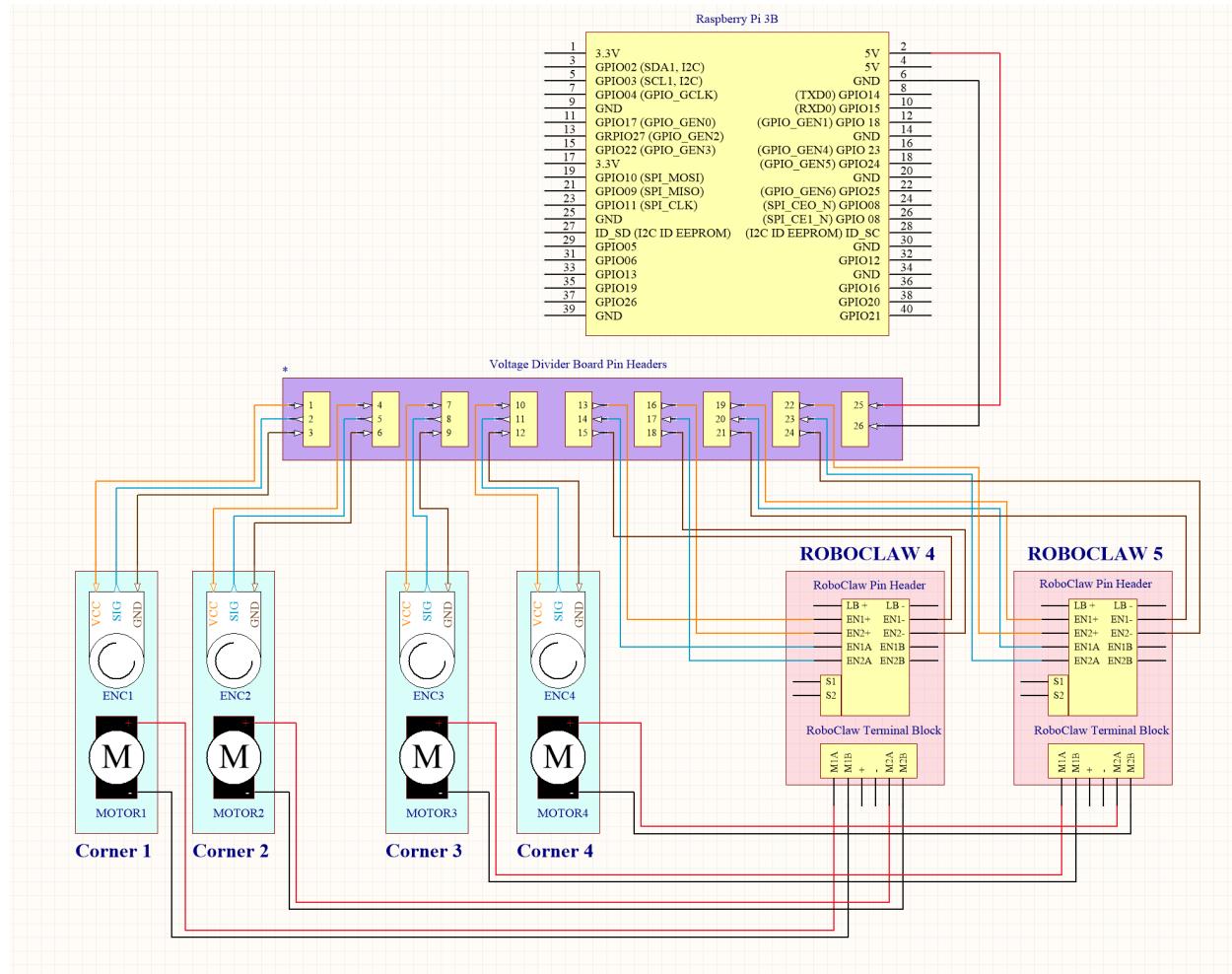


Figure 30: Connecting RoboClaws to Corner Motors

Once all 10 of the motors are connected and in the system, we are ready to move onto the motor tuning and calibration process which is documented in the Calibration.pdf document. Make sure to double check all your wire connections and that they are into the correct pins before moving on. We recommend calibrating your motors and having the robot functioning before moving on to the LED Matrix due to simplicity and ease of debugging, though either order would be fine.

4 Connecting the LED Matrix

We followed along with a project on Adafruit to hook up the Raspberry Pi to the LED matrix, with the addition of the logic shifter board between the Pi and LED matrix. That project can be found at:

- <https://learn.adafruit.com/connecting-a-16x32-rgb-led-matrix-panel-to-a-raspberry-pi/overview>

Figure 31 shows an overview of how the logic shifters are used between the signals from the Raspberry Pi and the LED Matrix. This diagram does not include the actual pin connectors to the Logic Shifter board; it is just to demonstrate the signal path.

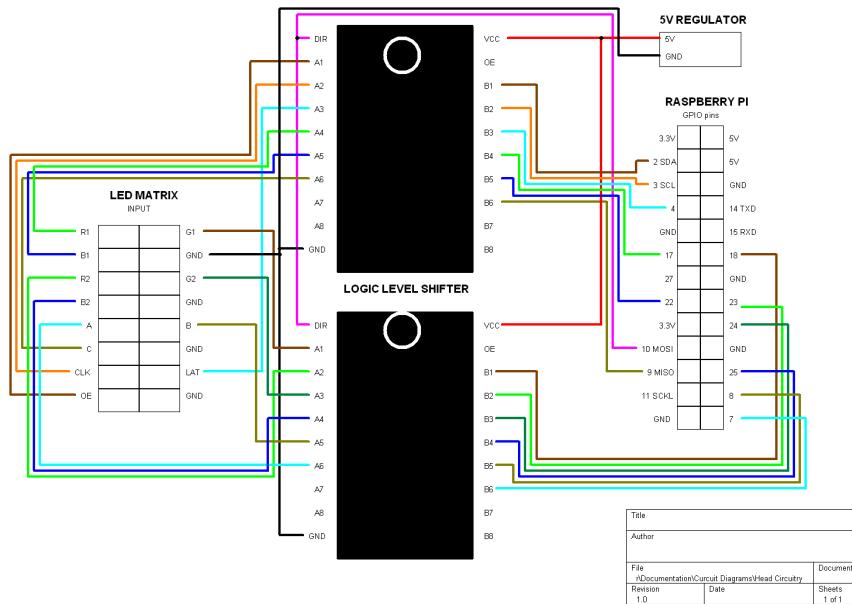


Figure 31: Raspberry Pi connections to LED Matrix

The Logic Shifter PCB is not included in the Adafruit project. The following steps describe how to include the Logic Shifter PCB in the wiring and routing. Figure 32 shows the pin connections on the logic shifter, labeled with which pin they connect to on the Raspberry Pi. We suggest putting the logic shifter board in the 3D printed head, and routing the wires through the body and up through the PVC pipe "neck".

4 CONNECTING THE LED MATRIX

Make sure to use the thicker gauge wire (20AWG) for powering the LED matrix as it can pull up to 5 amps of current. All the other signals to the LED matrix are digital signals for which the 30 AWG wire is sufficient.

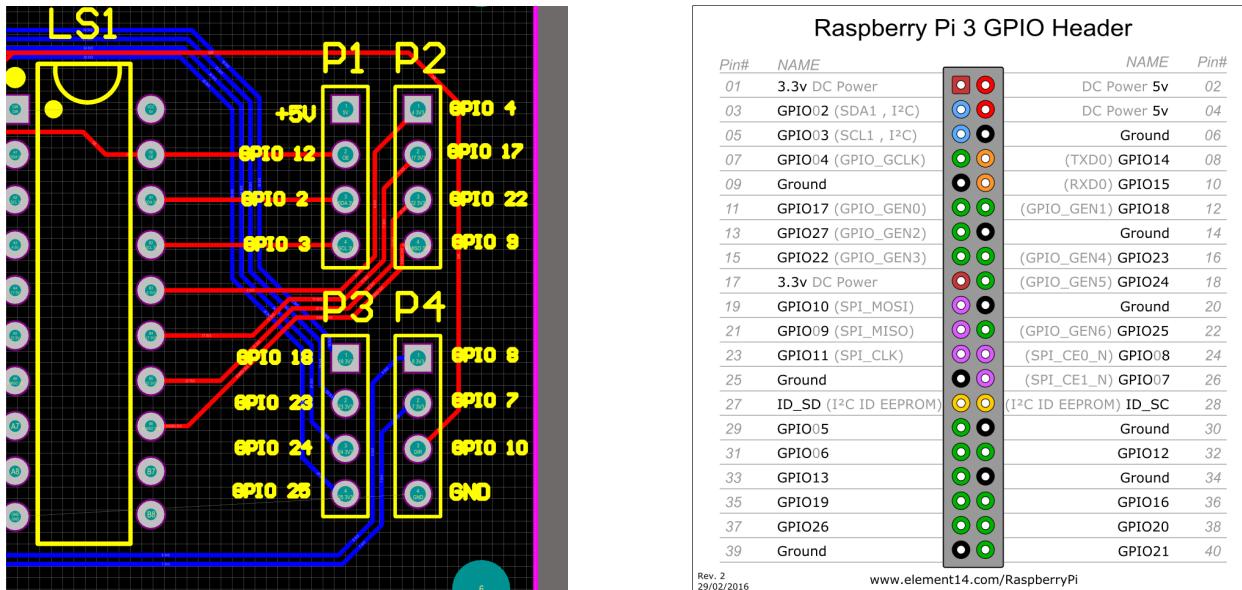


Figure 32: Raspberry pi to Logic Shifter board

****Note**** In Figure 32, the pin labels on the board (left) correspond to the pin NAME and not the physical pin number. For example, GPIO 12 is GPIO 12 and **NOT** physical pin #12.

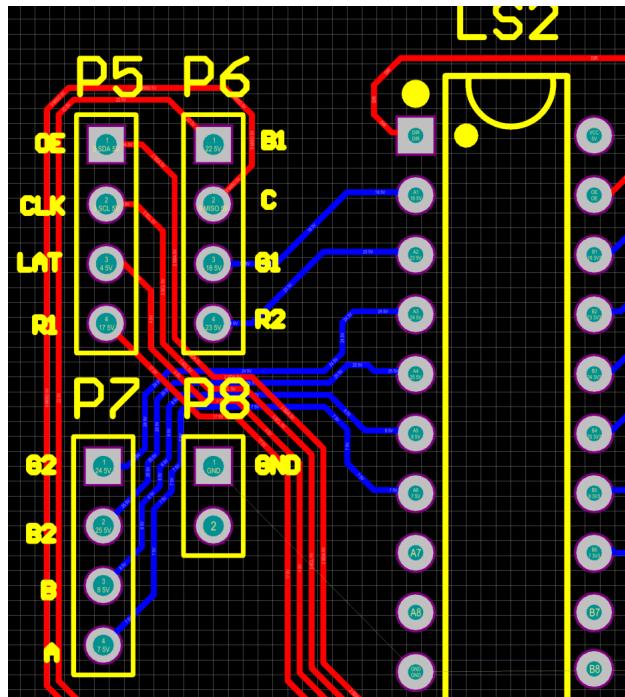


Figure 33: Logic Shifter to LED Matrix

Figure 33 shows the connections out of the Logic Shifter board to the LED Matrix. Make sure to use the INPUT side of the LED matrix. The pins' names/labels can be found directly on the LED Matrix.