

RoboJackets Electrical Training Week 3 Lab Guide

Kyle Nguyen

August 23, 2024
v1

Contents

1	Background	2
2	Initial Steps	2
2.1	Checking Libraries for Required Parts	2
2.2	Create a Project	2
3	Materials	2
4	Relevant Information	2
4.1	Adding parts to your schematic	2
4.2	Adding global labels	3
4.3	Changing your grid size	4
4.4	Decoupling Capacitors	5
5	Guided Lab	5
5.1	Symbol Placement	5
5.2	Organization	6
5.3	Wiring	7
5.4	Placing No Connect Flags	10
5.5	Running Electrical Rules Checker	11
6	Troubleshooting	11

1 Background

This week's lecture topic was on designing schematics with KiCAD. A majority of RoboJackets will use KiCAD to design printed circuit boards (PCBs) for their robots. For this reason, it's important to feel comfortable using this software and learn about the many different features KiCAD offers for PCB design.

In this lab, you will be creating your own schematic from scratch. Your job is to place the required components and connect them with nets. You will need to download some parts from [SnapEDA](#) as mentioned in Lab 2.

2 Initial Steps

2.1 Checking Libraries for Required Parts

The last lab required you to download some parts and put them into your KiCAD symbol and footprint libraries. Go into your symbol and footprint libraries and make sure the following parts are in there:

- [Teensy 4.1 \(Microcontroller\)](#)
- [Cherry MX Black Switches \(MX1A-11NN\)](#)
- [Motor Driver \(A4950ELJTR-T\)](#)
- [Voltage Regulator \(TSR 1-2450\)](#)

If you did not attend last week or you do not see these parts in your libraries, please talk to an instructor.

2.2 Create a Project

The last lab also had you make a project. If you have not created a project yet, click on **File → New Project** and choose whatever project name you want. Make sure the Create a new folder for the project checkbox is ticked, then click **Save**. This will create your project files in a new subfolder with the same name as your project.

3 Materials

- Computer/Laptop
- KiCAD
- Mouse

4 Relevant Information

4.1 Adding parts to your schematic

You can click the **Add a Symbol** button located in the right menu of the schematic editor or press “A” to open the part selection menu.

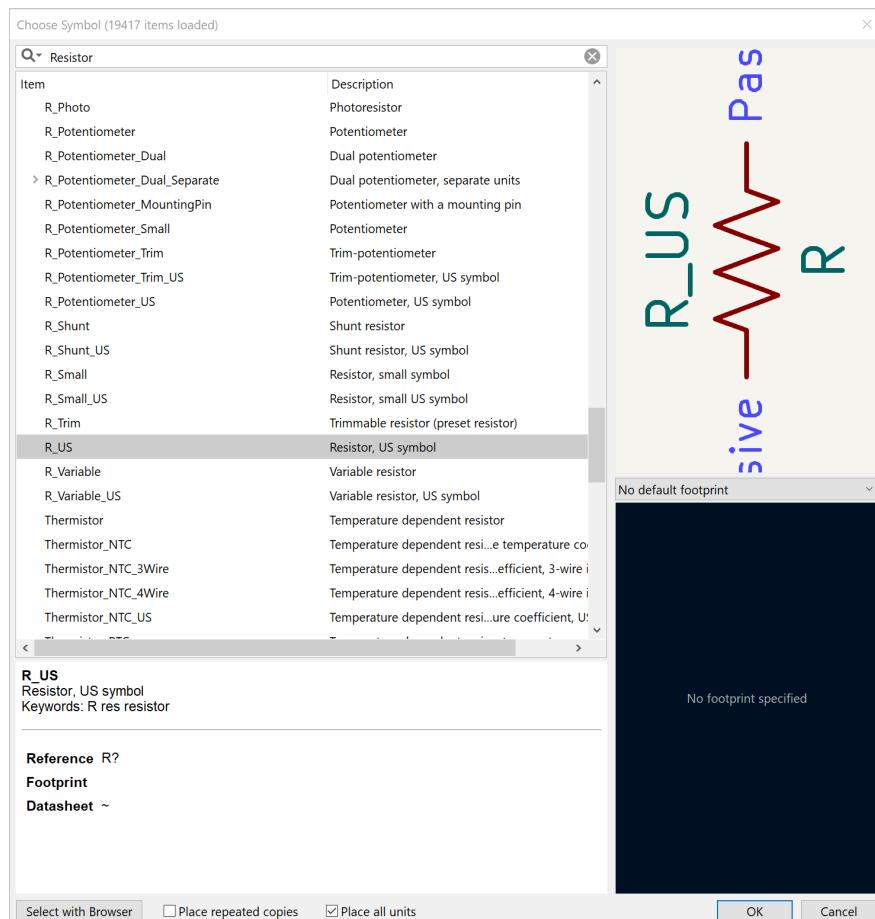


Figure 1: Add Symbol Menu

4.2 Adding global labels

If there are connections that need to be made far away from each other, you can use the “Add a global label” button in the right menu to do this.

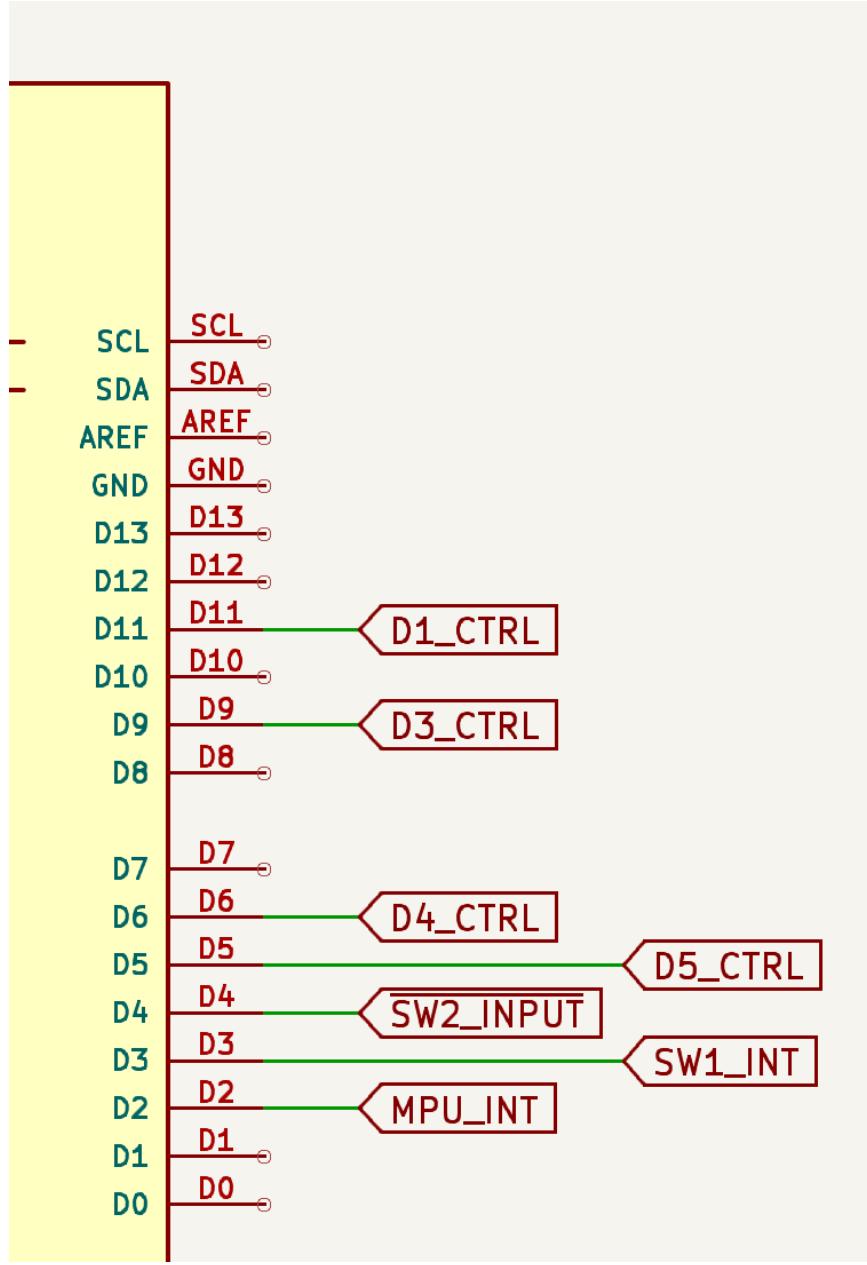


Figure 2: Adding labels

4.3 Changing your grid size

If you encounter any difficulties trying to connect pins in the schematic editor, you can press your CTRL key to get more precise placement. Alternatively, you can change your grid size. This can be done by going to View > Grid Properties and changing the size under “Current Grid”. A grid size of 2.54 mm (0.1 in) is recommended because most if not all symbols should have pins separated by 0.1”. However, if this is too large, then 0.254 mm (0.01 in) is recommended so that the grid is simply divided by 10.

You can also define your own grid size under the “User Defined Grid” section. Remember to set the current grid to User grid to actually use the custom grid size. Lastly, the Fast Switching section allows you to quickly switch between two different grid sizes which may be handy if you need to zoom in to any part of your schematic and have more gridlines (for Windows, you press Alt+1 or Alt+2 to switch to the

corresponding grid size).

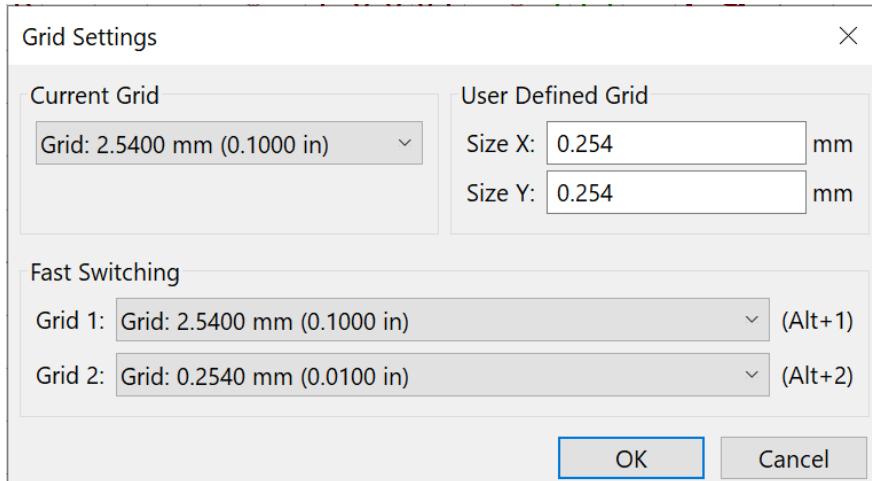


Figure 3: Grid Settings Menu

4.4 Decoupling Capacitors

Decoupling capacitors are capacitors that are normally placed between power and ground to protect the circuit parts from random spikes in voltage. These capacitors normally take values of $0.1\mu F$, but depending on the part, it could be different. To check what value capacitor you need or the wiring configuration, check the datasheet for the part.

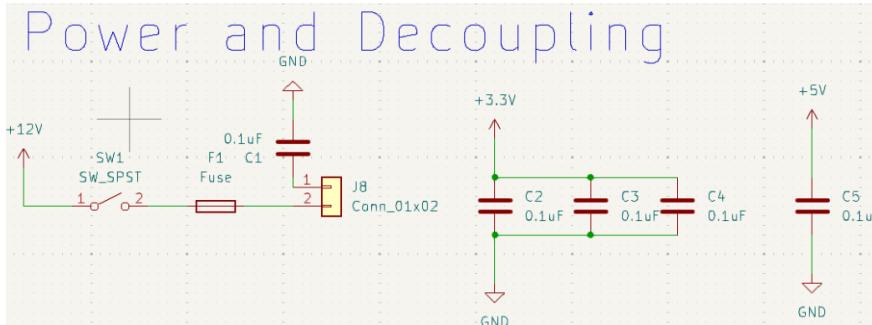


Figure 4: Some examples of decoupling capacitor applications in KiCAD.

5 Guided Lab

Open your project. Get ready to tweak! :3

5.1 Symbol Placement

The first step to making a schematic is placing all your parts down. When placing your parts, make sure all of your similar parts are next to each other. For bigger parts with lots of connections, try to place those farther away from the simpler parts.

5.1.1 Add the Microcontroller

- Select **Add a symbol** in the right bar and search for your microcontroller. The specific one we wanted should be labelled something along the lines of "DEV-16771" for the Teensy 4.1.

- Place this somewhere convenient.

5.1.2 Add the Motor Drivers

While writing this lab, I realized that the voltage regulator is going to take a lot of time to wire. To make your life easier, please click on this link that will redirect you to the electrical training Github.

- Download CopyAndPasteThisLab3.kicad_sch inside of the Week 3 folder.
- Copy and paste this into your schematic.

5.1.3 Add the Mechanical Switches

- Select **Add a Symbol** and search for the MX1A-11NN buttons you added.
- Place 2 of these switches somewhere convenient. Again, they don't have to be close to any of the other parts you have already placed.

5.1.4 Place the Voltage Regulator

- Select **Add a Symbol** and search for the TSR 1-2450 voltage regulator you added.
- Place this part somewhere convenient.

5.1.5 Place your Battery Connection

You may have noticed we did not include a battery! This is because KiCAD already has a connector symbol for what we are going to be using for this.

- Select "Add a symbol" and type "connector_generic" into the search bar.
- Select the 1x02 size and place it somewhere convenient.

5.2 Organization

At this point in your schematic creation, it is a good idea to organize your parts so it is more readable. To do this, we will border off our parts to separate them and label them so we know what specific purpose they serve.

5.2.1 Adding Borders

- In the panel on the right, select **Draw a Rectangle** and draw rectangles around your similar parts.

5.2.2 Adding Titles

- In the panel on the right, select **Add Text** and add descriptions of each rectangle at the top.
- A good text size is 200 mils, but you can choose whatever value you want (as long as you can read it).

5.2.3 Labelling Parts

To specify which symbol corresponds to what purpose, it is good practice to label them. To do this, right click on the symbol, click on **Properties**, and change the **Value** parameter to something meaningful. The symbols you do this for is completely up to you, but for this lab you should at least:

- Change the button values to "Button1" and "Button2".
- Change the motor driver values to "MotorDriverLeft" and "MotorDriverRight" (should already be done for you).

- Change the generic connector value to "BatteryConnector".

Hopefully, your schematic looks like this now:

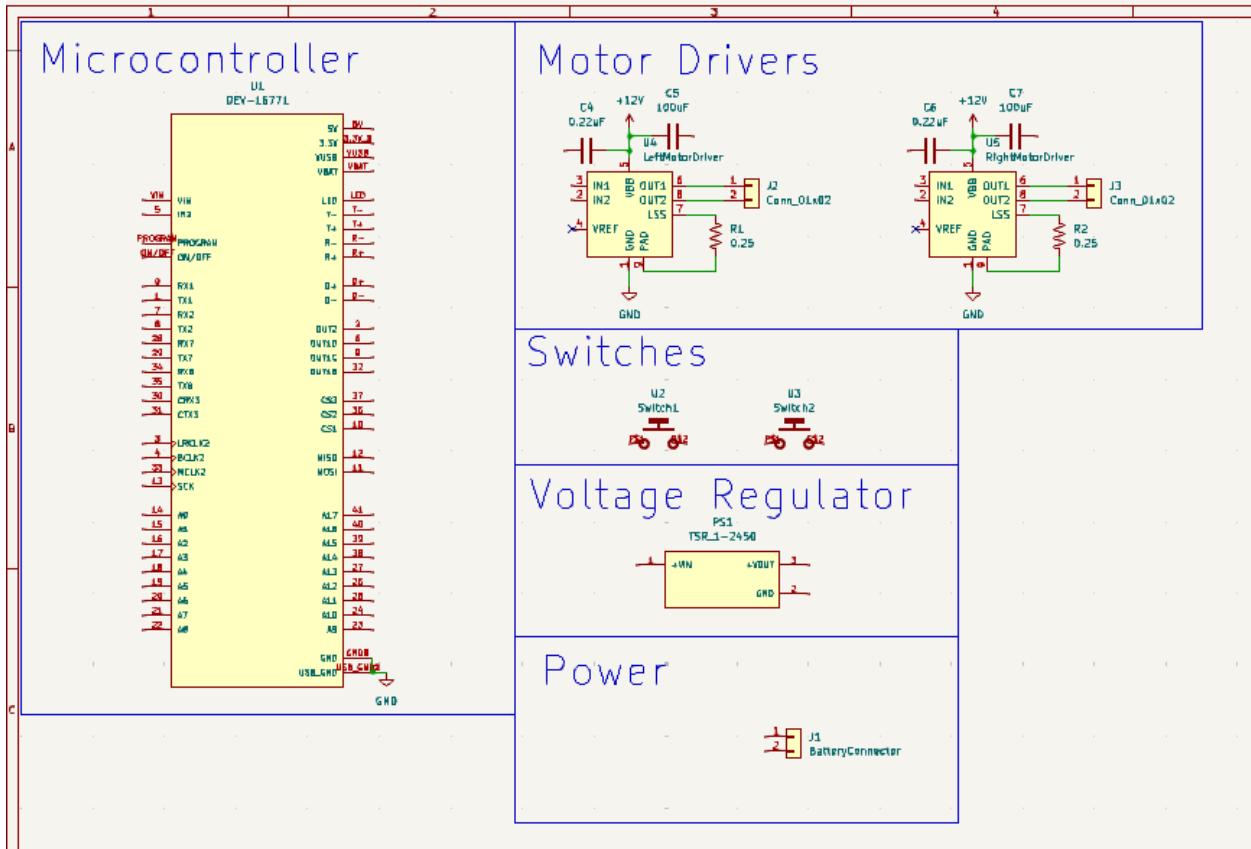


Figure 5: KiCAD schematic with all parts placed and organized for a specific PCB layout. Ignore the unconnected motor driver grounds.

5.3 Wiring

5.3.1 Microcontroller Wiring

- Start by wiring power to the Teensy. To do this, click on the **Add Power** panel on the right (or press P) and select 5V.
- Connect this to Vin by clicking on the tiny circle on either the microcontroller or the power symbol, and connecting the wire to the corresponding pin.
- Do the same to connect the 5V output and 3.3V outputs on the Teensy.
- Do the same to connect the Teensy ground(s) to GND.
- Decouple the powers! For more information on how to do this, refer to section 4.5 on **Decoupling Capacitors**, or copy the image below.

Hopefully, your microcontroller looks like this now:

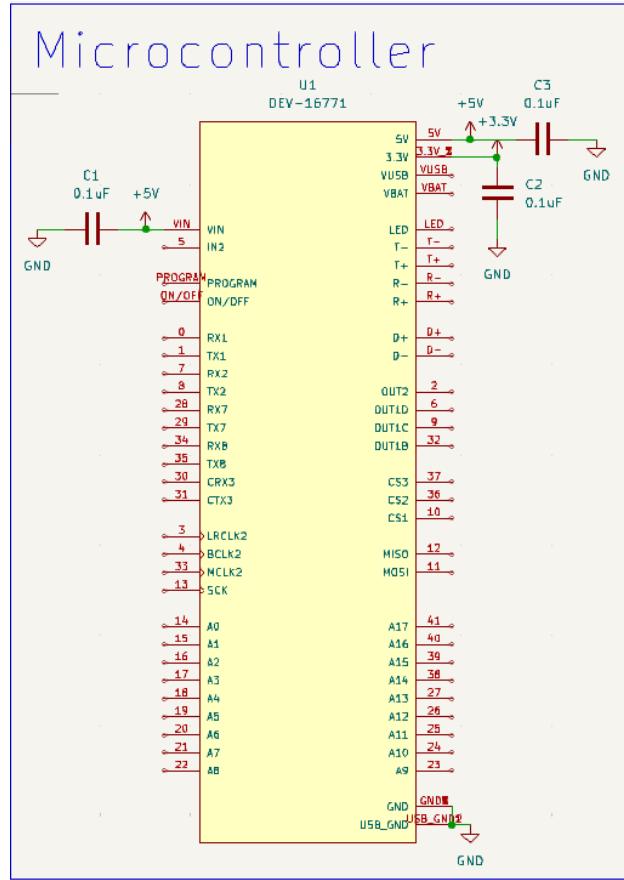
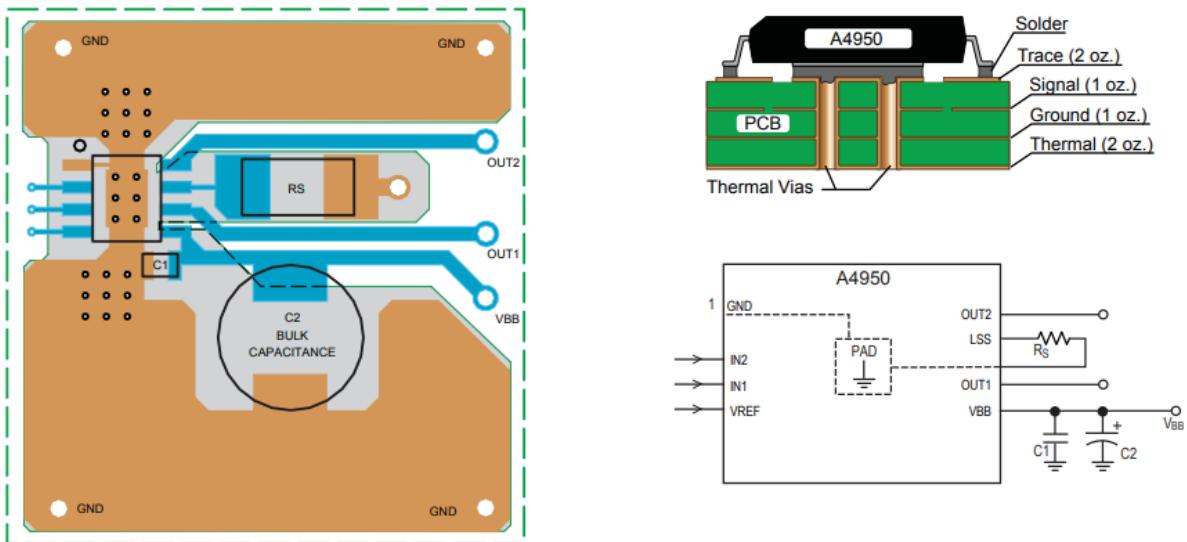


Figure 6: Teensy 4.1 with pins connected to power. The microcontroller is properly decoupled.

5.3.2 Motor Driver Wiring

I have already done the motor driver wiring. I did it by looking at the datasheet for this part. However, there are two important connections left to make, and those are the motor driver inputs, which come from the microcontroller.

- To add a global label, click on "Add Global Label" on the right panel. Name this label "MD_Left1" and wire it to IN1 of the left motor driver.
- Wire another global label with the same name to any PWM pin on the microcontroller. Click this [link](#) for a picture of the Teensy 4.1 pinout.
- Repeat this process for IN2 of the left motor driver, and IN1 and IN2 of the right motor driver.



Bill of Materials

Item	Reference	Value	Units	Description
1	RS	0.25 (for $V_{REF} = 5$ V, $I_{OUT} = 2$ A)	Ω	2512, 1 W, 1% or better, carbon film chip resistor
2	C1	0.22	μF	X5R minimum, 50 V or greater
3	C2	100	μF	Electrolytic, 50 V or greater

Figure 7: Section of the datasheet for the A4950ELJTR-T motor driver showing how to wire the component.

5.3.3 Wiring the Switches

Add power, resistors, and global labels to wire the switch like so:

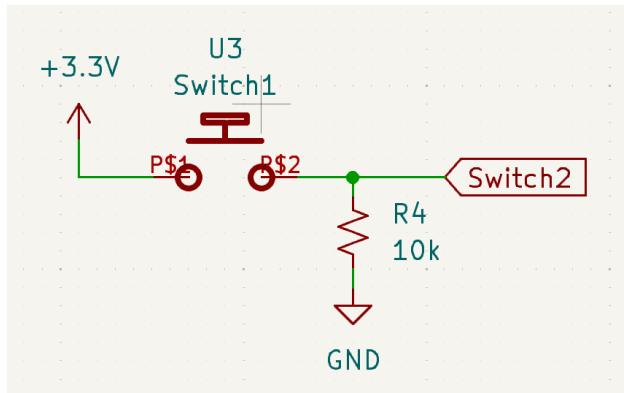


Figure 8: Demonstration of a pull-down resistor in a KiCAD schematic.

Some things to note:

- Power here is 3.3V because the PWM pin takes in a 3.3V signal.
- The resistor is there to guarantee that the signal is 0 (grounded) when the switch is up. This resistor is called a pull-up or pull-down resistor. For more information, you can look up pull-up or pull-down resistor on

Google for many more examples.

- Attach the other end of the global label to any PWM pin on the microcontroller.

5.3.4 Wiring the Voltage Regulator

The voltage regulator takes in a certain range of voltages and outputs a fixed voltage. Wiring this component is fairly simple.

- Vin will be the voltage of our battery, which is 12V in this case (it depends what battery you are using). Attach 12V to the Vin pin.
- Vout will be the output voltage of the voltage regulator, which is 5V for this part. Attach 5V to the Vout pin.
- Wire ground to the GND pin.
- Decouple your powers to ground (12V and 5V).

5.3.5 Wiring the Battery

A battery has a positive end and a negative end. We will assume the negative end to be GND and attach each end of the connector to a terminal of the battery.

- Wire pin 1 of the connector to GND.
- Wire pin 2 of the connector to 12V.
- Add a decoupling capacitor between pin 1 and GND.

For our purposes, we will use a simple battery connector like this without any circuit protection. However, you will probably have to add a fuse and a switch to your battery connections to protect your components from getting fried. For more information on this, check out the **RW Schematic Tutorial Slides** that I made in the Troubleshooting section.

5.4 Placing No Connect Flags

- For your unconnected Teensy pins, place **No Connect Flags** on them. The icon is on the right, and it looks like an X.

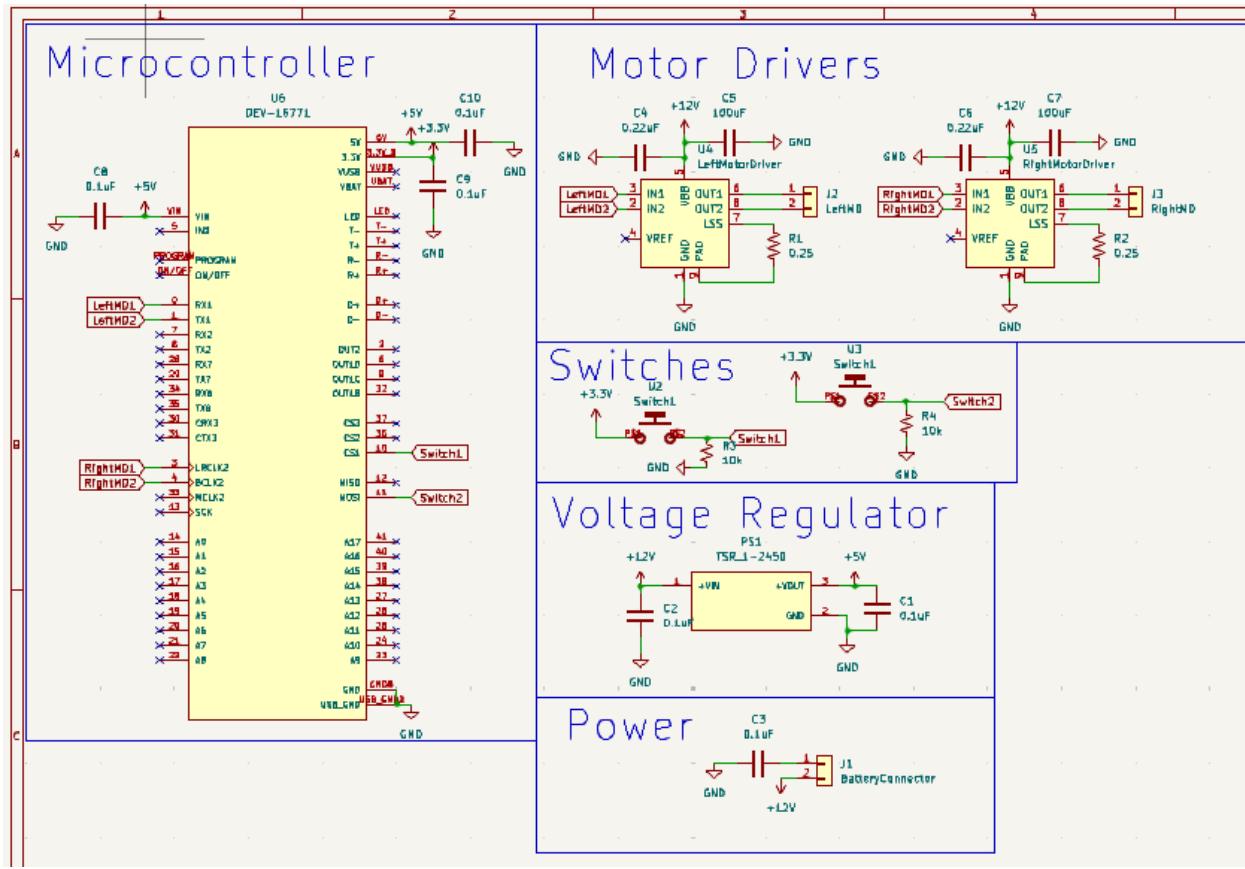


Figure 9: Final KiCAD schematic of a specific PCB.

5.5 Running Electrical Rules Checker

To check if there are any issues with your schematic, run the **Electrical Rules Checker (ERC)** by clicking on the checklist icon in the top bar. It's recommended to run ERC while your grid size is on 0.1 in to ensure all of the wires and parts are spaced appropriately.

6 Troubleshooting

If you have any questions or concerns, please ask your instructor. Most ERC errors can be fixed quickly with a simple Google search. Here are some additional resources if you need help.

- [KiCAD Getting Started Guide](#) - Includes how to use the schematic editor
- [RW Schematic Tutorial](#)