ΘT PI DELTA CHAPTER

2025

RushBot Project Report

PROJECT MANAGER:
DIEGO SOLORZANO

UNIVERSITY OF CALIFORNIA, IRVINE

Table of Contents

Chapter 1: Budget Breakdown

Electrical Budget Breakdown			
Item	**Price**	**Quantity**	**Total**
5V Power converter for Pi	$8.99	1	$8.99
nema17 stepper motor	$13.99	2	$27.98
magnetic encoder - 3 pack	$9.99	1	$9.99
Arduino power Converter	$8.99	1	$8.99
12V Buck Converter	$8.99	1	$8.99
Battery voltage sensor	$9.99	1	$9.99
Battery current sensor	$11.58	1	$11.58
Wheel servos	$43.00	1	$43.00
Power switch -30A	$7.99	1	$7.99
mux - 6 pack	$9.99	1	$9.99
5V Power Convertor for Servos	$18.69	1	$18.69
USB hub	$14.00	1	$14.00
12V power supply	$11.99	1	$11.99
Arduino mega	$22.99	1	$22.99
Limit switches	$4.99	1	$4.99
EVE 26V 18650 2550mAh 7.5A Battery	$1.70	32	$54.40
BMS	$8.07	1	$8.07
12V DC Motor with Wheel Encoder	$5.86	6	$35.16
12 DC Motor Drivers -4pc	11.99	1	11.99
INRFZ44N Mosfet	$1.36	1	$1.36
Stepper Motor Driver TB6600 4A 9-42V	$9.98	2	$19.96
Lidar Sensor	$99.99	1	$99.99
Total			**$451.08**

Chapter 2: Electrical Subteam

Name: Luke Vargas
Email: lukekvargas@gmail.com

- Creating Wiring Schematics and designed several components on Altium Designer for Arduino and peripherals and the power circuit
- Integrated electrical components

Name: Dalena Nguyen
Email: dalenanguyenwork@gmail.com

- Helped create schematic for ARDUINO and PI wiring diagram

Name: Arim Song
Email: arimsong0430@gmail.com

- Helped combine Arduino and Rasp pi wiring diagrams by linking key power components for full electrical schematic

---

Name: Keiko Yamamuro
Email: keiko.yamamuro@gmail.com

- Performed power calculations and helped to build a battery using 18650 cells

---

Name: David Culciar
Email: davidculciar@gmail.com

- Assisted in battery design and testing

Name: Shana Chao
Email: shanaxchao@gmail.com

- Integrated and tested electrical components, diagnosing and resolving performance issues

---

Chapter 2.1: Voltage System Hierarchy

RushBot is powered by a 14.8V 20400mAh battery comprised of 32 18650 Lithium Ion battery cells arranged in a 4s8p configuration. Since the nominal voltage of each cell is 3.7V, connecting 4 in series would give us an overall nominal voltage of 14.8V, with 16V when the battery is fully charged. Connecting 8 in parallel allows us to increase our battery life and reach 20400mAh. Lithium-ion was chosen as opposed to Lithium-polymer due to its more level discharger over time. This allows for more time at nominal voltage and this results in increased efficiency. A traditional 12V battery was found to not provide enough power to the system due to a voltage drop after the voltage regulator. Constructing a battery with a higher initial voltage proves to be more efficient as it requires less power to step down compared to stepping up the voltage to our required 12V. The final power calculations of the rover's battery life are shown below in Table 1. Current draw for each state is gathered from the datasheets of all components and the calculations are found using the following equation:

$$Battery\ Life\ =\ \frac{Total\ Amp\ Hours}{Current\ Draw}$$

	Idle	Active	Stall
Run Time (Hours)	33.12	4.13	0.79

Table 1: Battery Life for three different states

RushBot has a power management system in order to ensure safety and stable operation, which utilizes a switch, one 12 V step-down buck converter, and two 12V-to-5V regulators. The battery itself features a 30A battery management system ensuring even discharge and charging of each cell.

The battery switch is the first component that the battery is connected to and serves as a reliable method to disconnect power from the entire system. This controlled power activation ensures both safety and efficiency as it prevents potential component damage and overheating in the event of malfunction.

Rushbot uses two types of regulators: a 12 V step-down regulator and two 12V-to-5V regulators. Regulators help protect the system by ensuring that each component receives the correct voltage, preventing overvoltage that could damage the electronics. The 12V regulator provides a stable voltage for the NEMA 17 stepper motors. Meanwhile, the 12V to 5V regulators step down the voltage to 5V, delivering 3A to the Raspberry Pi and 8A to the servo motors, ensuring both components operate safely at their required voltage levels. The wiring diagram for the voltage systems is shown below in Figure 1.

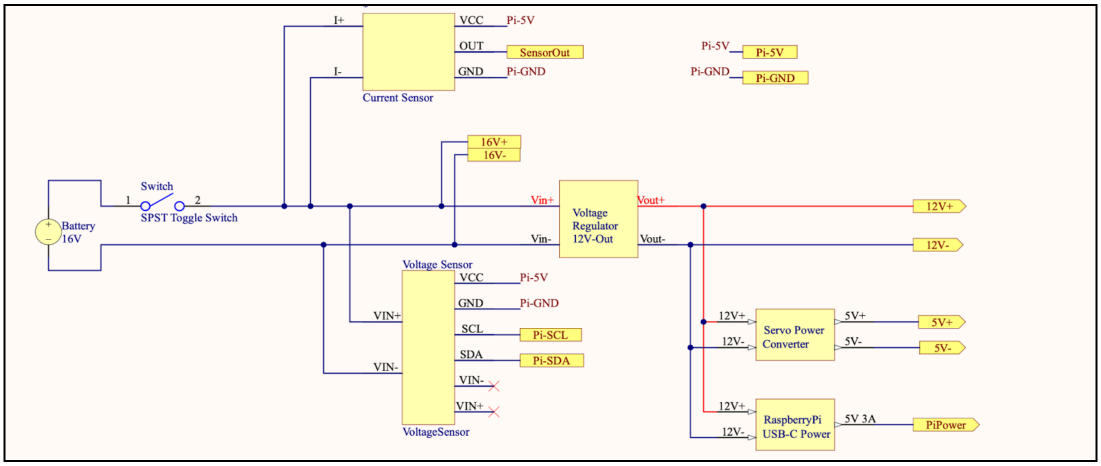

Figure 1: Power Wiring Diagram

Chapter 2.2: Motors and Components

Motor Control

RushBot uses a total of 6 12V brushless DC motors, 2 NEMA 17 stepper motors, 4 High Torque servo motors, and 1 vacuum pump motor for a total of 13 motors controlled by the Arduino Mega. A Mega is chosen due to its extra PWM signal pins which all of the motors operate on.

3 L298N drivers control all 12V DC Motors. Each unit acts as dual drivers allowing for 2 motors to be controlled by each one. Due to the varying turning radius of each motor, the DC motors must be controlled in 4 different to account for the varying speed of each wheel. For this reason, 1 of the 3 drivers will individually control 2 motors separately while the other drivers control their motors together. In total, this allows for 4 different speeds. Additionally, each DC motor is equipped with onboard encoders that are read via digital pins.

2 TB6600 drivers control the stepper motors. These drivers, while bulky, are chosen due to their excellent heat dissipation, which is a common issue seen in most stepper motor drivers. This was especially crucial for our design as these stepper motors would be under high load from moving the arm, resulting in a higher average current draw. Attached behind the stepper motors are AS5600 magnetic encoders used to accurately track the motors' rotation. This data is communicated through an I2C protocol. However since both drivers have the same address, a multiplexer is needed in order to properly differentiate the two when communicating to the Arduino.

The vacuum Pump Motor is controlled using a logic-level NMOS switch circuit. The INRFZ44N NMOS channel mosfet was chosen as the switch due to its triode operation region when a voltage of 5V is applied at the gate. This characteristic of the mosfet makes it ideal when interfacing with Arduino since its digital pins operate at the 5V logic level. A 10 ohm gate resistor is used to limit the voltage at the gate and a 100K ohm pull-down resistor is used to provide a return path for the current in order to properly switch off the mosfet.

Lastly, the 4 servos are directly controlled by the Arduino's PWM pins, and 2 switches between the joints of the arm to prevent over-extension and protect circuitry and structure. All Arduino connections can be seen in Figures 2-4.

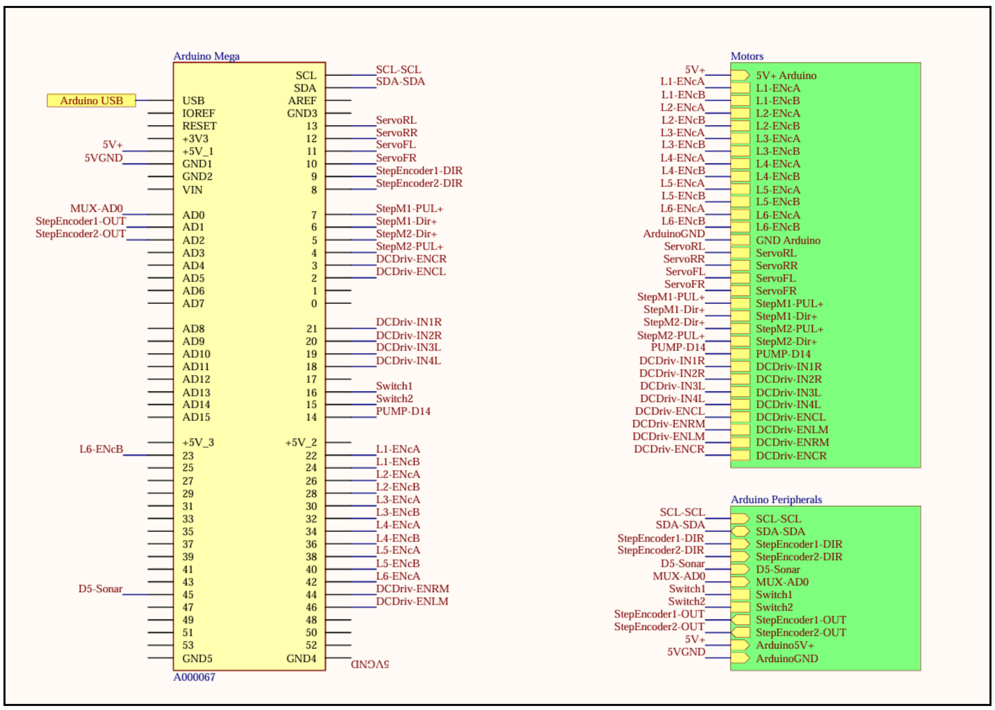

Figure 2: Arduino input/ouput connections

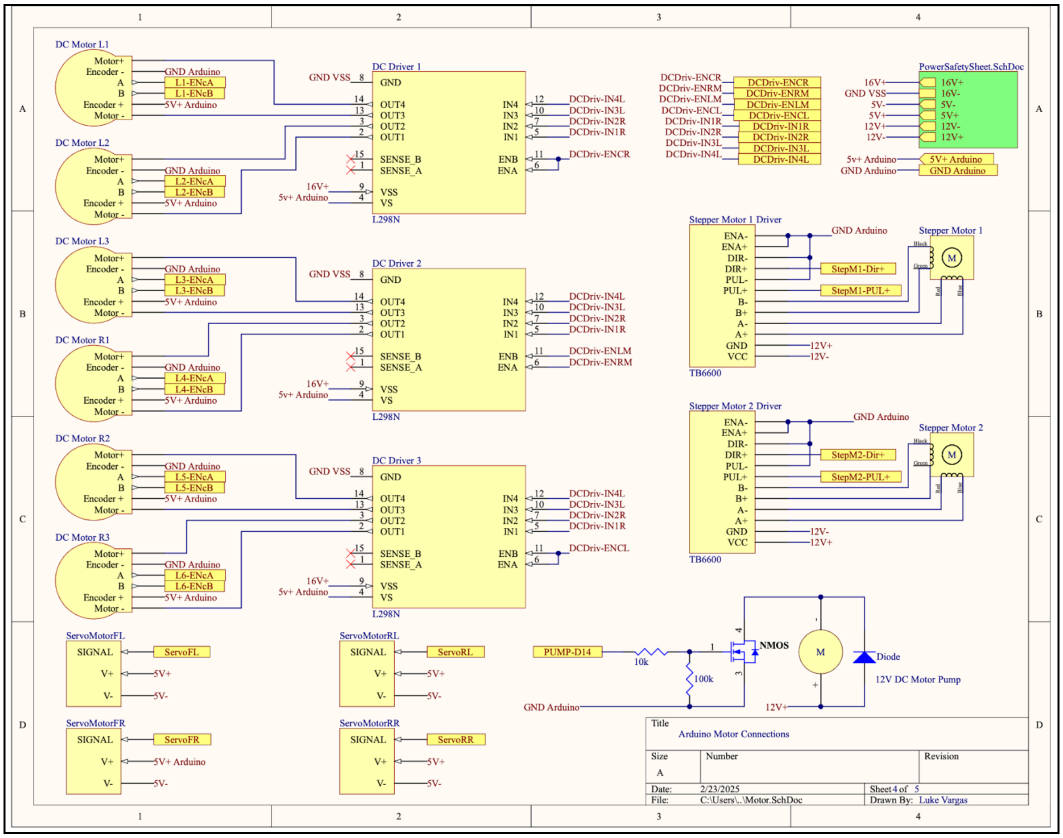

Figure 3: Motor Connection Wiring Diagram

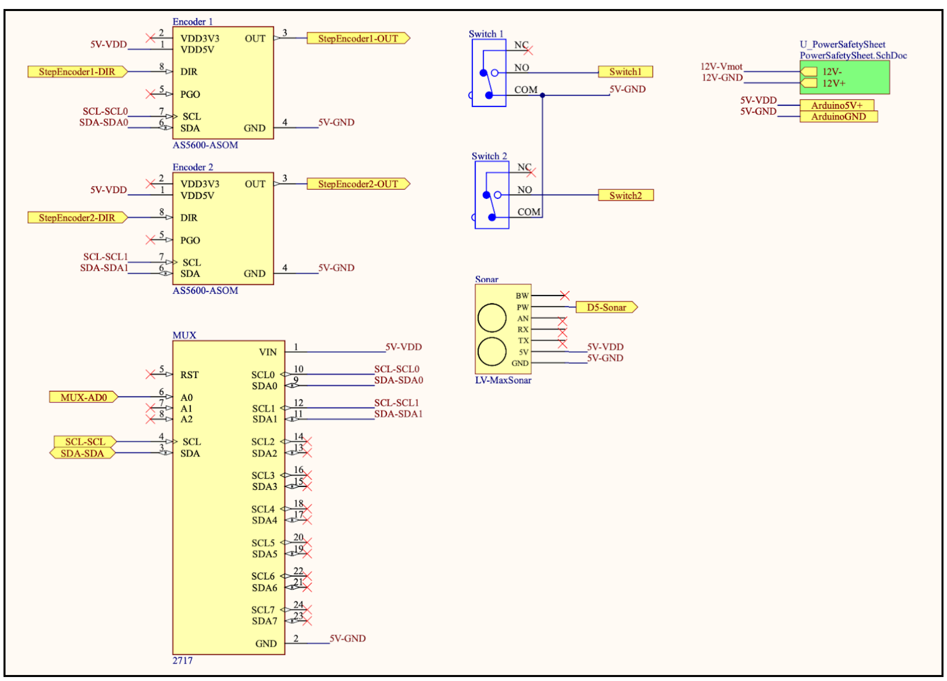

Figure 4: Encoder, Mux, and Limit Switch Wiring Diagram