ECE 445

Spring 2021

Design Document

Head-Motion Controlled Wheelchair

Team Members: Dev Manaktala, Arnav Jain, and Jiayuan Liu

Group 55

TA: Yichi Zhang

Date: 2/29/2021

1. Introduction

1.1 Objective

Classical wheelchairs are designed to be operated by people who have the ability to maneuver them with a functioning arm. There are people, such as amputees and those suffering from paralysis, that cannot operate such wheelchairs.

We propose a head-motion controlled wheelchair that would allow people with such disabilities to be able to travel. The wheelchair would have a mounted camera facing the user and use computer vision to detect head motion to move and turn the wheelchair. The wheelchair would also have a vibration module allowing users with compromised vision to maneuver. This would be done through IR/ultrasonic sensors on the wheelchair detecting obstacles in the path and giving vibrational feedback to the user.

1.2 Background

Existing prototypes of such wheelchairs either use eye movement as user input or require the user to wear some sort of device on their head. We believe that our design could provide a much simpler and more viable solution compared to these since head motion is a more natural way to provide input. Furthermore, one of the features that distinguishes us from current alternative options is the object detection system as this would add another layer of safety to the wheelchair.

1.3 High Level Requirement

- Wheelchair should be able to move instantaneously with head motion with minimal lag (less than 1 sec)
- Wheelchair motors should have a good enough torque to move any person upto 250 lbs between speeds of 4.5 and 8 mph
- Wheel-chair should not have false-positive inputs - the wheelchair should not move if the user did not intend it to.
- The ultrasonic sensor should be able to detect object upto 3 meters away accurately

2. Design

The design consists of four major subsystems and a microcontroller. The power supply subsystem consisting of both high and low voltage batteries provides power to the microcontroller, the Raspberry Pi, and the motor controller circuits. The motion detection subsystem involves the camera and the Raspberry Pi. Frames are sent from the camera to the Raspberry Pi [1]. The Raspberry Pi processes the data and sends output to the microcontroller. The microcontroller gives instructions to the motor controller circuits and thus drives both traction motors. The object detection subsystem measures objects close to the surface by ultrasonic sensors and gives vibration feedback to the users through the vibration motors. The Raspberry Pi, the microcontroller, and two sets of motor control circuits will be on a single board to avoid external wiring and extra power distribution.

2.1 Block Diagram

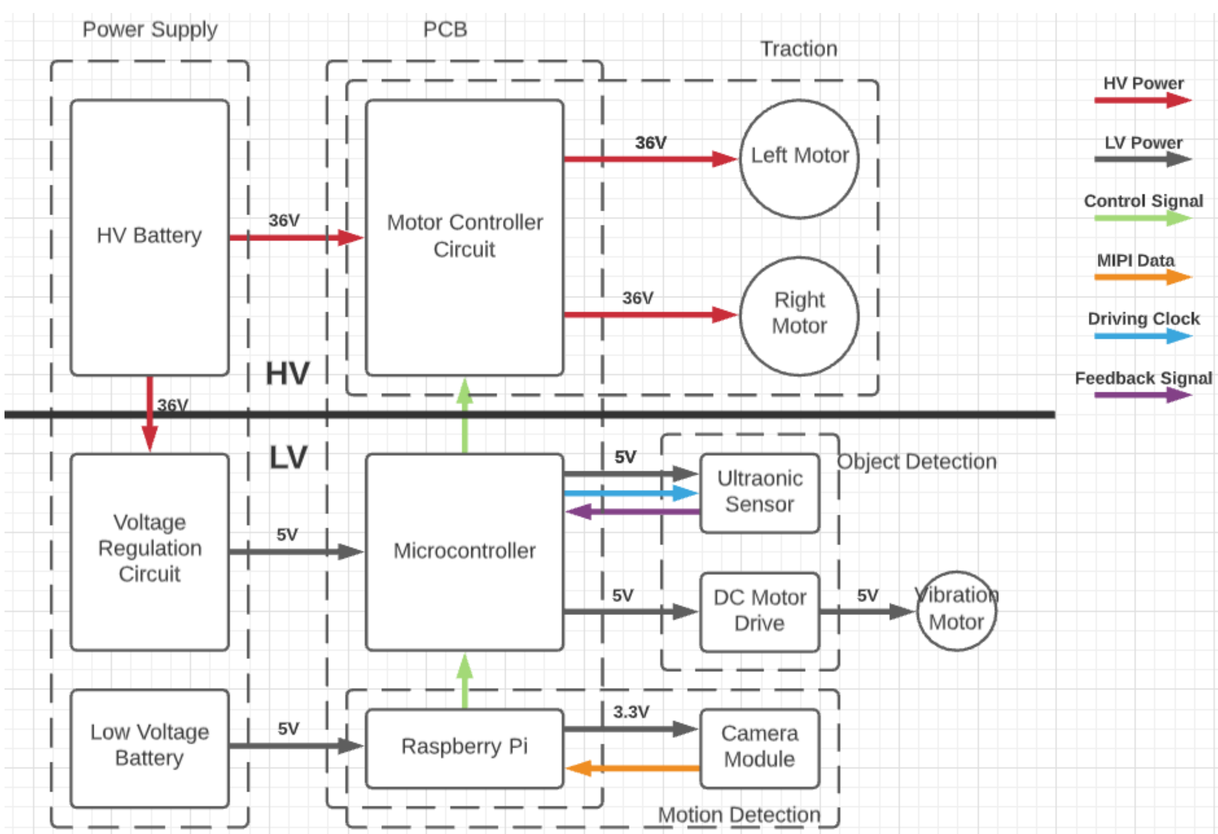

Figure 1. Block Diagram

2.2 Physical Design

Figure 2. Wheelchair Physical Diagram

3. Subsystems

3.1 Power Supply

3.1.1 Functional Overview

The power supply consists of a 36V high voltage battery pack and a power regulation circuit. The high voltage battery powers a pair of motor control circuits and thus powers two traction motors. The 36V voltage is regulated to both 24V and 5V by a power regulation circuit. The 5V voltage powers both the microcontroller and the Raspberry Pi. It also directly powers the vibration motor because the current required by the vibration motor may exceed the current limit of the microcontroller. The 24V voltage is the pull-up voltage for the power mosfets.

3.1.2 High Voltage Battery

Inputs: Recharge Port

Outputs: +36A, +36B

Description:

The high voltage battery provides power for two 500W motors, so the power rating of the battery pack should be at least 1kW. The battery pack should be rechargeable because batteries at this voltage are expensive and switching high voltage batteries is not safe for users. Users need to be able to recharge their wheelchair without changing the high voltage battery. Since 36V is a relatively high voltage and more than 20A of current will be drawn from the battery, we carefully choose the battery connector and the motor connectors. Since the power rating of our connector is 9.5A, we choose to use two high voltage batteries and limit the current of each battery to be under 9.5A.

Requirement	**Verification**
1. The battery should be able to provide at least 1kW to power both traction motors. 2. The voltage cannot decrease over 10% at full load. 3. The battery should be rechargeable.	1. Use a current sensor to measure the current from battery to circuit and use voltage meter to measure voltage. Calculate the power. 2. Use a voltage meter to measure the voltage across the battery when both motors are drawing full power from it and see how much voltage it decreases. The voltage difference should be less than 3.6V. 3. Recharge the battery according to the datasheet and measure the voltage between the terminals and see if it is at least 35V.

3.1.3 Power Regulation Circuit

Inputs: +36A

Outputs: +24, +5

Description:

To achieve both 24V and 5V from the 36V supply voltage, we need to step down twice. We choose the LM317 linear regulator because it has a 40V maximum input voltage, which is higher than the 36V of our supply voltage. It also has a 1.5A output current, which is enough for the microcontroller, Raspberry Pi, and also the vibration motor.

The output of the linear regulator can be calculated as $V_{out} = 1.25 \cdot (1 + R2/R1)$ [9]. Therefore, we select R2 = 3.65k and R1 = 200 for stepping down to 24V. Besides, we select R2 = 3.6k and R1 = 1.2k to step down from 24V to 5V. Since the power of a linear regulator is approximately $(V_{out} - V_{in}) \cdot I$, a lot of power is dissipated by it. Therefore, we choose a surface mount package and will design a large metal polygon to the heat dissipation pad of the linear regulator.

Requirement	**Verification**
1. The voltage variance at the 5V output cannot exceed 5% which is 250mV. 2. The power regulator should be able to continuously provide at least 50mA at 5V to power both the MCU and the ultrasonic sensor. This is because the MCU draws 42mA [12] and the ultrasonic sensor draws 8 mA [6].	1A. Use an oscilloscope to probe the output at the 5V terminal. 1B. Measure for at least a minute and note the highest and lowest values. Both values cannot be more than 250mV from 5V. 2A. Connect a 100 ohm resistor between the 5V output and ground, 2B. Use a multimeter to measure current through the resistor. The temperature of both LM 317 linear regulators should not increase significantly.

3.2 Motion Detection

3.2.1 Functional Overview

Inputs: MIPI Data from Camera Module, External 5V from Separate LV Battery

Outputs: SpeedL, SpeedR

Description:

The motion detection subsystem involves the camera and the Raspberry Pi. The camera will be mounted on the wheelchair facing the user and send each frame to the Raspberry Pi. We will then be running HaarCascade face detection [2] algorithm on startup. We will then extract features and calculate deviation of eyes from the center, compare it to a threshold value and predict desired input by the user. Furthermore we will be testing the need for facial detection and tolerance for the number of frames that do not require face detection.

Requirement	Verification
Face motion recognition must operate with minimal lag (<1 sec)	1A.Implement HOG face detection model. 1B. Create test samples of different faces/backgrounds and calculate accuracy as detected faces/total samples. 1C. Calculate false positives as incorrect detections/total samples. 1D. Expected values for accuracy is 90% and false positive rate is close to 0. 1E. Run a field test and calculate frames/second.

3.3 Traction System

3.3.1 Functional Overview

The traction system is powered by the high voltage battery at 36V. It allows the wheelchair to both move straight and turn left or right by adjusting the speeds of two motors according to the signals received from the Raspberry Pi. When receiving a moving forward signal from the microcontroller, the motor controller controls two motors to operate at the same speed. On the other hand, when receiving a turning signal, the motor controller moves the two motors in opposite directions to turn the wheelchair.

3.3.2 Motor Controller Circuit

Inputs: SpeedL, SpeedR, +24, +5, Echo, Trig

Outputs: MotorL, MotorR, MotorV

Description:

We use a combination of BJT and power mosfets to control the motor. Since the current rating of a BJT is limited, we use a BJT to control three mosfets. Since we have three mosfets in parallel, the large current through the motor can be divided into three paths. Each mosfet only needs to handle ⅓ of the current. Just like the LM317 linear regulators, we choose the surface mount package so that it can dissipate up to 2W [10] through the drain pad (heat dissipation pad). We choose BC547 as the BJT. BC547 is the high voltage version of the commonly used BC548 and it has a 45V breakdown voltage [11], which is safer for our 24V design. Although the 30V voltage rating of BC548 is also above 24V, we always want to leave a larger safety margin.

Requirement	**Verification**
1. When the MotorL signal at the base of the left BC547B BJT is pulled to ground, all three left IRFZ44S MOSFETS should be turned on. 2. When the MotorR signal at the base of the right BC547B BJT is pulled to ground, all three right IRFZ44S MOSFETS should be turned on. 3. When the MotorV signal at the base of the BC548 BJT is powered at 5V, the BC 548 BJT should turn on.	1A. Apply ground to the base of the left BC547B BJT. 1B. Measure the resistance between the drain net (MotorL-) of the three left MOSFETS and ground. The resistance should be less than 0.2V, which means that this net is connected to ground. 2A. Apply ground to the base of the right BC547B BJT. 2B. Measure the resistance between the drain net (MotorR-) of the three right MOSFETS and ground. The resistance should be less than

4. The traces of the three phase output current on the PCB should be at least 150mil in width [3]. The traces of the DC power input should be at least 300mil in width.	0.2V, which means that this net is connected to ground. 3A. Apply 5V to the base of the BC548 BJT. 3B. Measure the resistance between the drain net (MotorV-) of the BC548 BJT and ground. The resistance should be less than 0.2V, which means that this net is connected to ground. 4. Verify the circuit with Altium current check.

3.3.3 Microcontroller

Requirement	**Verification**
1. When the SpeedL signal on PC0 is high at 5V, the MotorL signal on PD3 should be pulled to ground. 2. When the SpeedR signal on PC1 is high at 5V, the MotorR signal on PD6 should be pulled to ground. 3. When the object detection algorithm calculates a distance below 5m, the MotorV signal on PB1 should be 5V.	1A. Apply 5V to PC0 on the MCU. 1B. Use a multimeter to measure the resistance between PD3 and ground. The resistance should be less than 2 ohms, which means that PD3 is grounded. 2A. Apply 5V to PC01 on the MCU. 2B. Use a multimeter to measure the resistance between PD6 and ground. The resistance should be less than 2 ohms, which means that PD6 is grounded. 3A. Operate the object detection algorithm. 3B. Use the Arduino console to display the

	detected distance. 3C. When the displayed distance is less than 5m, use a multimeter to measure the voltage at pin PB1. It should be 5V.

3.3.4 Traction Motors

Inputs: +36A, +36B

Outputs: Friction Wheels

Description:

We use the BestEquip 36V 500W brushless motor because both its voltage and power rating fits our application. Brushed motors have an efficiency range between 75% to 80%, so the estimated heat dissipation of each of our motor is at most $0.25 \cdot 500W = 125W$. Since we are running our motors at at most 9.5A, which is the current rating of the connectors, we are not running at full power. Our heat dissipation of each motor is at most $0.25 \cdot 9.5A \cdot 36V = 85.5W$. The length of the motor is more than 15cm and the radius is over 6cm, so the total surface area is more than $565cm^2$. Therefore, the power density over the motor is less than $0.151 J/cm^2$. This is securely under the safety line.

3.5 Object Detection

3.5.1 Functional Overview

Inputs: +5

Outputs: Echo, Trig

Description:

The object detection subsystem measures objects close to the surface and gives vibration feedback to the users. When an object is detected by the ultrasonic sensors, a signal is transmitted to the microcontroller. The microcontroller toggles a signal to make the DC motor drive to operate the vibration motor. The vibration motor would be activated based on a gradient depending on the proximity of the object - closer objects would give higher vibrations.

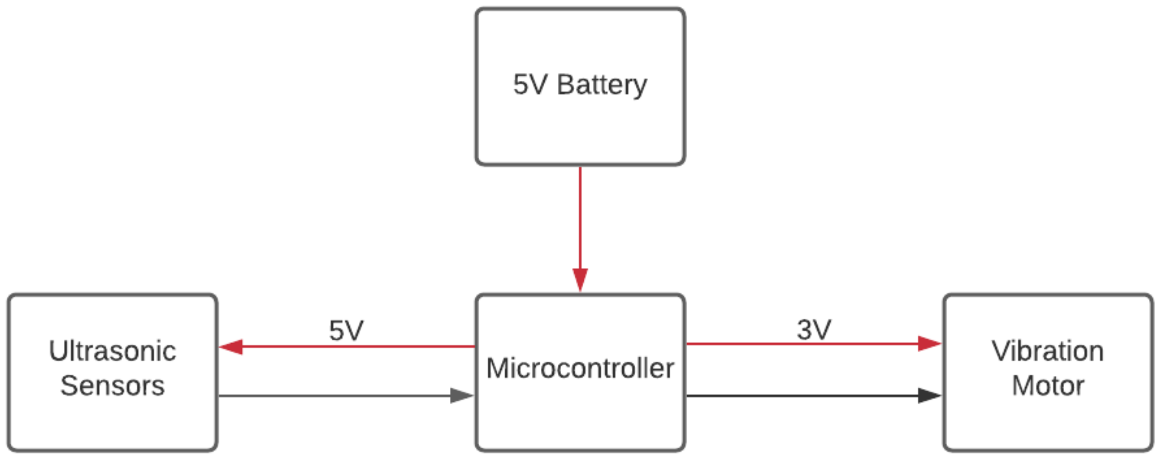

Figure 3. Object Detection Subsystem Block Diagram

Requirement	**Verification**
1. The sensor would need to accurately detect objects upto a distance of 3 meters.	1A. Assemble circuit on breadboard with ultrasonic sensors and microcontroller - get distance measurements on a monitor. 1B. Test sensors with the setup and place objects at 1, 2 and 3 meters in succession. 1C. Test multiple ultrasonic sensors to check which one gives the most accurate values.

3.6 Schematics

3.6.1 MCU

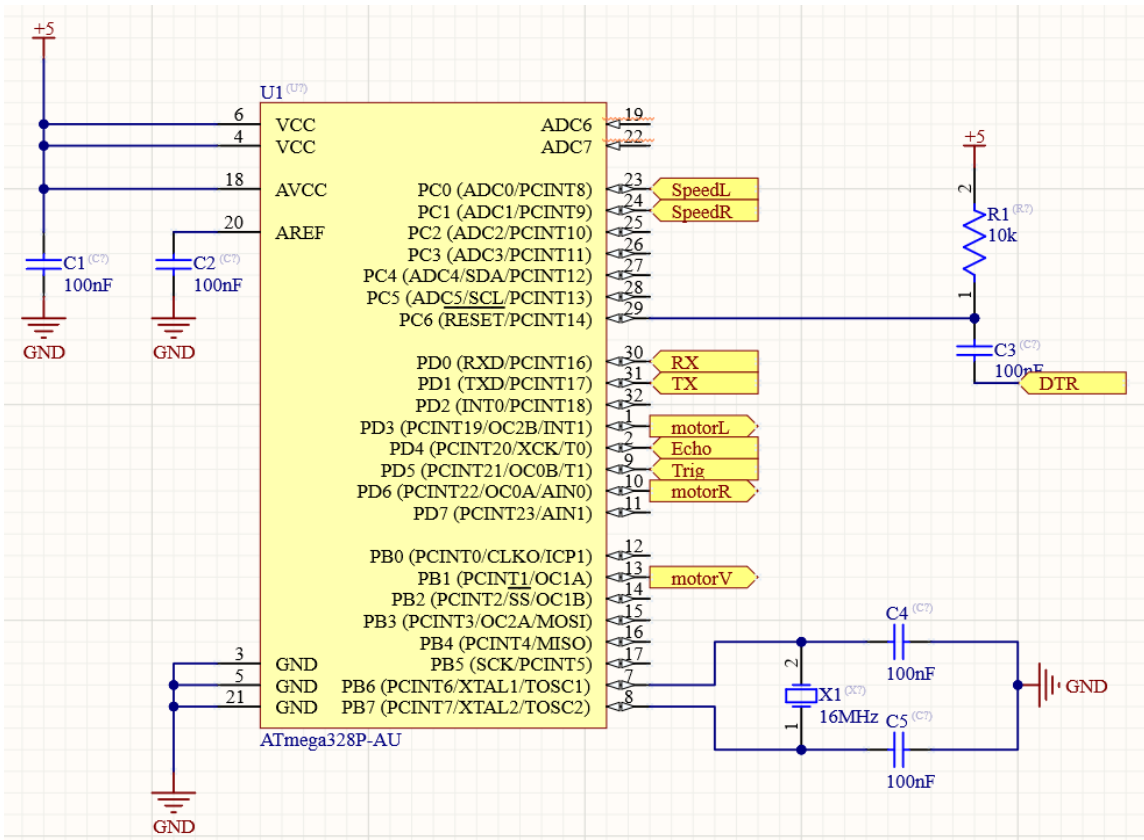

Figure 4. ATMega328P Microcontroller Subcircuit Schematic

3.6.2 Connectors

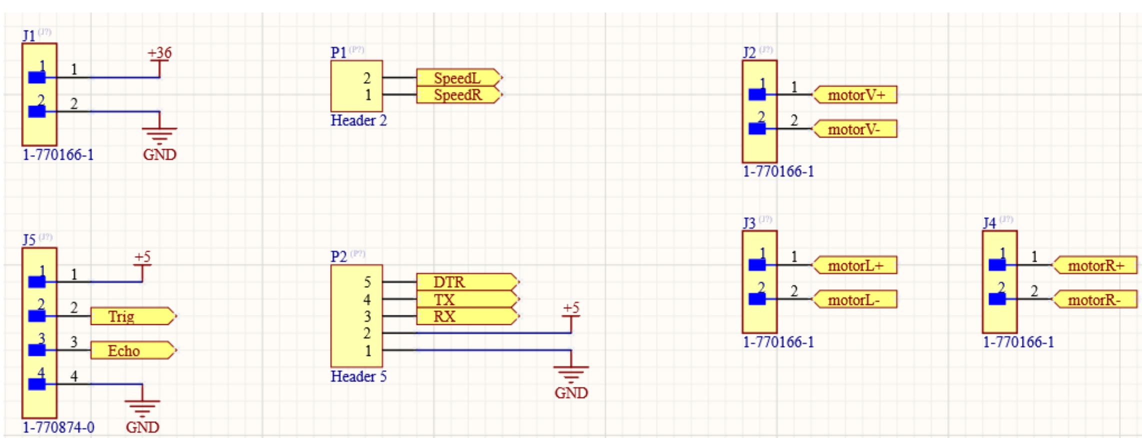

Figure 5. Connectors Schematic

3.6.3 Traction Motors

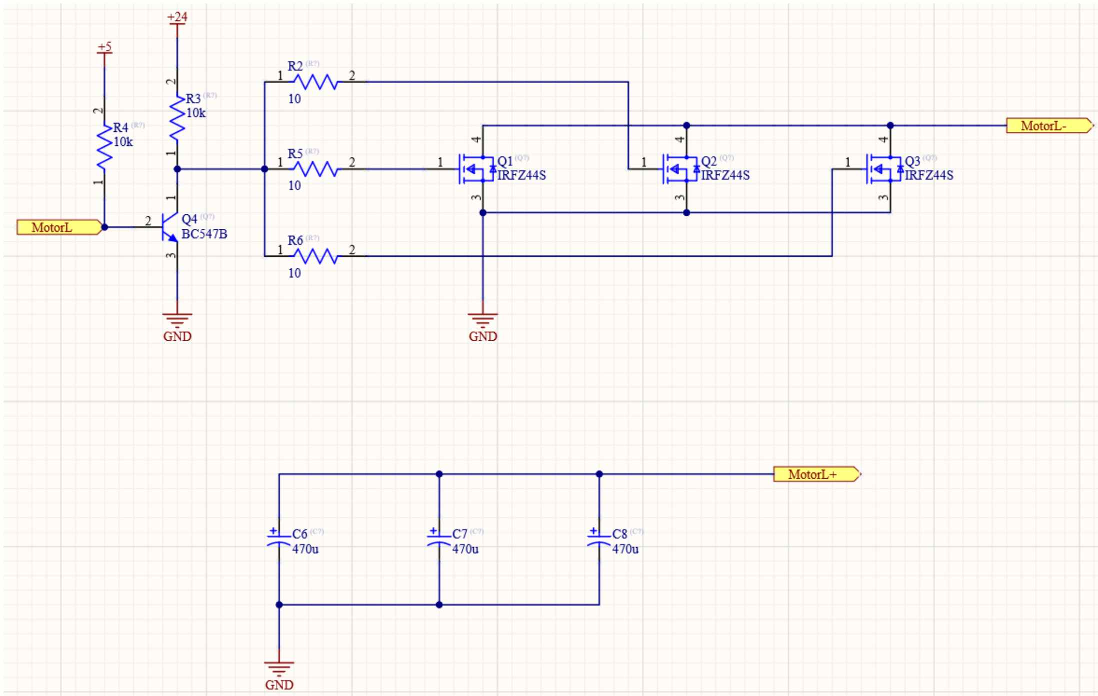

Figure 6. Left Traction Motor Schematic

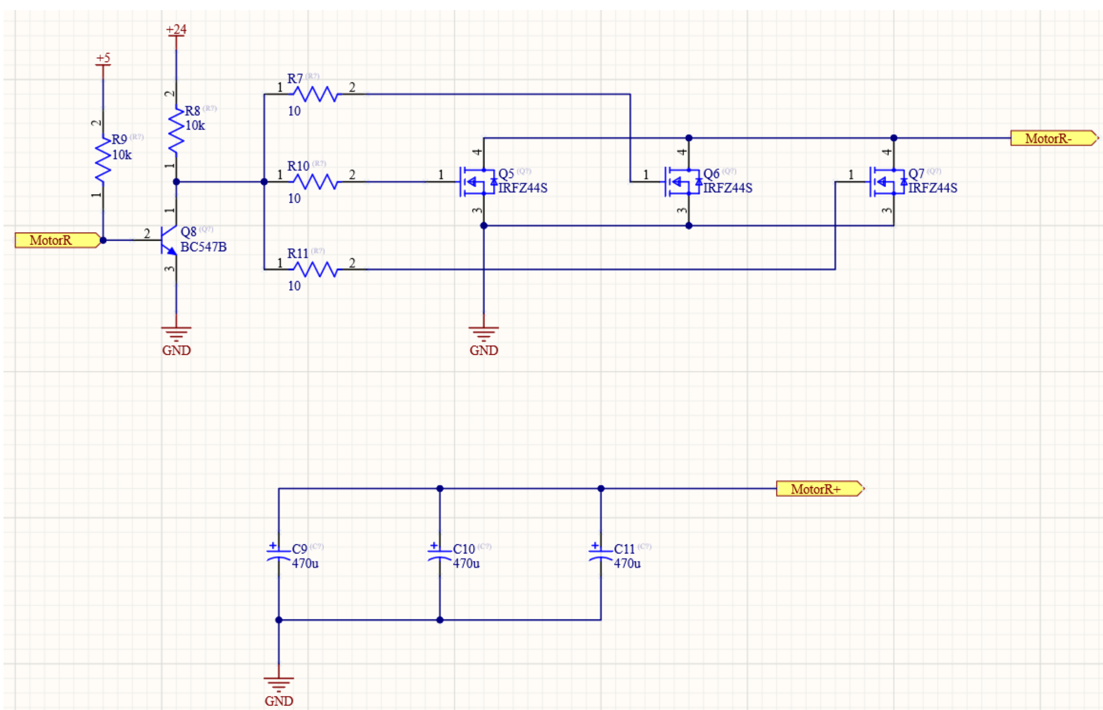

Figure 7. Right Traction Motor Schematic

3.6.4 Vibration Motor

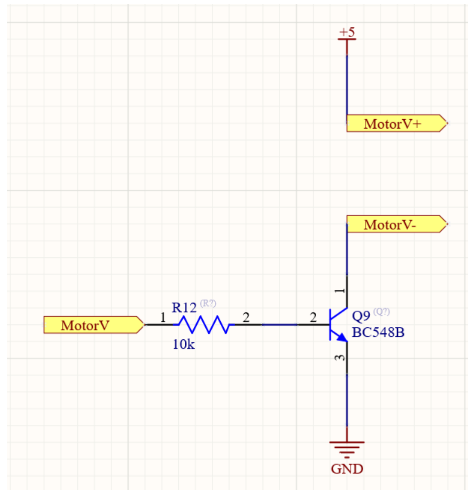

Figure 8. Vibration Motor Schematic

3.6.5 Power Distribution

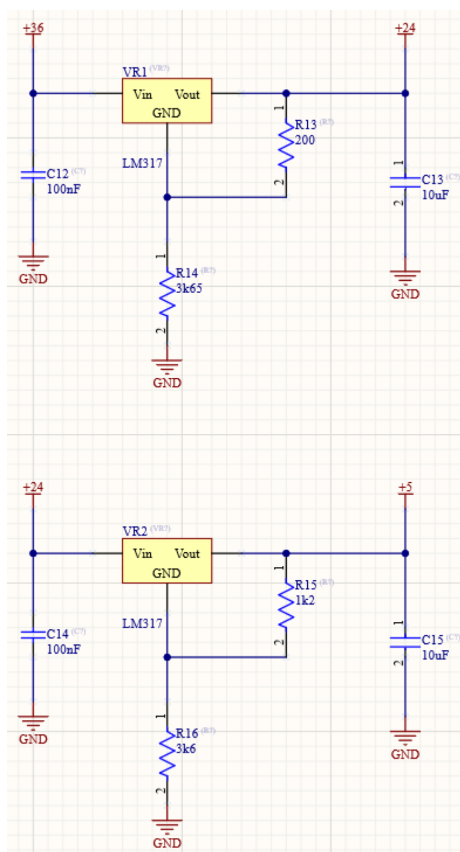

Figure 9. Power Distribution Schematic

3.7 PCB Layout

3.7.1 Layout 2-D View

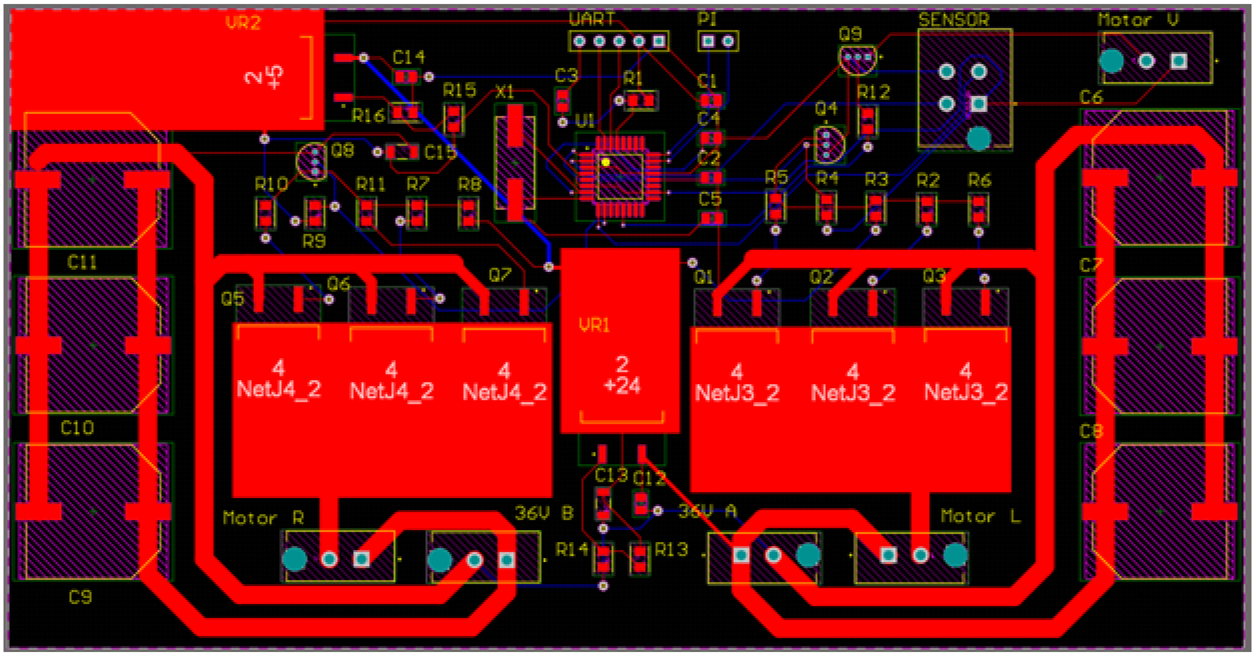

Figure 10. 2-D View Front Side

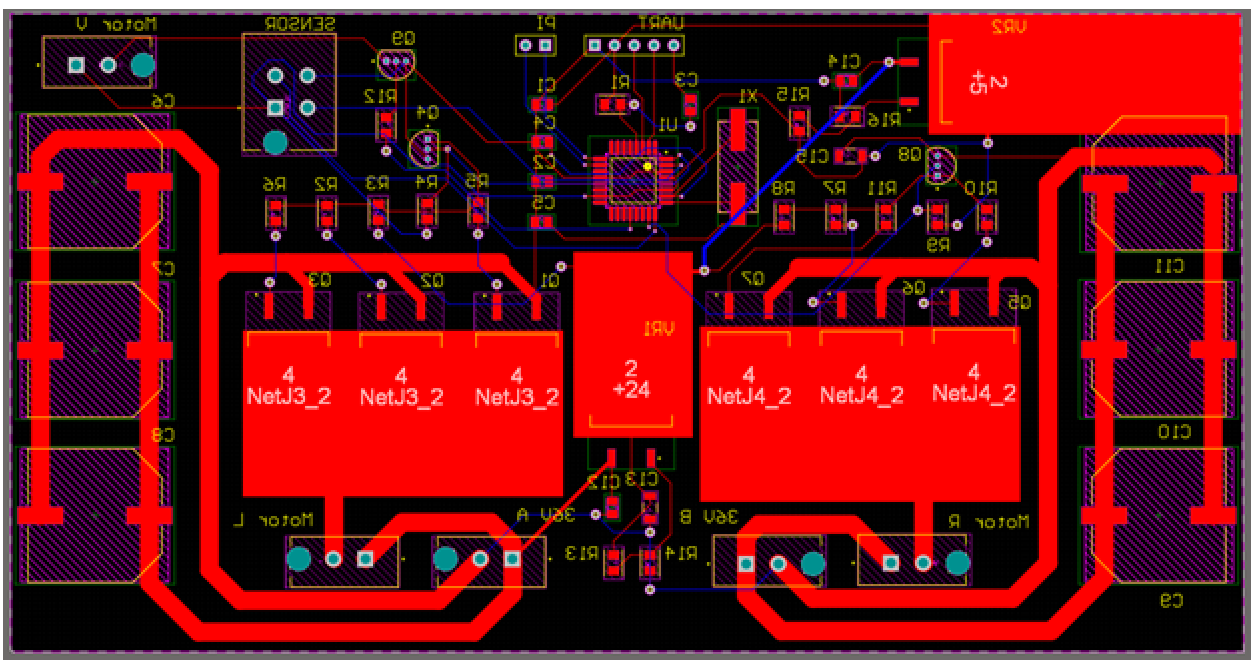

Figure 11. 2-D View Back Side

3.7.2 Layout 3-D View

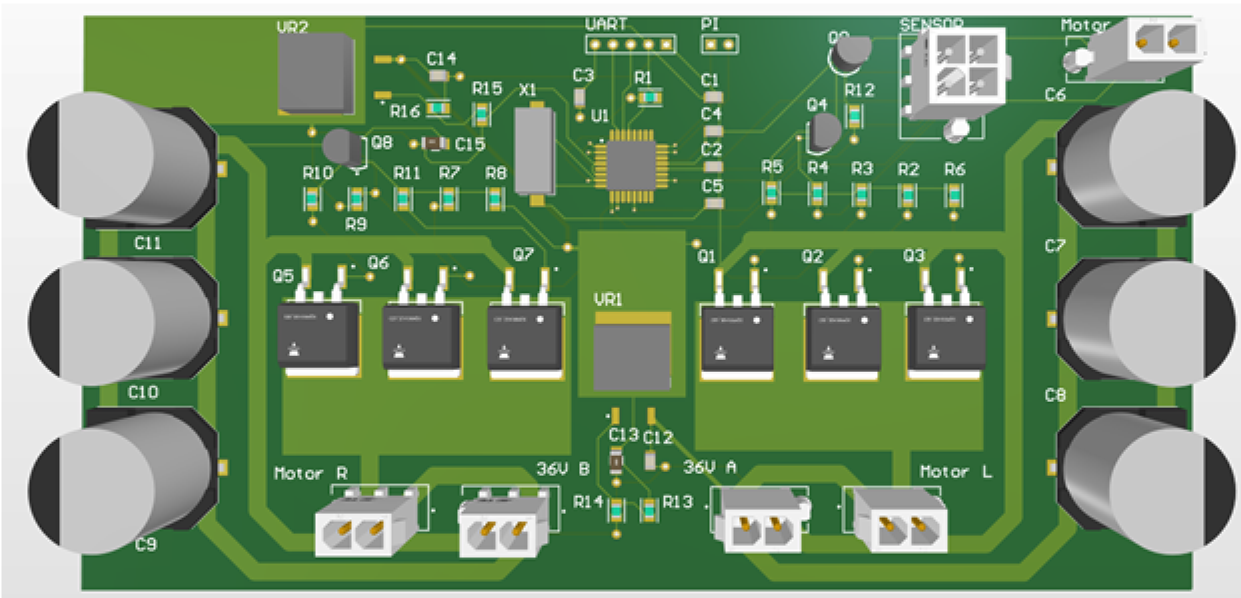

Figure 12. 3-D View Front Side

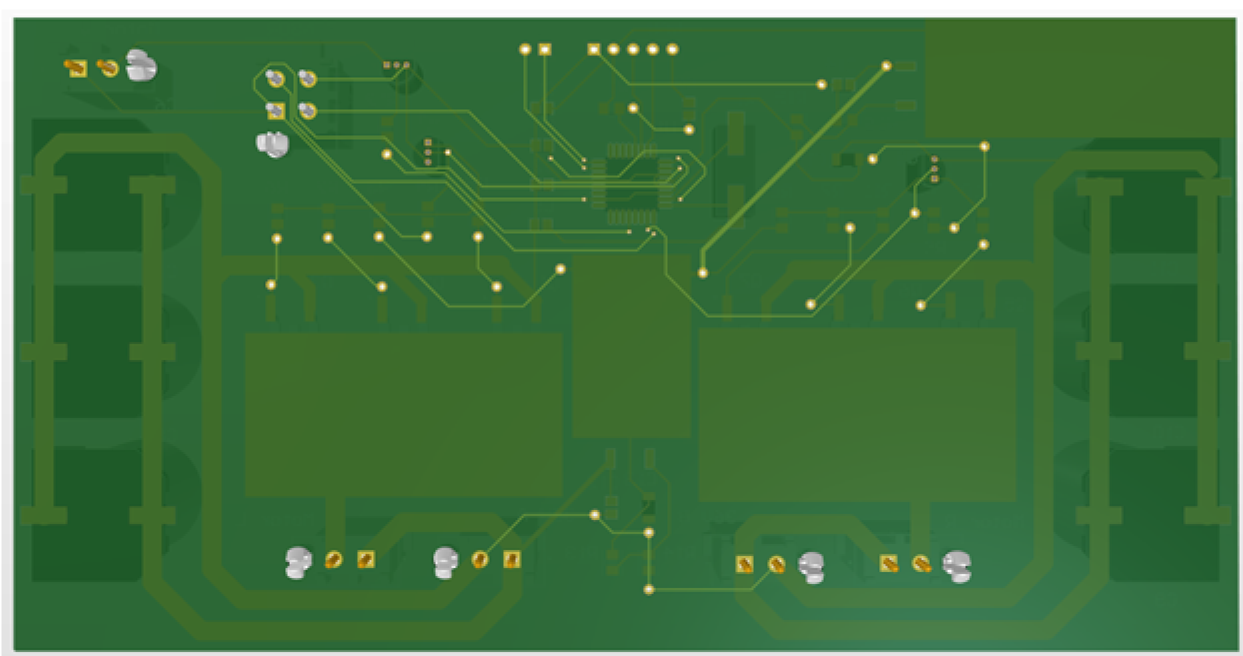

Figure 13. 3-D View Back Side

3.8 Software

3.8.1 Face Motion Detection Software

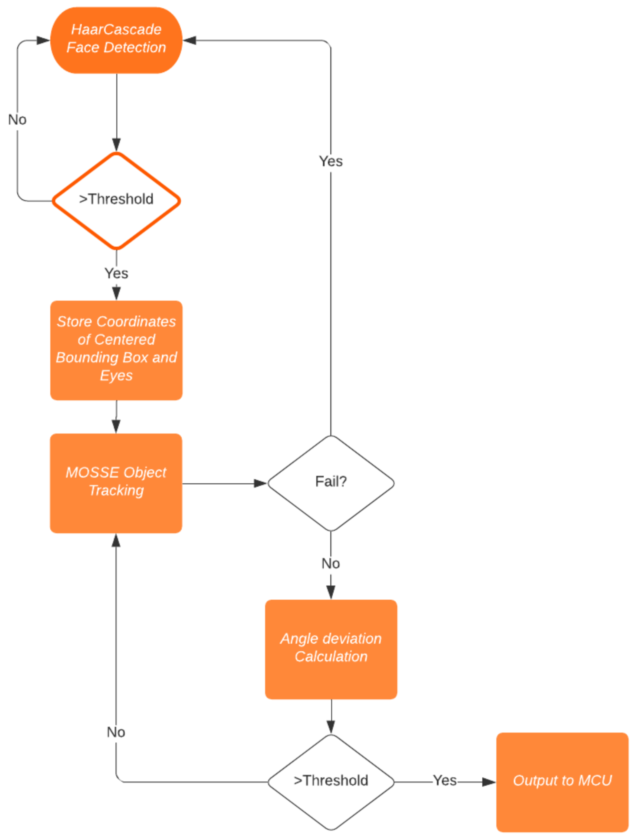

Figure 14. Face Motion Detection Software Flow Chart

The facial detection algorithm we'll be using is Haar Cascade classifier which has a frame rate of 9.25 fps and low false positives for a close up face frame. Even though Histogram of Graphs method has a higher accuracy overall, the fact that it is slower with a frame rate of 5.41 fps makes it practically more ineffective for the wheelchair [6]. To reduce the runtime, we will be running face detection only when required. The majority of the time the software will be performing face tracking to find the coordinates of the face on the frame. We will be using MOSSE object tracking which is the fastest object tracking algorithm with a frame rate of 669 fps. The speed of the face tracking algorithm is the most important part of the face motion tracking subsystem.

Requirement	Verification
Face motion recognition must operate with minimal lag (<1 sec)	1.Implement HOG, HaarCascade, and DNN face detection models 2. Identify and compare accuracies/ false positive rates 3. Pick model with accuracy greater than 95% and fastest speed

3.8.2 Motor Control Algorithm

The microcontroller reads the throttle signal between 0 to 5 volts from an analog input pin for each motor. When the throttle is high, the microcontroller outputs a low voltage to the base of the BJT. Since we use an npn BJT, the BJT is now turned off. Then the gates of the power mosfets are pulled up to 24V, so the mosfets are on. 36V of voltage is supplied to the motors and turns them on. When the throttle signal is low, the microcontroller outputs a high signal to the base of the BJT to turn it on. Then the gates of the power mosfets are grounded, so the mosfets are off. The V- of the motors are not connected to ground and they are turned off in this way. The speed of the motors can be controlled by adjusting the signal frequency at the base of the BJTs. The microcontroller has the ability to assign an arbitrary PWM frequency to its digital output pins.

3.8.3 Object Detection Algorithm

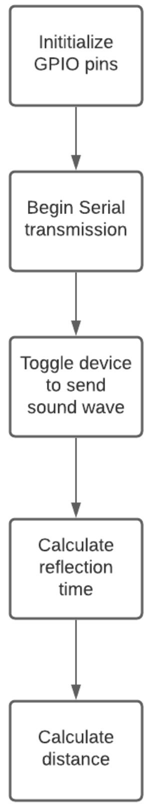

Figure 15. Object Detection Software Flow Chart

Once the distance has been calculated from the ultrasonic sensor. A vibration coefficient is calculated using the following formula:

$Vibration\ coefficient\ =\ distance/9000$

Based on the vibration coefficient, the proportionate analog output is sent to the vibration motor.

4. Tolerance Analysis

4.1 Latency Calculations

One Important tolerance to keep in mind is for the ultrasonic sensor. It should be able to detect objects with a tolerance of ~5cm. For this there are a few important things to look out for. Firstly, we need to ensure that the sensors are placed high enough so that they aren't affected by the distance to the ground. Secondly we would send test pulse signals with different ultrasonic sensors to see which one suits our application the best. The ultrasonic sensor works in the following steps:

- High level signal is sent for 10us using Trigger.
- The module sends eight 40 KHz signals automatically, and then detects whether pulse is received or not.
- If the signal is received, then it is through a high level. The time of high duration is the time gap between sending and receiving the signal.

The distance (in cm) is calculated with the following formula:

$distance = time \cdot 340/20000$

The limitation of the sensor allows us to calculate the distance to a maximum of 10 meters - out of which upto ~8 meters is accurate.

The head motion detection subsystem needs to be almost instantaneous (<1sec). To achieve this we will be avoiding face detection as much as possible and be using MOSSE face tracking which is one of the fastest object tracking algorithms. MOSSE tracking operates at 669 frames per second and tolerates changes in lighting, pose and scale, which is also extremely important for the system. Initial tests have proven that we can run the detection natively on the raspberry pi with a framerate of 10-20 FPS. This would be ideal for us but we can also add more complexity to reduce the number of false positives and get the framerate down till 5 FPS.

4.2 Torque Calculations

In this section, we approximate the wheel diameter to be 0.6m and speed to be 0.5m/s.

We start from:

$Power\ (W)\ =\ Force\ (N)\ \cdot\ Velocity\ (m/s)$

$Torque\ (N\ \cdot\ m)\ =\ Force\ (N)\ \cdot\ Radius\ (m)$

$Velocity\ (rpm)\ =\ Velocity\ (m/s)\ /(2\pi\ \cdot\ Radius\ (m))\ \cdot\ 60$

So that:

$Power\ (W)\ =\ Force\ (N)\ \cdot\ Velocity\ (rpm)\ \cdot\ (2\pi\ \cdot\ Radius\ (m))\ \cdot\ 60$

$\qquad\qquad =\ Torque\ (N\ \cdot\ m)\ \cdot\ Velocity\ (rpm)\ \cdot\ (2\pi/60)$

We have two motors of 500W. This adds up to 1000W.

0.5m/s with wheels of 0.6m in diameter is 16rpm.

$1000W\ =\ Torque\ (N\ \cdot\ m)\ \cdot\ 16\ rpm\ \cdot\ (2\pi/60)$

$Torque\ =\ 600\ N\ \cdot\ m$

For a wheel of diameter 0.6m and radius 0.3m, the force on the wheels is $600Nm/0.3m\ =\ 2000N$. Assume a 50% overall efficiency, the force on the wheels is still 1000N and 100kg. This is enough no matter how much the user weights. Assume a gear ratio of 10:1 (wheelchair wheel to friction wheel). This means that our friction wheels and thus our motors are moving at $16\ rpm\ \cdot\ 10\ =\ 160\ rpm$. The maximum rpm of our motors is 2800 rpm, which is much larger than 160rpm.

5. Schedule

Week	Arnav	Dev	Jiayuan
2/22/20	Find components for motion detection subsystem and appropriate wheelchair.	Complete raspberry pi and camera interface	Find the appropriate motor, battery pack, and circuit components.
3/1/20	Finish design for object detection system.	Implement HaarCascade face detection and perform quantitative analysis on accuracy and speed	Finish the design of the first version of schematic
3/8/20	Work on algorithm for object detection and feedback	Implement HOG facial detection and perform quantitative analysis on accuracy and speed	Start the design of the first version PCB layout and place the order of circuit components
3/15/20	Build test circuit to test ultrasonic sensors and ensure that they are accurate for given limitations	Implement DNN facial detection and perform quantitative analysis on accuracy and speed	Finish the design of the first version PCB layout and place the order of the PCB
3/22/20	Connect and calibrate vibration motors with ultrasonic sensors.	Implement MOSSE object tracking and perform quantitative analysis on accuracy and speed	Solder and test the first version of design
3/29/20	Test System with wheelchair mounting	Implement KCF object tracking and compare results with MOSSE to find best fit	Redesign to solve all the problems in the testing and place the new order
4/5/20	Integrate and calibrate system with wheelchair	Implement state machine for face detection and tracking	Solder and test the second version of design
4/12/20	Integrate motor system with wheelchair	Perform functional tests on wheelchair to find optimum threshold values in natural environment	Integrate the electrical system to the wheelchair

4/19/20	Prepare for demonstration	Prepare for demonstration	Prepare for demonstration
4/26/20	Begin final report	Begin final report	Begin final report
5/3/20	Finish final report	Finish final report	Finish final report

6. Cost Analysis

Assume that the labor cost for each member in our group is $30/hour and assume that each member works 15 hours per week. We have 3 members in our group and 16 weeks this semester to complete this project.The total labor cost is calculated as:

$(\$30/hour) \cdot (15\ hours/week/member) \cdot (3\ members) \cdot (16\ weeks) \cdot 2.5 = \$54000$

Description	Manufacturer	Part Number	Quantity	Cost/Unit	Total
Wheelchair	Walmart		1	$150	$150
Brushed DC Motor	Vevor	500W 36V	2	$60	$120
40kHz Ultrasonic Sensor	Adafruit Industries LLC	3942	2	$3	$6
USB to TTL Serial Converter	SongHe	FT232RL	1	$10	$10
8-bit Microcontroller	Microchip	ATMEGA328P-AU	6	$2	$12
Camera Module	Sony	DEV-14028	1	$25	$25
Li-ion Battery Pack	Okoman	36V 30Ah	1	$110	$110
Assorted resistors capacitors, mosfets, bjts, crystals, linear regulators, connectors	Mouser	/	/	/	$30
PCBs	JLCPCB	/	10	$2	$20
Machine Shop Service	/	/	2 weeks	/	/
				Total	$483

7. Ethics and Safety

Since we will be using facial detection, a big ethical concern for the users of the product would be that none of their data is stored. We will not be storing any frames captured by the camera on an external server. We will also not be performing any facial recognition.
Since the wheelchair will be operated on head-motion as input, causing the wheelchair to move, safety is a priority for the product. The biggest safety concern is false positive input, that is, the wheelchair moving without the user intending it to. To ensure this does not happen, we will be designing the forward movement command to be something that cannot be provided in an unexpected way such as a specific facial expression.

Another concern is the instantaneous halting of the wheelchair when required. To achieve this we must ensure that the facial detection is fast and that there is low latency between the detection subsystem and motor controller. An important feature we are adding is an emergency switch to turn the wheelchair on and off. This would allow the user to halt the wheelchair at any time which would add another layer of safety. This would also allow the user to switch the wheelchair off when not in use to conserve energy.

Besides, the head-motion controlled wheelchair is a project consisting of both high voltage and low voltage electrical systems. Generally, 30 volts is considered as a conservative threshold value for dangerous voltage [8], which means any system above 30V could be dangerous to humans. To avoid any hazard brought by high voltage, we first need to refuse any exposed high voltage anodes and cathodes. This includes the high voltage battery output and the three phase power output of the motor controllers.
Second, we need to set apart the high voltage and low voltage circuits on the PCB. Since both systems are existing on a single board, we need to avoid the low voltage system to be shorted with anything above 5V, especially the high voltage system which may go over 30V.

References

[1] ArduCam, ‘Raspberry Pi MIPI CSI Camera Pinout’, 2014. [Online]. Available: https://www.arducam.com/raspberry-pi-camera-pinout/. [Accessed: 18-Feb-2021].

[2] G. Behera. “Face Detection with Haar Cascade”, 2020. [Online]. Available: https://towardsdatascience.com/face-detection-with-haar-cascade-727f68dafd08. [Accessed 18-Feb-2021].

[3] Bittele Electronics, ‘PCB Trace Width Calculator’, 2021. [Online]. Available: https://www.7pcb.com/trace-width-calculator.php. [Accessed: 18-Feb-2021].

[4] JST, ‘Wire Size & Current Rating (A) Guide’, 2021. [Online]. Available: https://www.jst.fr/doc/jst/pdf/current_rating.pdf. [Accessed: 18-Feb-2021].

[5] JST, ‘Wire Size & Current Rating (A) Guide’, 2021. [Online]. Available: https://www.jst.fr/doc/jst/pdf/current_rating.pdf. [Accessed: 18-Feb-2021].

[6] Towards Data Science, ‘Face Detection Models: Which To Use And Why’, 2020. [Online]. Available: https://towardsdatascience.com/face-detection-models-which-to-use-and-why-d263e82c302c. [Accessed: 2-March-2021]

[7] Murata, ‘Ultrasonic Sensors FAQ’, 2021. [Online]. Available: https://www.murata.com/en-us/support/faqs/products/sensor/ultrasonic/char/0002. [Accessed: 18-Feb-2021].

[8] All About Circuit, ‘Ohm’s Law’, 2020. [Online]. Available: https://www.allaboutcircuits.com/textbook/direct-current/chpt-3/ohms-law-again/. [Accessed: 18-Feb-2021].

[9] Texas Instruments, ‘LM317 3-Terminal Adjustable Regulator’, 1997. [Online]. Available: https://www.ti.com/lit/ds/symlink/lm317.pdf. [Accessed: 4-March-2021]

[10] Vishay, ‘Power Mosfet’, 2011. [Online]. Available: https://www.vishay.com/docs/91293/91293.pdf. [Accessed: 4-March-2021]

[11] Fairchild Semiconductor, ‘BC546/547/548/549/550’, 2002. [Online]. Available: https://www.sparkfun.com/datasheets/Components/BC546.pdf. [Accessed: 4-March-2021]

[12] Alan Mitchell, ‘Operating an Arduino for a Year from Batteries’, 2011. [Online]. Available: https://alanbmitchell.wordpress.com/2011/10/02/operate-arduino-for-year-from-batteries/. [Accessed: 3-April-2021]