FUN-E-MOUSE

Team 50: Chuangy Zhang, Yingyu Zhang, John Casey

ECE 445 Design Document - Spring 2022

TA: Jamie Xu

Mar 5th, 2022

Table of Contents

1. Introduction

1.1 Problem and Solution Overview

Cats found all small objects that move in an irregular way captivating due to their chase instinct. Providing support for their instinct can help maintain the cats' well-being. The provision of playthings is most cat owners' choice. According to the Journal of Veterinary Behavior, the most common provision of toys and activities used by cat owners in a survey of 277 cat owners were 64% of furry mice, 62% of catnip toys, and 62% of balls with bells [1]. Having fast moving furry mice occur in the vicinity of the cat will induce domestic cats to exercise, as well as provide environmental enrichment for them[2]. A remote-controlled FUN-E-MOUSE can be a solution that allows more cat owners to get interactive playtime with their pets. It can also facilitate the relationship between cat and cat owner.

Many existing electronic cat toys have preset modes that do not allow human control leaving the owner feeling not as involved in playing with their cat. The cat toys that do allow humans to play with their animals directly are not as accessible to those who may be wheelchair-bound or have disabilities. Current battery-powered cat toys on the market are rarely rechargeable leaving owners to purchase and go through lots of batteries. The use of remote controllers for some of these toys leaves them useless when the controller is lost or broken. Cheap motors or tires used on these devices often leave them struggling to move on certain floor materials like carpets, limiting many cat owners.

Cat owners need a solution in the form of a smartphone remote-controlled mouse that their pets can chase around for hours. A strong lithium-ion battery will allow easy rechargeability and allow longer playtime than current toys on the market. For safety measures of the house and pet, sensors will be used to alert the toy to stop when approaching walls or when attacked by a pet. To allow use in any household flooring the mouse will be equipped with gripped rubber tires powered by strong motors to ensure it can transition from different flooring materials.

SUMMARY OF THE INNOVATION	
Goals	• Solve the shortcomings of electronic-mouses that are out in the market (including battery endurance is very low, for hard floors only, and cannot be control remotely)
Benefits	• Fulfill the need for cats' hunting behaviors

	• Create a high-quality interactive cat toy with adaptive features including smartphone remote control to allow greater usability for people with disabilities or injuries. • Provide a fun safe way to engage your cat's inner prey drive, by letting them hunt and chase a toy that will go dormant when they pounce on it.
Features	• A On-Off switch • Two Operating Modes: 1. Auto Driving 2. Remote Control • A micro-USB port for charging • A web based application
Functions	• Navigates around objects, corners and under furnitures • Works on any home flooring surface: tile, wood, carpet, concrete,..etc. • Attractive appearance for cat (with feathers attached to the tail's end and LED eyes) • Rechargeable with a built-in Micro-USB charging port connected to a lithium-ion battery that can last 30+ min with one charge • With the AUTO mode on, the mouse can go around on its own and goes dormant for five seconds when an obstacle is within the range of 10 cm x 10 cm, changes direction after reactivates, repeatedly until it is being turned off or the battery dies • With the REMOTE mode on, the mouse can be accessible via smartphone

1.2 High-level Requirements list

1. **Remote Control via Smartphones:** With the REMOTE mode on, the owner can remotely control the direction of the mouse via a web app from at most 15 feet away.
2. **Rechargeable Battery:** This device is rechargeable through the build-in micro-USB port and able to supply at least 30 minutes in one charge.
3. **Self Obstacle Avoiding:** The mouse can automatically sense and make a detour of obstacles on its path without running straight into it when in AUTO mode.

1.3 Visual Aid

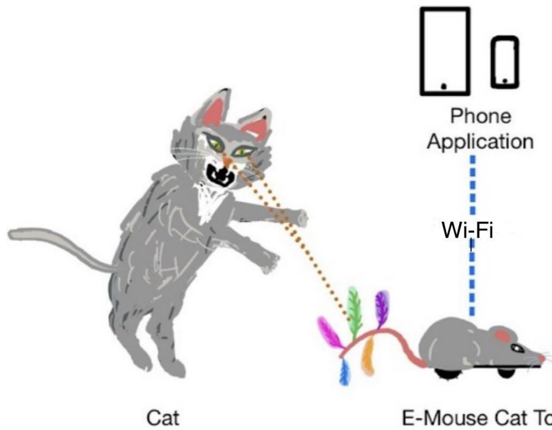


Figure 1.3 Visual Aid of the E-Mouse Cat Toy

2. Design

2.1 Block Diagram

The Fun-E-Mouse project has five major subsystems: power management system, control unit system, drive system, sensory system, and network system. The power management system has a battery charger, a lithium-ion rechargeable battery, and a voltage regulator. The power management system allows us to provide a stable 3.3V supply to all other four subsystems, it also allows us to recharge the lithium-ion battery. The control unit system has an ESP32 MCU designed by the Espressif System, it is responsible to process all the data coming from the sensory system and network system. The control unit system controls the drive system, and that allows the Fun-E-Mouse to drive smoothly and avoid any obstacles. The sensory subsystem consists of four ultrasonic sensors and two LEDs. The four ultrasonic sensors are located at the front, right, back, and left respectively. This allows the Fun-E-Mouse to detect any obstacles nearby and send the data to the control unit system. The network system has a Wi-Fi module, RESTful APIs, and an App interface(IOS/Android/Web). The network system allows us to control the Fun-E-Mouse remotely from a long distance, and we also can configure different play modes remotely.

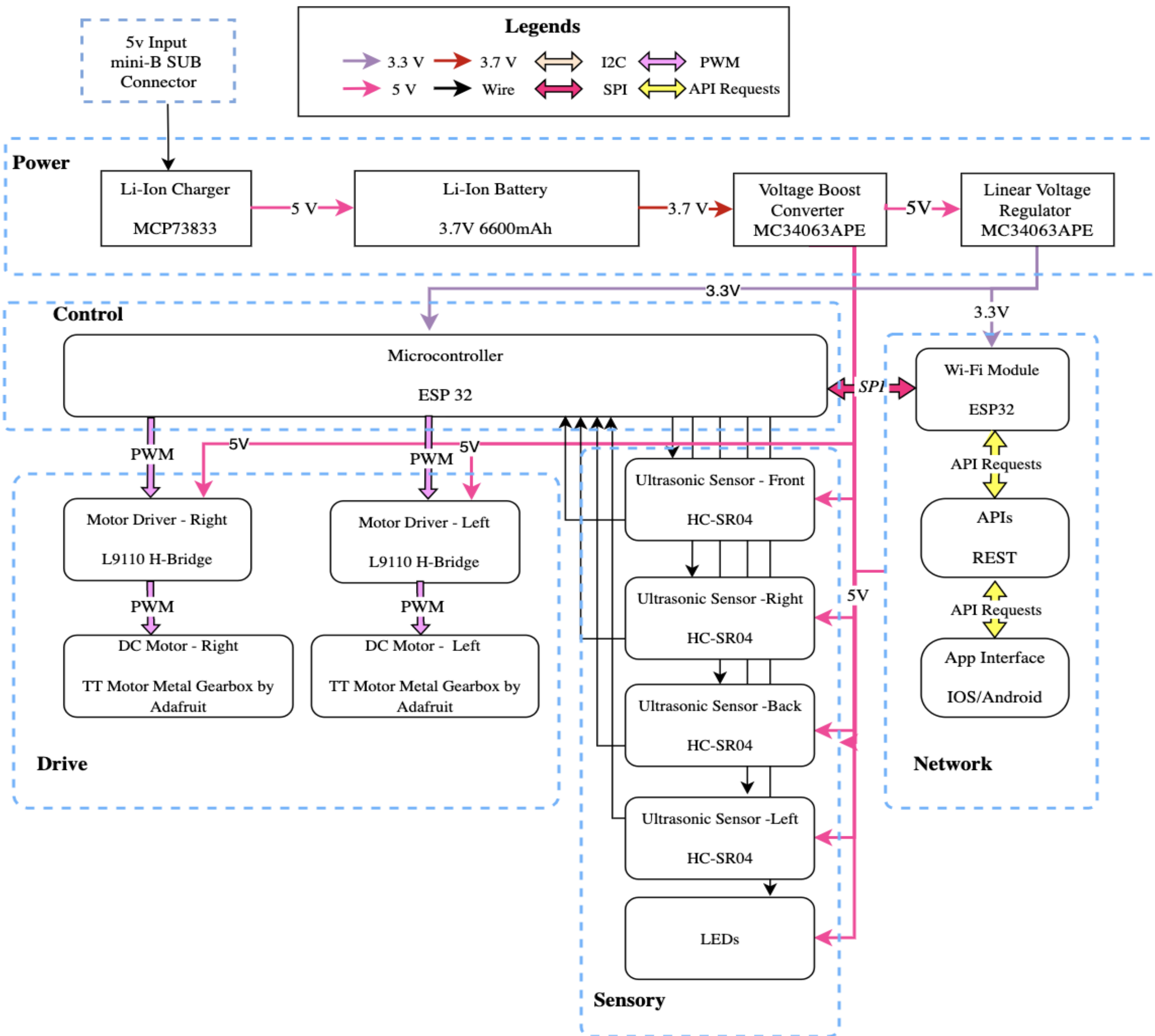


Figure 2.1 Block Diagram of the Fun-E-Mouse

2.2 Physical Design

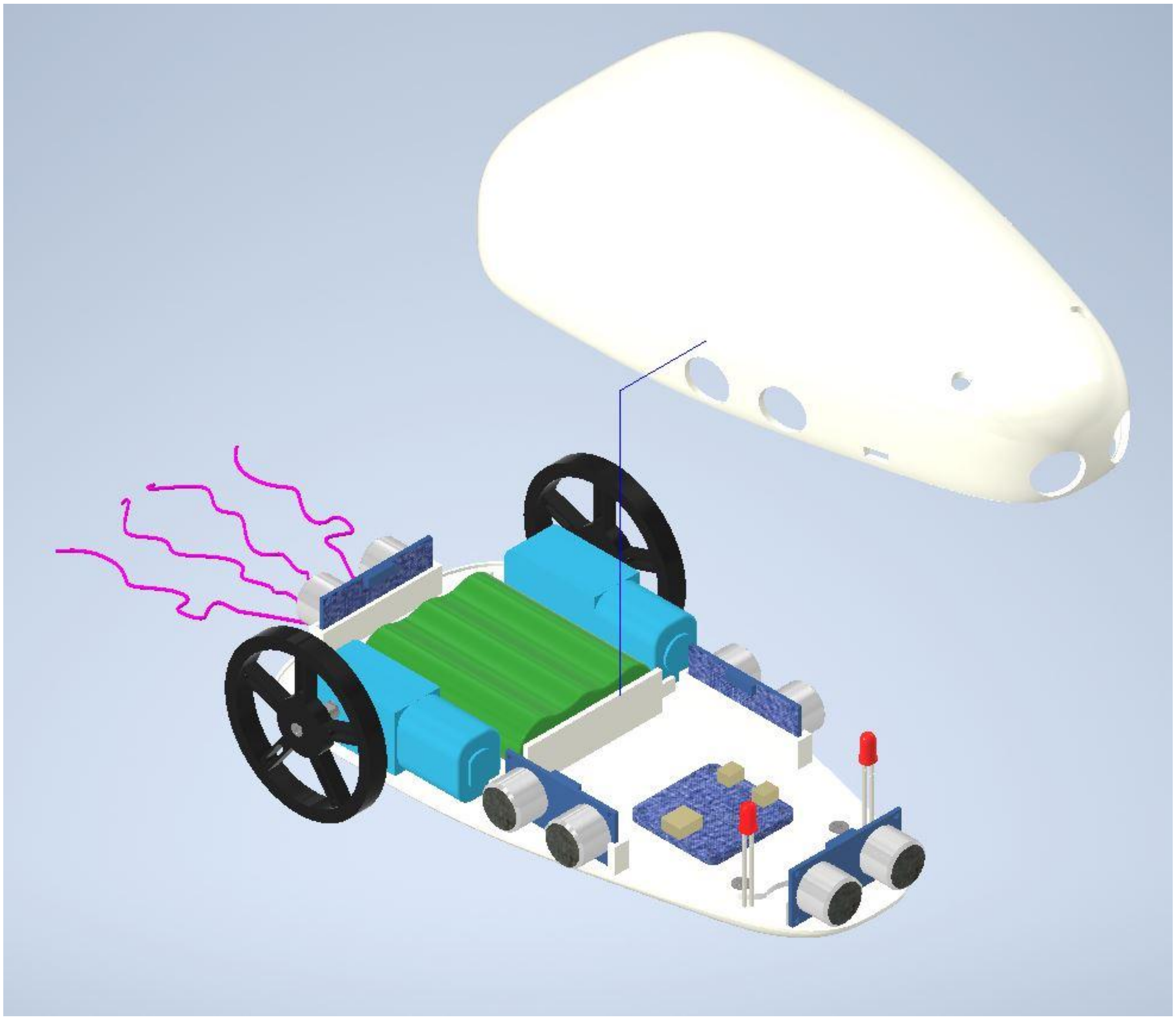


Figure 2.2 Physical Design of Fun-E-Mouse

We will 3D print the shape of the mouse in two parts, the top shell and bottom baseplate. We will have the sensors interlock onto the shell of the mouse; the PCB, motors, and battery mount on the baseboard of the mouse inside the "stomach". This shape encloses all the components of the electronic mouse and has reserved openings for the sensors and LEDs, as well as a micro-USB charging port. An extensive amount of work has gone into the 3D design based on the assumption of dimensions found on datasheets or websites for the parts we have ordered. Due to fluctuations in part size, measurements and fitments will need to be tested to ensure proper sizing of the 3D printed shell. There will be some feathers and strings attached to the tail of the electronic mouse to make it attractive for cats. We will make sure the material we use is strong and safe enough for cats.

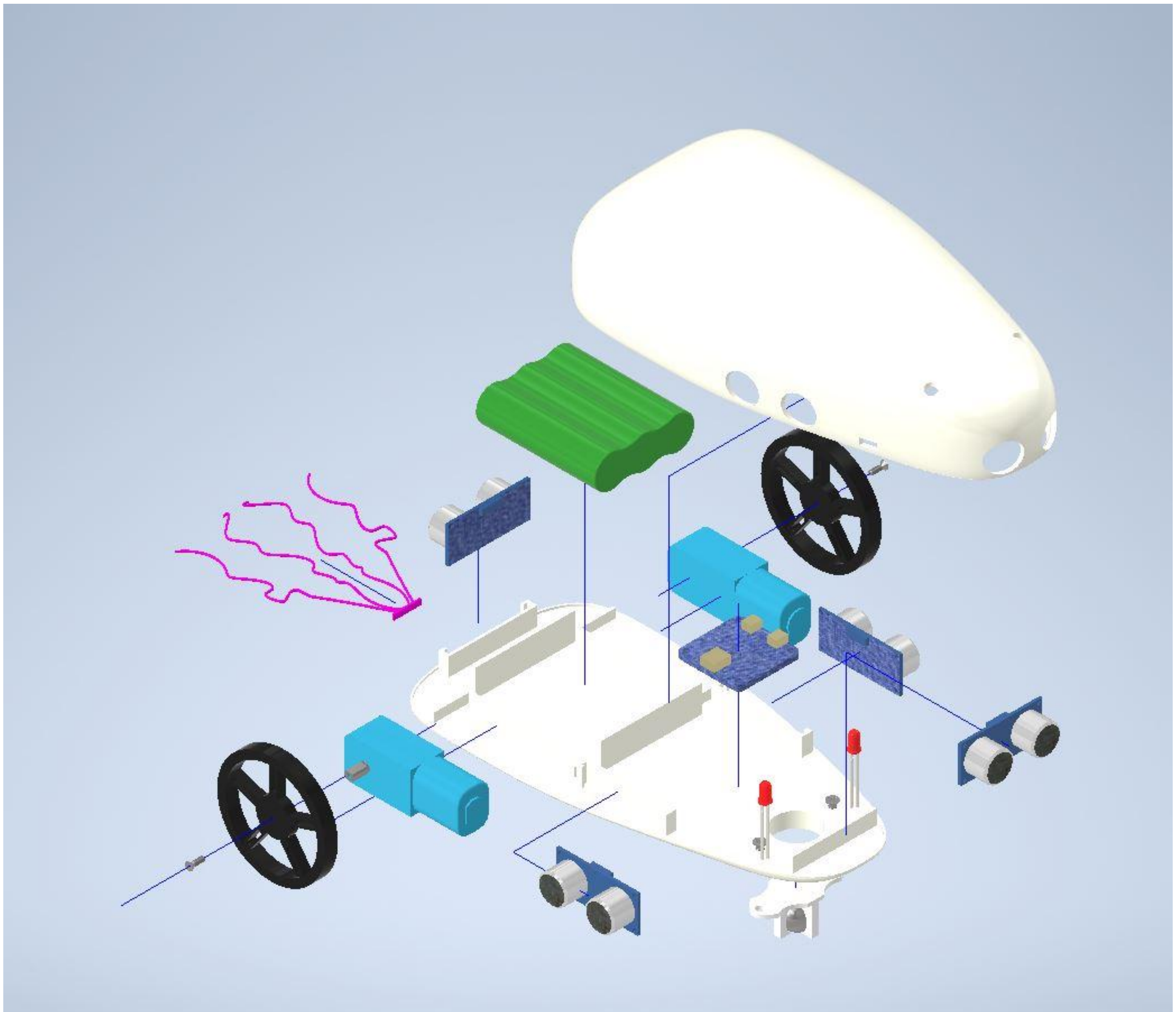


Figure 2.2.1 Physical Design of Fun-E-Mouse

2.3 Functional Overview & Block Diagram Requirements

2.3.1 Power Management Subsystem

The power management subsystem consists of a 3.7v Polymer Li-ion Battery with 6600mAh by Adafruit, a Lithium Polymer battery charger based on the MCP73833, and a 3.3V linear voltage regulator that can provide up to 250mA, and a voltage boost converter that can output 5V. The power management subsystem allows us to provide a stable 3.3V output that powers up all other four subsystems. The lithium-polymer battery charger uses a USB mini-B interface to charge the battery. That can come from any computer or any USB wall adapter. The fast-charge current is 500mA by default, but we can easily adjust the charge current from 100mA to 1000 mA. The lithium-polymer battery is made of 3 balanced 2200mAh cells. Each cell can provide 1.1A, therefore, the peak current that we can draw is 3.3A which allows us to power the Fun-E-Mouse for more than 30 minutes. The voltage regulator that we chose to use is the LV1117-33 because this linear voltage regulator can provide a nice clean 3.3V with 2% regulation. It is perfect to power up the microcontroller, and the 5V output will power up the other subsystems.

Requirement(s)	Verification
1. Provide a nice clean 3.3V and 5V output voltages with 5% regulation 2. Able to recharge the battery from Computer, Wall USB adaptor, or power bank 3.Able to power the Fun-E-Mouse at least 30 minutes of continuously running	1A. Connect a 10K resistor as load at the output of 3.3V and 5V 1B. Use a multimeter to check if the voltage and current meet the requirement 1C.Use an oscilloscope to check if the voltage signal is clean and within 5% regulation 2A. Plug the charging module via any micro USB cable into a computer, wall USB adaptors, or power bank 2B. Check if the Charging LED on the PCB board is on 2C. Leave the battery charging overnight, and check if the Done LED on the PCB board is on 3A. Once the Fun-E-Mouse Project is done, we will leave the mouse run continuously and record the time 3B. Ensure the the Fun-E-Mouse meet the time requirement

The following schematic is the power management subsystem:

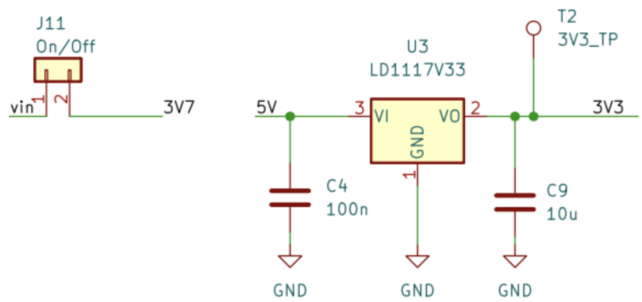


Figure 2.3.1.1 Linear Voltage Regulator

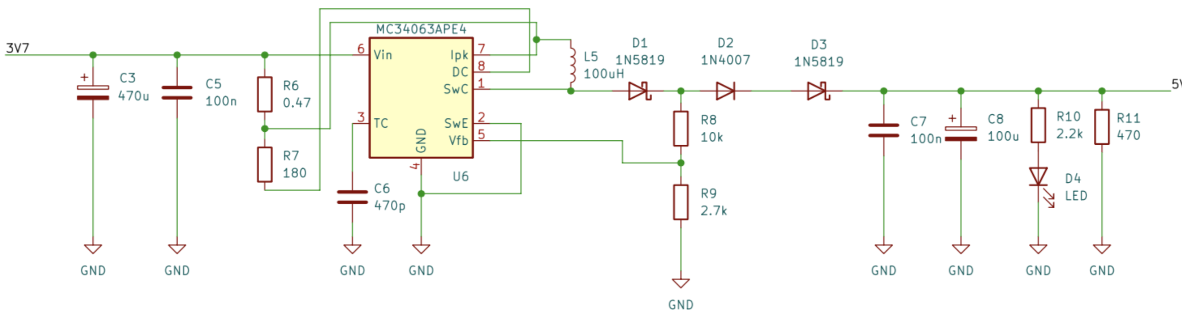


Figure 2.3.1.2 Voltage Boost Converter

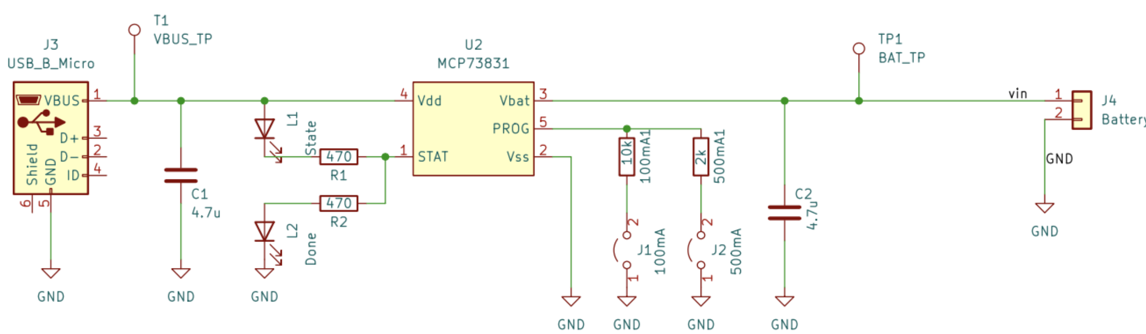


Figure 2.3.1.3 Li-Ion Charger

2.3.2 Control Subsystem

The control unit subsystem has an ESP32 MCU designed by the Espressif System. ESP32 has been widely used in many IoT devices, and it is well documented and easy to develop and program. The ESP32 is a low-cost, low-power system on chip with Wi-Fi and Bluetooth capabilities. It also supports PWM, I2C, and SPI communication protocols. The control unit subsystem sends commands to the drive subsystem through PWM communication protocols. This allows us to drive Fun-E-Mouse left, right, backward, or forward. We also can control the speed of the Fun-E-Mouse as well. The control unit subsystem is also responsible for receiving data from the networking subsystem, and that allows us to control the Fun-E-Mouse within 15 feet distance. In addition, the controller is powered by 3.3V power from the voltage regulator of the power management subsystem.

Requirement(s)	Verification
1. Able to maintain a stable Wi-Fi connection of at least 15 feet 2. The rate of RESTful Request is at least 10-30 per second 3. After mounted to PCB, the microcontroller should be program, and it should able to transmit data at baud rate of 115200	1A. Set the microcontroller 15 feet away from the Wi-Fi router 1B. Send a control command through RESTful API request 1C. Make sure the microcontroller is performing the right command 2A. Set counter variable in the program that keeps track of how many numbers of the requests 2B. Continuously to send requests in 1 second 2C. Ensure the total number of requests is within the range of 10 - 30 3A. Connect the USB2UART program breakout board to the Tx, Rx, and GND pin 3B. Run the Wi-Fi Unit test code via Arduino IDE, and burn bootloader, The program should able to read the data output at a rate of 115200

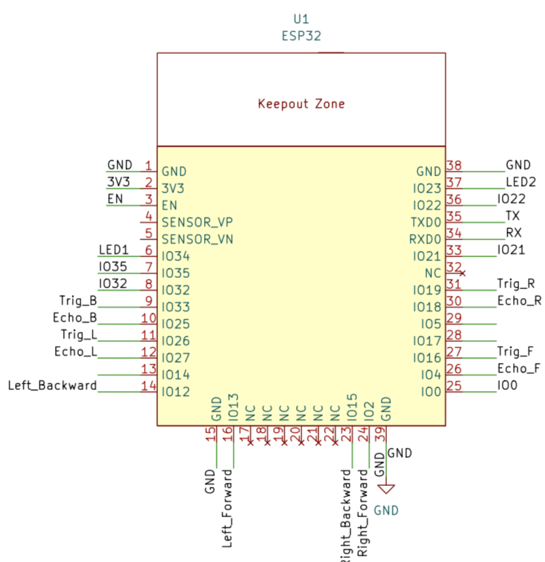


Figure 2.3.2.1 Esp32 Schematic

2.3.3 Sensory Subsystem

The sensory unit consists of four Ultrasonic Sensors and two LEDs. Ultrasonic Ranging Module HC-SR04 was chosen due to its feature of having a 2cm - 400cm non-contact sensing range. They require low power which is suitable for the mouse since it is a battery-powered device. Each side (front, back, left, and right) of the mouse will have one ultrasonic sensor. When one of the sensors detects an obstacle, the mouse will move away from it. To be further specific, the Echo pins of the ultrasonic sensors will go low when the pulses sent out by the sensors are reflected back. The output of the Echo pins will go into the microcontroller as inputs.

Requirements	Verification
1. Able to detect a chair leg or water bottle within the range of 2cm - 50cm in front of the ultrasonic sensor	1A. Place a chair or a water bottle in front of the ultrasonic ranging module in the range of 2cm - 50cm 1B. Write an unit testing program that output if the sensor detects an object 1C. Check the serial output, then see if the sensor detected the object
2. Able to detect a 4x4x10 cm cube within the range of 2cm -50cm in front of the ultrasonic sensor	2A. 3D printed a 4cm x 4cm x 10cm cube, and place the cube in front of the ultrasonic ranging module in the range of 2cm - 50cm 2B. Write an unit testing program that output if the sensor detects an object 2C. Check the serial output, then see if the sensor detected the object
3. Able to turn on or turn off the LED remotely at most 15 feet away	3A. Write an API request to turn on/off the LED 3B. Send a control command from the user interface at most 5 feet aways 3C. Check if the LED meet the corrected status

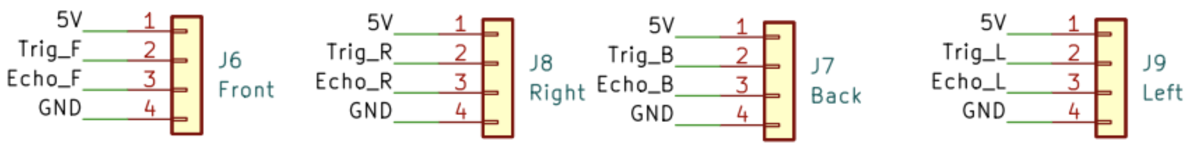


Figure 2.3.3.1 Ultrasonic Sensors Schematic

2.3.4 Drive Subsystem

The drive subsystem is to drive the Fun-E-Mouse to move forward, backward, rightward, or leftward. This subsystem has 2 L9110 H-bridge motor drivers for DC motors and 2 mental gear TT DC motors. The L9110 H-Bridge motor driver can direct a single DC motor bi-directionally, and it can provide a peak current of 800mA. It is good to drive any motor power voltages from 2.5v up to 12V. There's a PWM input that we can control the speed of the direction of the motor. The all-metal gear TT motors are an easy and low-cost way to get the Fun-E-Mouse to move on different surfaces. It has higher torque and slower rotational speed, and they provide more strength and less speed. It can operate from 3V to 6 V, and it can go faster at higher voltages. At 3V, it can operate 60 RPM with no load, and it only draws about 0.5 mA when it is stalled.

Requirements	Verification
1. Able to move forward and backward at speed of 1m/s 2. Able to turn a 90 degree right turn or a 90 degree left turn 4. Able to stop at forward speed of 1m/s	1A. Find an empty hallway and make a 10m path 1B. Write a program that control the motors to drive forward or backward at speed of 1m/s 1C. Check if the drive subsystem able to d0 90 degrees right-turn and 90 degrees left-turn 2A. Find an empty hallway and make a 90 degree left-turn path and a 90 degree right-turn path 2B. Write a program that can control the motor to turn left or turn right 2C. Check if the drive subsystem meet the requirement 3A.Find an empty hallway and make a 10m path 3B. Write a program that control the motors to drive forward at a speed of 1m/s, and then after 5 second, control the motors to a complete stop 3C. Check if the motors come to a complete stop

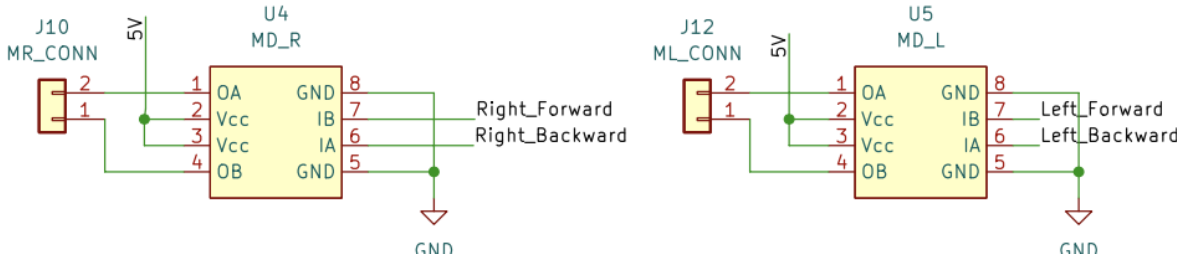


Figure 2.3.4.1 Motor Driver Schematic

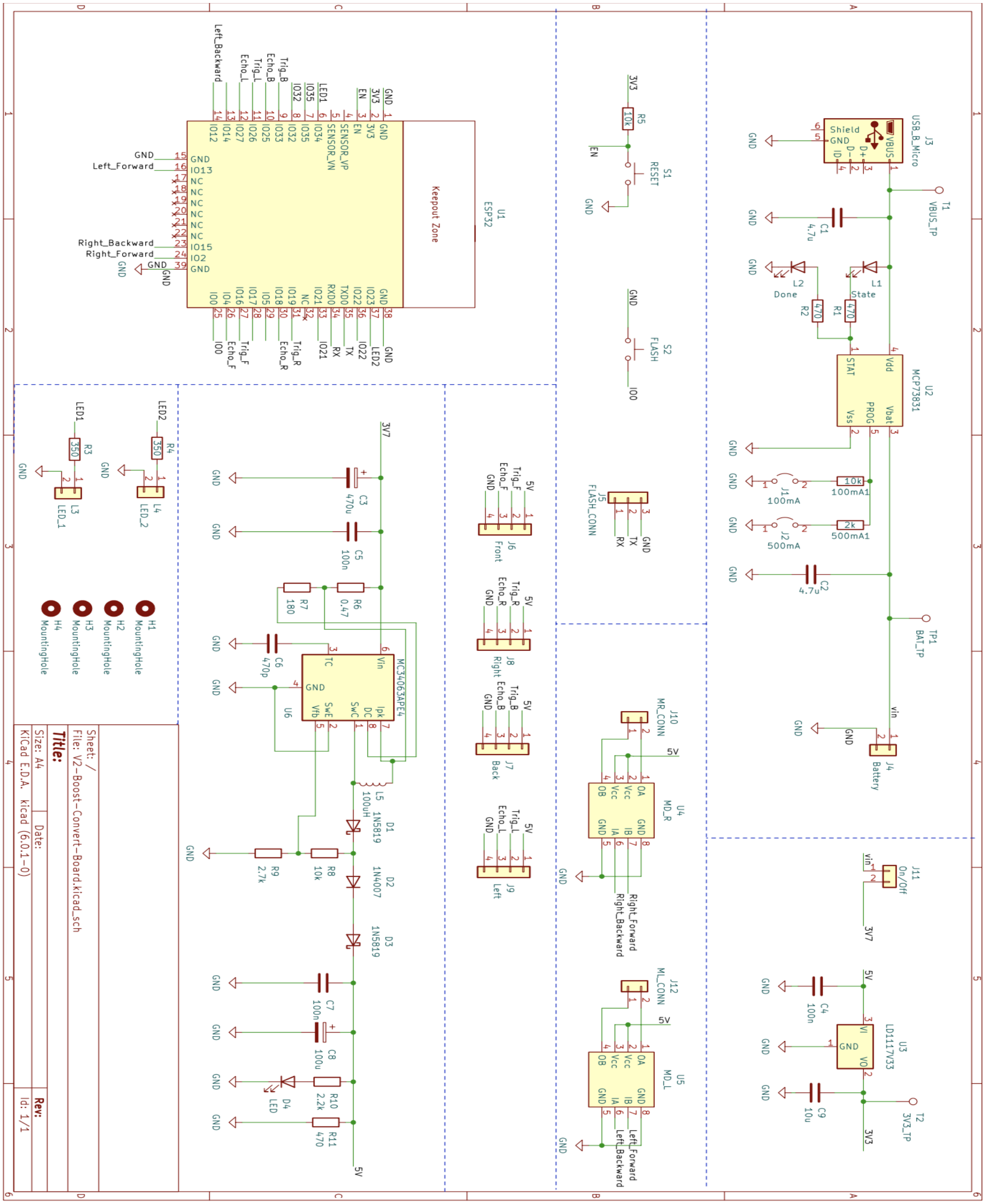
2.3.5 Network Subsystem

The network subsystem has a Wi-Fi module, RESTful APIs, and a user interface. The Wi-Fi module is integrated into the ESP32 microcontroller, and it can operate at 2.4GHz, and the rate of data transmission can be up to 150 Mbit/s. It is a very powerful Wi-FI module and allows us to control the Fun-E-Mouse remotely. We also will build some RESTful APIs in Python. RESTful APIs provide a great deal of flexibility. It can handle multiple types of calls, return different data. It also can be used in web services, applications, and software. This allows us to build a more stable and reliable user interface. RESTful APIs also provide a fast response time. The response time under 1 second is capable of controlling the Fun-E-Mouse remotely. The user interface enables us to control the Fun-E-Mouse to move right, left, backward, and forward. It also allows us to configure 2 different modes(automatic, remote) of operation.

Requirements	Verification
1. Able to detect and connect with any 2.4GHz Wi-Fi 2. Able to send/receive data to/from the server in most 1 second 3. Able to process the request in most 1 second 4. Able to control the Fun-E-Mouse to move left, right, forward, or backward in 1 second 5. Able to configure the 2 different	1A. write a program that will scan all the nearby Wi-Fi 1B. Inspect the output of the Wi-Fi scan program and determine if it detects all the 2.4GHz Wi-Fi 1C. Provide a known Wi-Fi's SSID and password, and ping google.com to check if it is connected with network 2A. Write a program that will send a POST request and a GET request, and have a time variable to keep track of how long the request took 2B. Check the time variable to determine if the Wi-Fi module has met the time requirement 3A. Write a program that will send a POST request or a GET request, and each API has had a time variable to keep

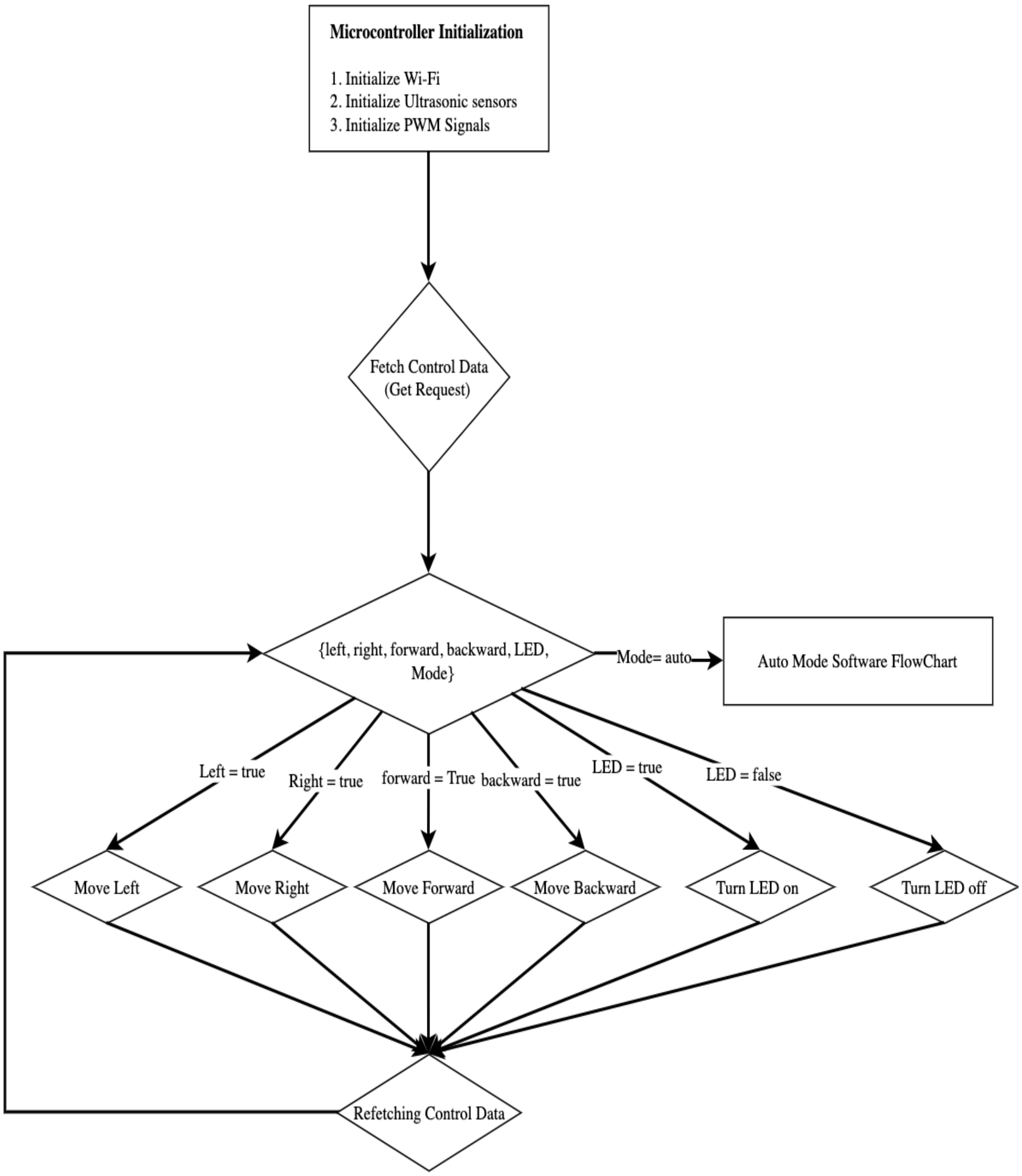
modes (automatic, remote)	track of how long the request took 3B. Check the time variable to determine if the API process that data in most 1-second 4A. Set a time variable to record when the control button is pushed, set another time variable to record when the microcontroller received the control command 4B. Check if the time when the controller received the command minus the time when the button was pushed is less than 1 second 5A. Set the mode to automatic from User 5B. Check if the Fun-E-Mouse controls itself 5C. Switch the mode to remote 5D. Check if the user interfaces are able to control the Fun-E-Mouse

2.4 Schematics of Overall System

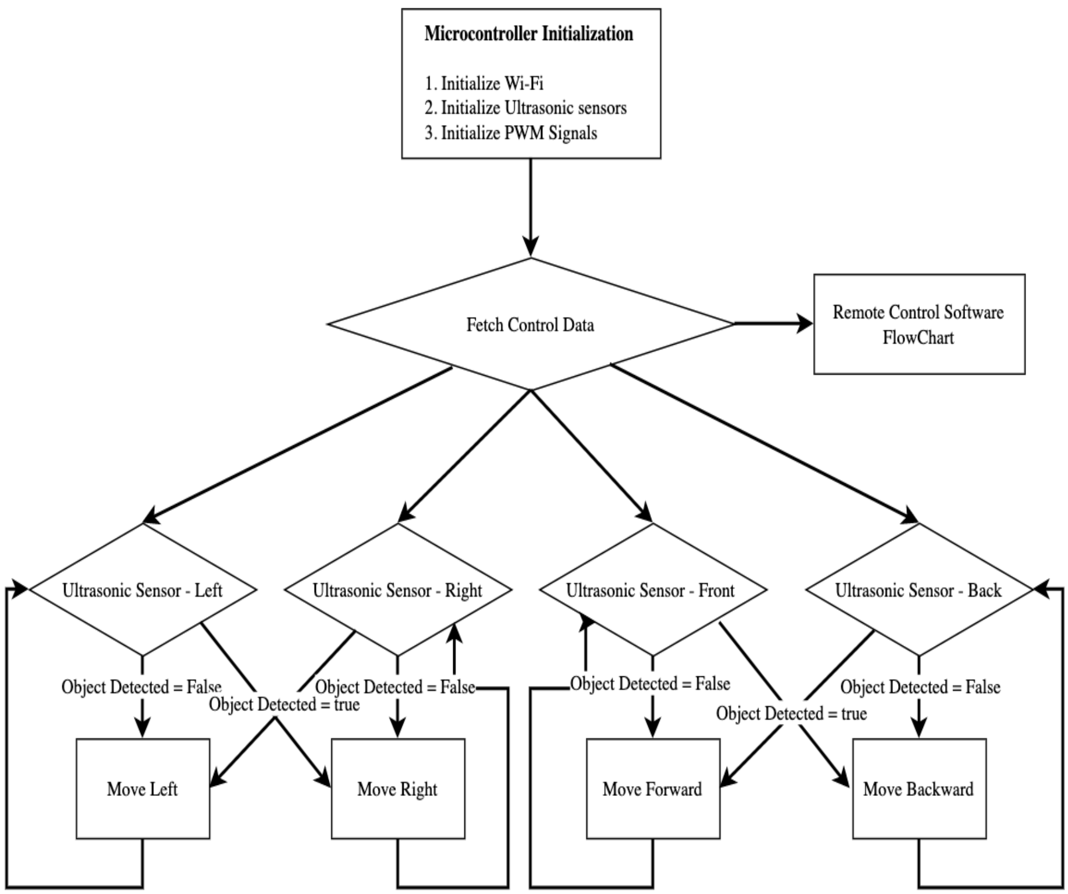


2.5 Software Flowchart

2.5.1 Remote Mode



2.5.2 Auto Mode



The auto mode of the Fun-E-Mouse is using the four ultrasonic sensors to detect any nearby object within 10cm distance. Let False denote no objects detected in front of the ultrasonic within 10cm, let True denote objects detected in front of the ultrasonic within 10cm. Let {Right, Left, Forward, Backward, Stop} denote the direction of the Fun-E-Mouse, and Let Choose{ } denote a random function that picks any given direction. The following table describes how the Fun-E-Mouse operates in the Auto Mode.

Ultrasonic Sensors				**Direction**
Left	**Right**	**Front**	**Back**	
false	false	false	false	Choose{forward, backward, right, left}
false	false	false	true	Choose{forward, right, left}
false	false	true	false	Choose{ backward, right, left}
false	false	true	true	Choose{ right, left}
false	true	false	false	Choose{forward, backward, left}
false	true	false	true	Choose{forward, left}
false	true	true	true	Left
true	false	false	false	Choose{forward, backward, right}
true	false	false	true	Choose{forward, right}
true	false	true	false	Choose{ backward, right}
true	false	true	true	Choose{forward, backward, right, left}
true	true	false	false	Choose{forward, backward}
true	true	true	false	Backward
true	true	true	true	STOP

2.6 Tolerance Analysis

2.6.1 Ultrasonic Sensors Analysis

The critical part of the Fun-E-Mouse project is the ultrasonic sensor. We have four ultrasonic sensors located at the front, back, right and left, then this allows us to detect any object nearby the Fun-E-Mouse. It also enables us to design the Mouse to navigate around objects, corners and under furniture.

The ultrasonic sensor works on the same principle as a radar system. It converts electrical energy into acoustic waves and vice versa. This wave signal is an ultrasonic wave that travels at a frequency of 40KHz . To measure the distance between the Fun-E-Mouse and the object, the ESP32 microcontroller sends a 10 us trigger signal to the ultrasonic sensor. When the ultrasonic sensor is triggered, it generates 8 ultrasonic wave bursts and initiates a time counter. As soon as the echo signal is received, the timer will stop.   The following diagram shows the timing of each signal of the ultrasonic sensors [5].

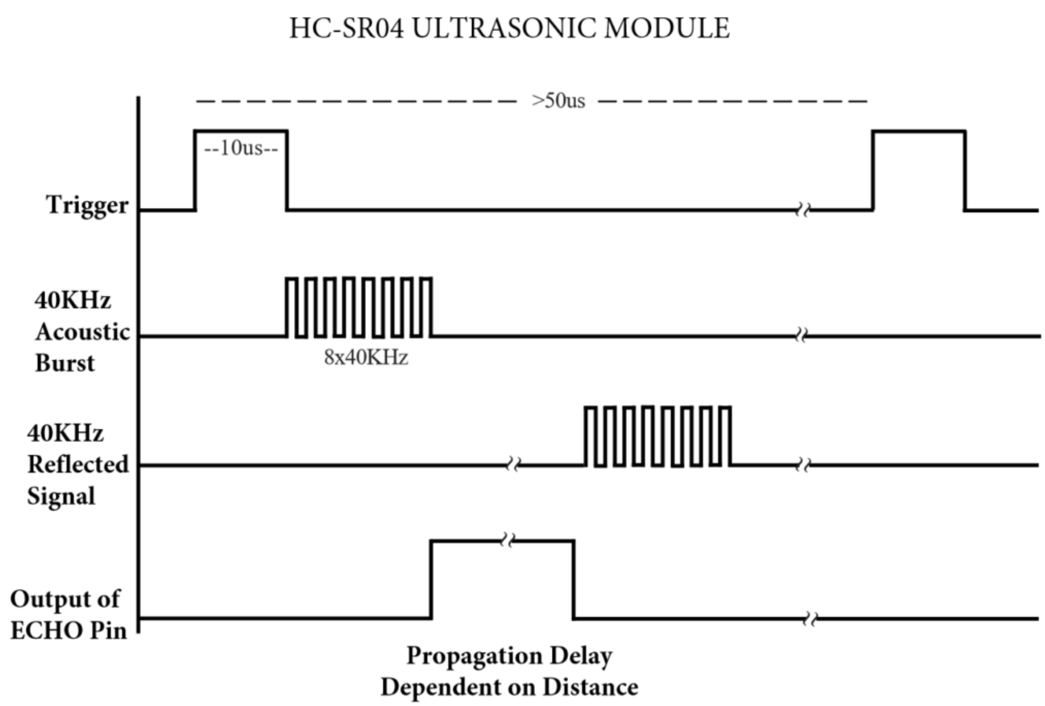


Figure1: Representation of trigger signal scountic bursts, reflected signal and output of ECHO pin (HC-SR04)

Theoretically, the distance between the Fun-E-Mouse with nearby objects can be calculated using how long the wave had traveled and the speed of the sound. Because the acoustic sound traveled from the ultrasonic transducer and traveled back to the transducer, therefore, it is a two-way trip, therefore we need to divide the distance by 2 to calculate the actual distance between Fun-E-Mouse and the object. The following equation calculates the distance to an object placed in front of the Mouse [5].

$$distance = \frac{time\ taken\ x\ speed\ of\ sound}{2}$$

Under these following circumstances that ultrasonic sensor may not work properly [5].

1. Object like fabric or Carpet can absorb acoustic signals, then the signal cannot reflect back to the sensor, therefore, the distance cannot measured correctly
2. Small object only reflects a small amount of acoustic signal and the sensor cannot detect it, therefore, the distance cannot measured correctly
3. High/Low intense sensitivity of ultrasonic sensor can detect false signal, such like airwave or fan can cause false signal, therefore, the distance cannot measured correctly

2.6.2 Power Analysis

The energy pulled out from the battery can be measured by calculating the amount of current that we draw from it. We can use the following equation to compute the required number of energy(in amp-hours) for the mouse to operate for 60 minutes. We can then use this result to select the kind of battery that we want to use.

$$E = P * t$$

Which is the load factor multiples the rated power P (in W), and the total operating hours t (in hours) equal to the total power consumption E (in Wh).

But first, we need to calculate the power for all the units in our design.

	Power = VI^2
Two DC motors	3.3V * 150^2mA * (0.7) * 2= 139.5 mW
Four Ultrasonic Sensors	3.3V * 2^2mA * 4 = 0.0528 mW
Microcontroller (Active mode)	3.3V * 128^2mA = 54.067 mW
Total Power	193.62 mW

Assuming we have the mouse operate for 60 mins, and neglects the power for LEDs and friction, the total energy used is about 193.62 * 10^{-3} * 1 = 193.62mWh. Using the formula of

$$(Wh)/(V) = (Ah)$$

We can solve for the current needed, having a 193.62mWh battery rated at 3.7V, the current is 193.62mWh / 3.7V = 52.4 mAh. With this in mind, we believe that a 3.7v Polymer Li-ion Battery with 6600mAh by Adafruit is overqualified for its job. It can provide sufficient power for the two DC motors and the circuit to run for 30 minutes as we expected.

3. Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

According to Salary.com, the average hourly wage for Engineer I in the United States is $34, and assumed with 250% overhead multiplier. Also, Senior design class is a 4 credit hour class, that means each student must spend at least 12 hours per week. We will spend about 7 weeks on this Fun-E-Mouse project, therefore the total labor cost is calculated below:

$$\$34\ /\ hr\ *\ 2.5\ (overhead)\ *\ 12\ hr\ /\ week\ *\ 3\ people\ *\ 7\ weeks\ =\ \$21,420$$

3.1.2 Parts

Part	Part No.	Quality	Price	Total
ESP 32	ESP32-WROOM-32E-N8	1	$3.30	$3.30
Ultrasonic Sensor	HC-SR04	4	$3.95	$15.80
Voltage Regulator	LV1117-33	1	$0.95	$0.95
Lithium Ion Battery	IRC18650	1	$24.50	$24.50
Lithium Ion Charger	MCP73833	1	$12.50	$12.50
Motor Driver	L9110H	2	$1.50	$3.00
TT Motor	TTMotor	2	$5.95	$11.90
Wheel	3757	2	$2.50	$5.00

Boost Switching Regulator	296-36112-5-ND	1	$0.92	$0.92
On-Off switch	EG5619-ND	1	$0.67	$0.67
Headers & Connectors	—	—	$3.00	$3.00
Resistors & Capacitors	—	—	$3.00	$3.00
Total				$84.54

3.1.3 Cost of Custom Manufacturing

Manufacturing Method	Hours	Rate per Hour	Material Cost	Overall Cost
3D Printing	5	$34.00	$29.99	$709.99
PCB	N/A	N/A	$4.99	$4.99
Total				$714.98

3.1.4 Sum of costs into a grand total

The total cost for the project = cost for labor + cost for parts + cost for encasing = $22,219.52 .

3.2. Schedule

Date	Chuangy	Yingyu	John
Feb 28	Write Arduino code to unit test the Wi-Fi module, motor drivers, motors, and ultrasonic sensors	Work on the Drive Subsystem of the PCB, ordering parts as necessary, go to PCB Board review	Test sensors and determine appropriate placement for wheels and sensors on the chassis
Mar 7	Write RESTful APIs in Python to unit test APIs request, and able to control	Finish PCB and place orders for PCB and all the other components as	Design/3D printing chassis and wheels

	LEDs and movement	necessary	
Mar 14	Ensure the ultrasonic sensors are working properly and enable to detect nearby objects	Begin soldering for PCBs, and help test ultrasonic sensors accuracy	Set-up 3D printer and revise design for chassis and wheels as needed
Mar 21	Write User Interface (Web App) in Java & html to control the Fun-E-Mouse	Finish soldering and assembling the PCB with other mechanical parts and test.	Help with building frontend for user Interface as needed
Mar 28	Ensure the ultrasonic sensors are working properly and enable to detect nearby objects	Revise PCB as needed, prepare for the Second Round PCBway Orders	Revise 3D printing for chassis and wheels as needed
Apr 4	Optimize the code and debugging	Finalize PCB, integrate prototype	Integrate prototype
Apr 11	Reserved for debugs and testing	Reserved for debugs and testing	Reserved for debugs and testing
Apr 18	Prepare for demo, fix any last-minute bugs	Prepare for demo, fix any last-minute bugs	Prepare for demo, fix any last-minute bugs
Apr 25	Prepare for Final Presentation	Prepare for Final Presentation	Prepare for Final Presentation
May 2	Write Final paper	Write Final paper	Write Final paper

4. Discussion of Ethics and Safety

We will conduct ourselves according to the Code of Ethics published by IEEE[7] and ACM[8] as following:

1. Be sure "to not engage in harassment or discrimination, and to avoid injuring others"[7].
2. Not make use of user data without consent while users are using our web app that controls the electrical mouse.
3. Be sure to adapt the principle of justice, ensuring all users are equal.

4. Be sure to avoid harm and ensure “to minimize the possibility of indirectly or unintentionally harming others”[8], including pets.

We will also adhere to the Safety Guidelines set forth on the ECE 445 course website[4], as well as the document of General Battery Safety[3] provided on the ECE445 course website. Further, we need to protect ourselves and others in the lab when working with batteries. Batteries have many hazards, such as electrical shock, flammable gasses that could fire or explode, and battery acid that can burn skin or eyes. We need to consult the battery’s manuals for safety precautions and proper handling, as well as hazard identification. We need to make sure that the battery is properly secured in the equipment for the finished product to prevent injuries in end users (ex: cats) while improving our battery's performance. The outer shell of the electronic mouse might contain some electrical (sensors) and mechanical (wheels) parts. We need to make sure that they are safe and not harmful to pets or humans since this is designed to be a toy for cats. We will make sure the outer shell of the mouse is firmly mounted, with no external parts that can easily be ripped off, or swallowed, causing life-threatening damage to cats.

5. Citations

Work Cited

[1] Beth L. Strickler, Elizabeth A. Shull, "An owner survey of toys, activities, and behavior problems in indoor cats", *Journal of Veterinary Behavior,* Volume 9, Issue 5, 2014, Pages 207-214, ISSN 1558-7878, https://doi.org/10.1016/j.jveb.2014.06.005. (https://www.sciencedirect.com/science/article/pii/S1558787814000884)

[2] Miller, K., and K. Watts. (2015) "Environmental and behavioral enrichment for cats." *Animal behavior for shelter veterinarians and staff*, p. 234.

[3] "General Battery Safety." *Safe Practice for Lead Acid and Lithium Batteries*, [Online]13 April 2016, <https://courses.physics.illinois.edu/ece445/documents/GeneralBatterySafety.pdf> [Accessed 10 February 2022].

[4] "Safety ECE 445 - Senior Design Laboratory." *ECE 445*, <https://courses.engr.illinois.edu/ece445/guidelines/safety.asp.> [Accessed 24 February 2022].

[5] Reese, Lynnette. "The working principle, applications and limitations of ultrasonic sensors." *Microcontroller Tips*, 6 August 2019, <https://www.microcontrollertips.com/principle-applications-limitations-ultrasonic-sensors-faq>. [Accessed 24 February 2022].

[6] Paredes, Israel. “Light Emitting Diode with dimensions 5mm.” CAD Files. <https://grabcad.com/library/led-5mm-3 >. [Accessed 24 February 2022].

[7] “IEEE Code of Ethics.” *IEEE*, IEEE Policies, Section 7 - Professional Activities (Part A - IEEE Policies), <https://www.ieee.org/about/corporate/governance/p7-8.html>. [Accessed 10 February 2022]

[8] “ACM Code of Ethics and Professional Conduct.” *Association for Computing Machinery*, ACM, <https://www.acm.org/code-of-ethics>. [Accessed 10 February 2022].