ECE 445 Senior Design Laboratory

Design Document

Laser/Voice Assisted Cat Toy

Team # 9

Paul Jablonski (pjj3)
Rahul Grover (rgrover4)
Yutong Gan (yutongg9)

TA

Rui Gong

October 3rd, 2024

Table of Contents

1. Introduction **1**
1.1 Problem 1
1.2 Solution 1
1.3 Visual Aid 2
1.4 High Level Requirements 2

2. Design **3**
2.1 Block Diagram 3
2.2 Physical Diagram 4
2.3 Subsystem Overview 5
2.3.1 Sensor Subsystem 5
2.3.2 Microcontroller Subsystem 6
2.3.3 Motors & Output Subsystem 7
2.3.4 Power Subsystem 8
2.4 Subsystem Requirements 9
2.4.1 Sensor Subsystem 9
2.4.2 Microcontroller Subsystem 9
2.4.3 Motors & Output Subsystem 9
2.4.4 Power Subsystem 9
2.5 Subsystem Verifications 10
2.5.1 Sensor Subsystem 10
2.5.2 Microcontroller Subsystem 11
2.5.3 Motors & Output Subsystem 12
2.5.4 Power Subsystem 13
2.6 Supporting Material 14
2.6.1 Circuit Diagram 14
2.6.2 State Machine Diagram 14
2.7 Tolerance Analysis 15

3. Cost and Schedule **16**
3.1 Bill of Materials 16
3.2 Schedule 17

4. Ethics and Safety **18**
4.1 Ethical Analysis 18
4.2 Safety Concerns 19
4.2.1 CPSC Regulations 19
4.2.2 UIUC Lab Safety Guide 19
4.2.3 Safety Practices 19

References **20**
Datasheets **21**

1. Introduction

1.1 Problem

Modern cat toys have some systems for automatically moving around, but rarely use any sophisticated sensors. This is commonly seen in commercial toys like balls that roll around in random patterns[1]. However, these widespread and commercial systems could use some serious improvements as problems exist in longevity, noise generation, user friendliness, and an overall lack of interaction for the cats[2].

These toys typically bang into walls without any preventative systems in place, leading to potential damage not only to the toy itself but also to the pet owner's home - on top of being terribly loud[2]. This is significant as owners may need to replace their cat's toys far more than desired[2]. Additionally, the typical toy's constant speed and random directional movements diminish the quality of a cat's play experience, often resulting in toys being left under furniture or cabinets unnoticed. With their fixed and unexciting movement, cats may often stare or fear these toys rather than chase them down as they would a live animal[2]. Thus, given the importance of engaging play for a cat's health, owners are often burdened by the current market's lackluster and rudimentary options.

1.2 Solution

We propose that these problems be resolved through a mouse-like toy, which has been seen before, but is now refined with multiple more advanced systems. The primary sensors will consist of a distance measuring laser, a vibrational sensor, and a microphone in tandem with a miniature speaker. The first two will serve in engaging the cat and improving both longevity and noise generation, whereas the final sensor system will vastly improve user interaction through various voice commands. More specifically, the laser can be utilized to avoid collisions and detect ahead of the mouse, sparking movement changes that are more realistic for an animal. The vibrational sensor will furthermore detect when the toy has been caught, which will either dispense a treat or play dead depending on the toy's state. Finally, the aforementioned voice commands will allow for the user to locate the toy at any time, or manually activate the toy itself without a need for physical contact.

Physically speaking, the non-rolling shape of a mouse will also allow for more rigid and controllable movement. This is important for stabilizing the toy’s primary sensors and will allow for greater reactivity to its environment and consequently less noisy behavior. The sensors as mentioned will also be accompanied by several motorized and more physical systems. A moving tail will be used to mimic the more excitatory behaviors of prey, making it more engaging than a typical toy's static tail. This would be accompanied by faster motorized movements and more realistic movement states as will be regulated by our microcontroller and brushless DC wheel system. Further speaking, a latch will be controlled through a servo and used for dispensing treats from the back of the toy upon being caught. And finally, a rechargeable lithium ion battery will be incorporated and regulated by a battery management system for easy access USB-C charging.

1.3 Visual Aid

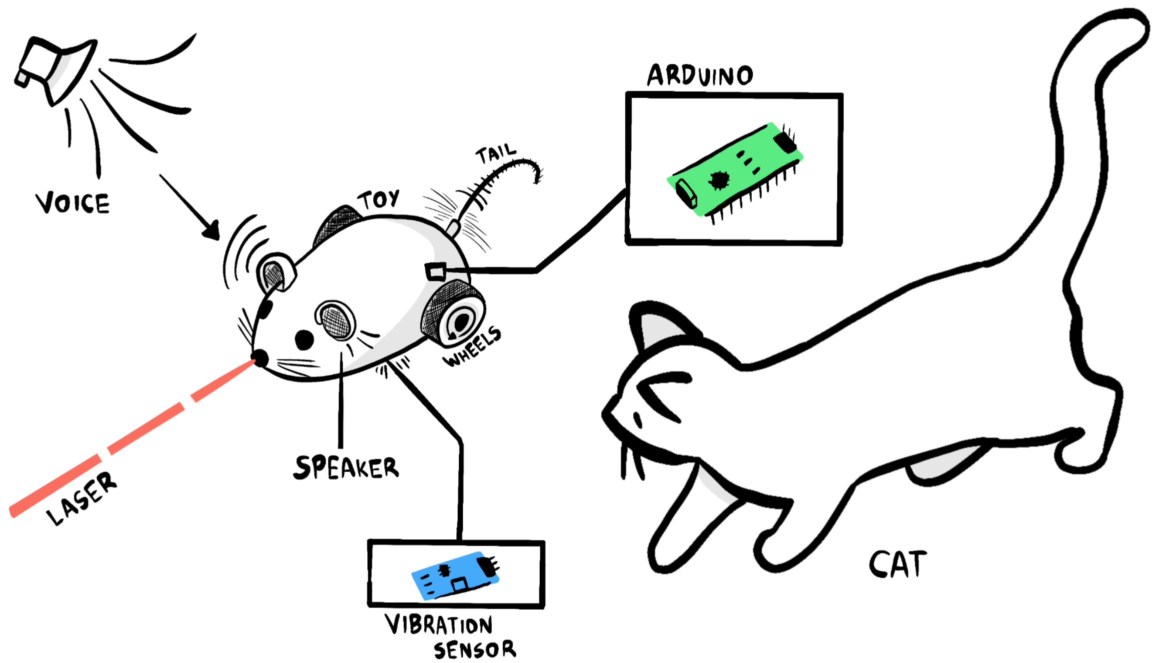

Figure 1. Example usage of cat chasing toy, laser distance measuring, and voice commands.

1.4 High Level Requirements List

In order to accurately claim that this project has been successful, certain parameters and quantitative goals must be met. These goals are as follows:

- *Obstacle Detection and Avoidance* - The toy must be capable of stopping and turning if a wall or obstacle is detected ahead of it. It must then move in another direction continuing its play state while still avoiding other barriers. The detection range of obstacles should be observed up to one meter ahead, such that the toy may stop and perform this behavior properly.

- *Noise Reduction* - The toy must be quieter than the common and commercially automated cat toy. This is directly measurable in decibels and can be evaluated through recording a set distance away from the toys while they are active. Motors may also be individually tested in an idle position such that the toy does not move away from the recording device. This requirement is also testable in regards to quantifying and comparing audible wall collisions between the toys - where this project's toy should have far less collisions and ideally none.

- *Interactivity and State Changing* - The toy must include at least four movement styles, which must vary significantly from the constant speeds seen in commercial toys. Furthermore, the tail must be capable of moving during these movement states. This can be tested through going through each of the microcontroller states and observing the wheel speeds changing variably, or through simply observing the toy's behavior in comparison to a standard cat toy.

- *Voice Commands* - The toy should be capable of detecting human speech and recognizing the various voice commands assigned to it. The commands should be functional from at least five meters away, and at least three different commands should be included. Ideally these commands function to at least locate, start, and stop the toy respectively.

2. Design

2.1 Block Diagram

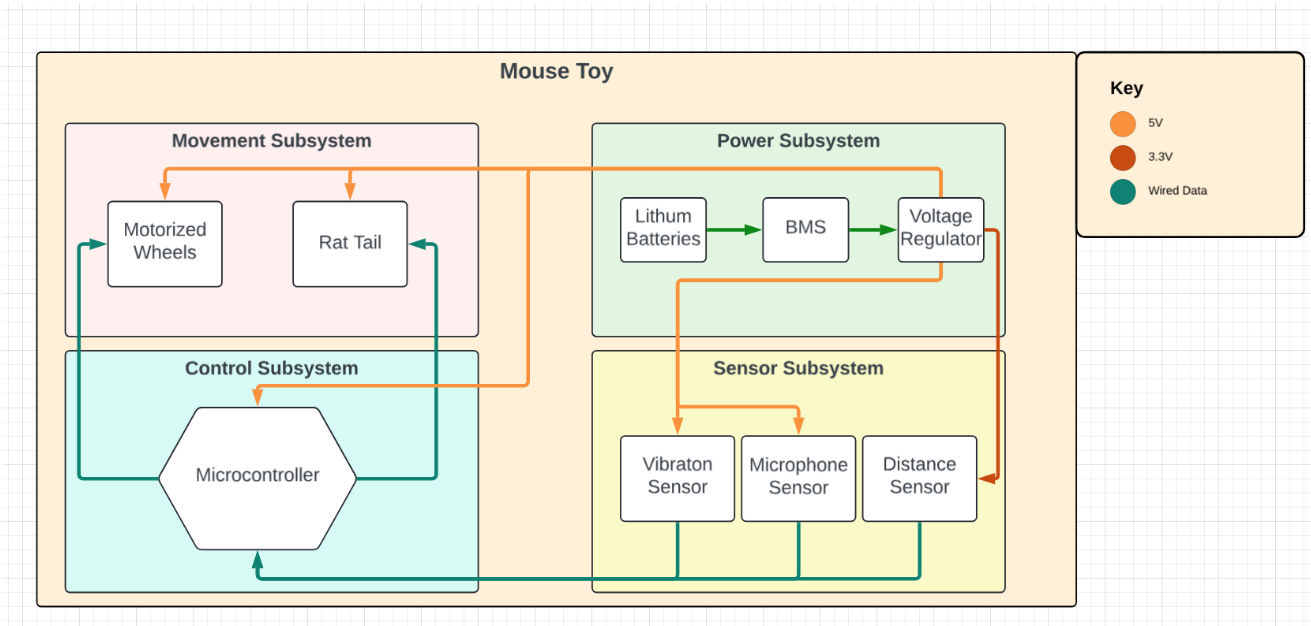

Figure 2. Block diagram representing each subsystem and their connections.

2.2 Physical Diagram

Figure 3. Physical diagram of disassembled front view of the toy.

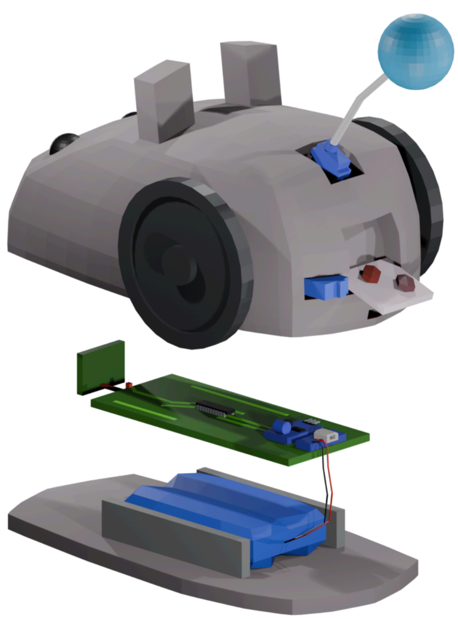

Figure 4. Physical diagram of disassembled back view of the toy.

2.3 Subsystem Overview

2.3.1 Sensor Subsystem (Laser, Vibration, Microphone)

The sensors should be capable of receiving data from the toy's environment in order to supply it directly to the microcontroller subsystem, which will decide upon motor and sound outputs. These sensors are vital for the microcontroller such that it can process the input data, but they are even more important for the motors and output subsystem. If the sensors fail to supply the microcontroller at the specified frequencies, then the toy will collide into walls, not change states upon being caught, and it won't be responsive to voice commands.

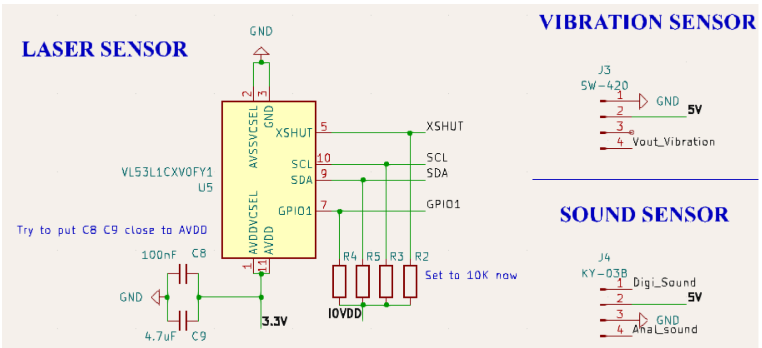

Figure 5. Sensor subsystem circuit schematic.

The sensor subsystem consists of a VL53L1X laser sensor, one SW-420 vibration sensor, and one KY-038 sound sensor.

The VL53L1X laser sensor has an adjustable maximum 4m sensing range and fast ranging frequency up to 50Hz. The laser sensor will be applied on the front side of the toy. This sensor can help detect the obstacles around the toy and generate control signals to avoid collisions. The outputs of this will be sent to the microcontroller for further processing to control motor behaviors.

The SW-420 is a highly sensitive non-directional vibration sensor. The sensitivity can be adjusted simply by changing the resistance of the potentiometer. The output signal will be passed to the microcontroller to determine whether the toy has been caught or not.

The KY-038 is a highly sensitive sound sensor. The sensitivity can be adjusted by changing the resistance of the potentiometer. The sensor should be able to detect and decipher human speech, then send the signal generated to the microcontroller. The sensor should be able to receive command to turn on/off the toy or play some specific sound to help locate the toy.

2.3.2 Microcontroller Subsystem

The microcontroller subsystem is generally responsible for receiving inputs from the sensor subsystem, while providing outputs towards the motors and outputs subsystem. For instance, this system should intake laser distance measurements and output the correct behavior to each motor to avoid collisions. This may include telling the motors to turn the toy until nothing is detected ahead of it, through which the microcontroller will then tell the toy to continue moving forward as normal. Beyond the example provided, this system is also responsible for properly processing voice commands and determining when the toy has been caught based on the vibration sensor's inputs.

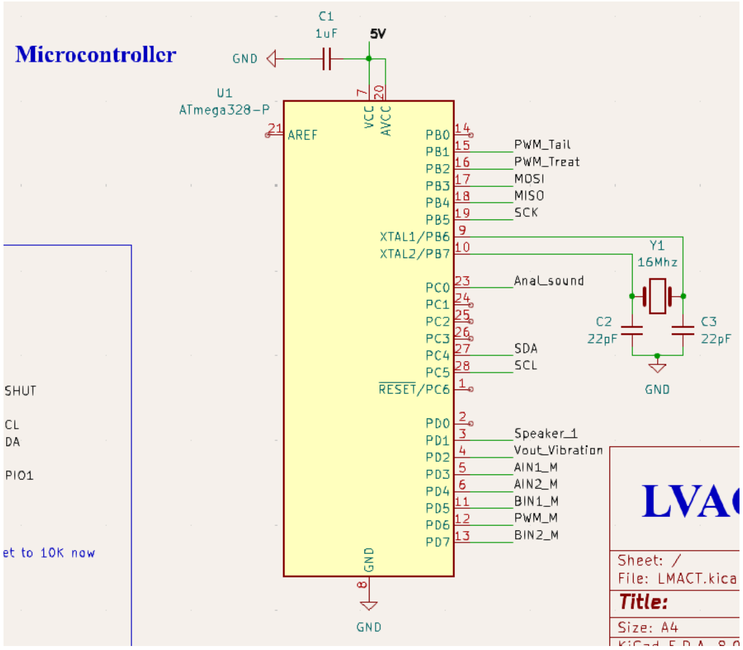

Figure 6. Microcontroller subsystem circuit schematic.

The microcontroller subsystem is driven by a small Arduino chip known as the ATmega328P. This chip is capable of handling up to 14 digital input/output and 6 analog inputs. PWM signals are also capable of being processed by 6 of the digital pins. The sensor subsystem will be correspondingly connected to the microcontroller subsystem as inputs.

The sound from the KY-038 sound sensor will be handled through analog inputs in order to preserve sound data, otherwise digital would treat it as binary.

The VL53L1X laser sensor will communicate through I2C protocol through a digital pin in order to provide information to our program about object proximity.

The SW-420 vibration sensor will also be handled through digital input as it will contain a threshold trigger for activation. This will be manually adjusted to determine the severity of impacts and if a cat has caught the toy.

PWM signals will be used for outputs of the microcontroller program, where the H-bridge motor inputs and the speaker outputs will both be functionally controlled through the PWM capable pins.

2.3.3 Motors & Output Subsystem (Wheels, Tail, Latch, Speaker)

The motors and output subsystem will allow the toy to move forwards, backwards, and directionally across any surface. Based on what the sensors send to the microcontroller, different predefined movements will be performed by the motors. This includes avoiding obstacles, different play styles, and differently set speeds. The tail will also move variably based on the microcontroller's sensor readings to randomize the toy's behavior further. The speaker system will also be attached to this subsystem, as it will play sounds when prompted by the microcontroller's processing of voice commands.

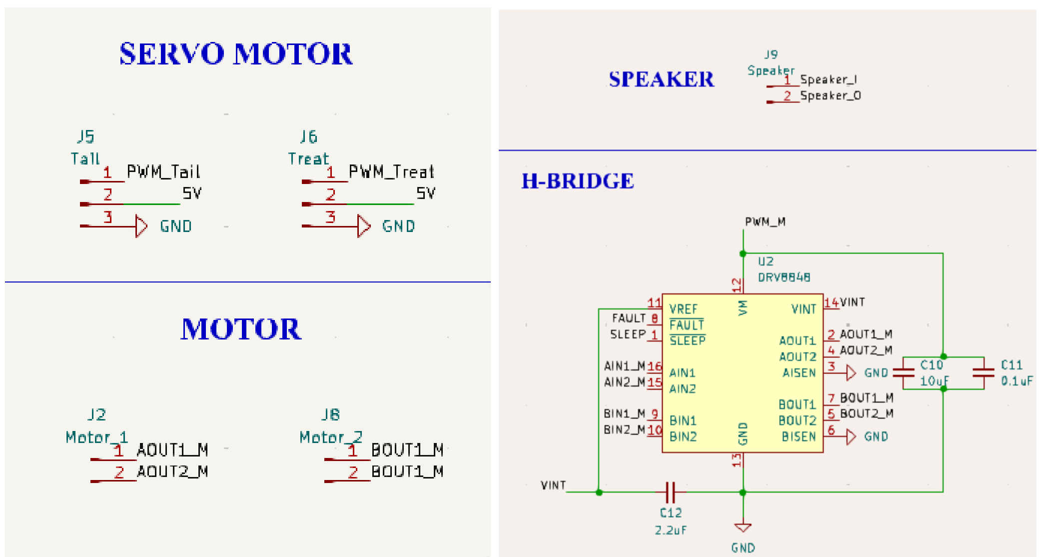

Figure 7. Motors & outputs subsystem circuit schematic.

The Motors/Output subsystem consists of two GA-12 N20 DC motors, two SG90 servo motors, one DRV8848 H-bridge, and one EK1794 speaker.

The GA-12 N20 DC motors have a maximum of 1000 RPM and can be controlled by PWM from 3V to 12V. When using a wheel with a diameter of 32mm, the toy can achieve a maximum speed around 1.675m/s.

The SG90 is a servo motor that can be driven by 4.8V to 6V. Based on the PWM input, the motor can be used to mimic the behavior of tails, or it can be used to dispense treats after the toy is caught.

DRV8848 H-bridge is a motor control unit. It has 2 logical signals for each motor input. Depending on the logic signal, it has 4 states: Forward, Reverse, Brake and Coast. It also has a PWM input that can be used to control the speed of the motor.

EK1794 is a 40mm microphone with 3W and 4 Ohms. It is used to play the recorded sound to help locate the toy.

2.3.4 Power Subsystem (Battery, BMS, Power Button)

The power subsystem provides power to the entire device, and it will handle charging, discharging, and voltage regulation to every other subsystem. It will utilize multiple fixed output regulators to supply different voltages to different sensors as required. This is seen as the laser sensor requires 3.3V, but the vibrational and microphone sensors need 5V. Furthermore, this subsystem includes a battery management system that prevents overcharging and protects the circuitry of all other subsystems. The overall output of the power subsystem will be enabled and disabled through a power button.

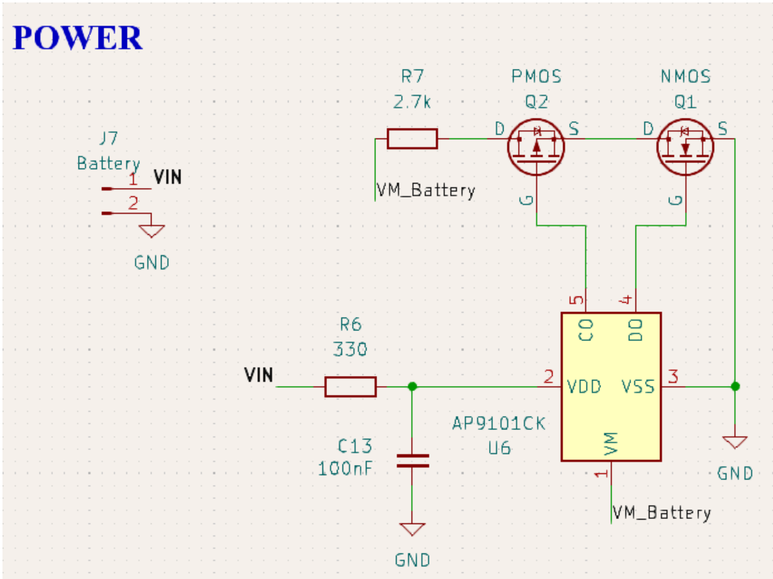

Figure 8. Power subsystem circuit schematic for battery input and management.

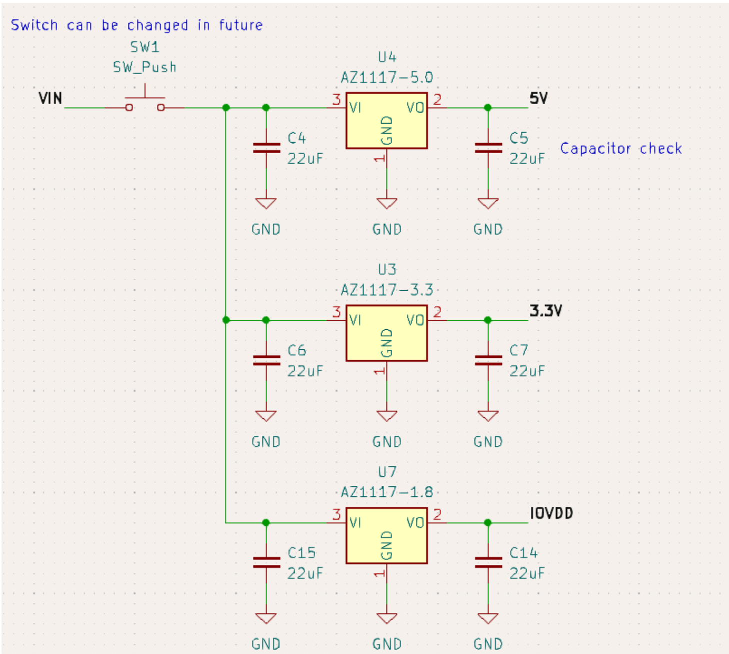

Figure 9. Power subsystem circuit schematic for voltage regulators and button.

The power subsystem consists of a 7.2V Li-ion battery, one AP9101CK battery regulator, and three AZ1117 voltage regulators. We will use a 7.2V 3600mAh rechargeable Li-ion battery to power our toy.

The AP9101CK is a Li-ion battery protection unit which detects and prevents overcharge, over discharge and other abnormalities.

The AZ1117 voltage regulator should provide stable voltage based on our needs. We need one AZ1117-5.0 to provide a stable 5V to power most of our sensors, the microcontroller, and the speaker. We need one AZ1117-3.3 and one AZ1117-1.8 to provide 3.3V and 1.8V for the laser sensor. We use a button to turn on/off the toy.

2.4 Subsystem Requirements

2.4.1 Sensor Subsystem (Laser, Vibration, Microphone)

- The VL53L1X laser sensor should be able to detect obstacles and walls from at least one meter away, and send information fast enough such that the toy can stop in time to avoid them.
- The SW-420 vibration sensor will operate constantly to detect if the toy has been caught by a cat based on a LOW-HIGH output threshold. It will then tell the microcontroller to change its movement and/or dispense a treat if a HIGH threshold is reached.
- The KY-038 microphone should decipher human speech in the 40-50 dB range and tell the microcontroller what it hears, which will resultantly locate the toy through a speaker cue, or power the toy on or off.

2.4.2 Microcontroller Subsystem

- The ATmega328P microcontroller should only signal to one of the wheel motors upon an obstacle appearing in front of it, causing the toy to turn.
- The microcontroller should also change the movement to a ‘caught’ state upon sudden impacts and vibration, either resting the toy momentarily or dispensing a treat based on how many times it has been caught - three catches will suffice.
- When voice commands such as ‘start’, ‘stop’, and ‘locate’ are received by the microphone alongside a keyword, the microcontroller will move the toy, stop it, or play a sound.

2.4.3 Motors & Output Subsystem (Wheels, Tail, Latch, Speaker)

- The N20 DC motors will individually receive inputs from the microcontroller and properly turn the toy or accelerate and decelerate when necessary.
- The tail SG90 servo motor will oscillate the tail twice per second when play states are active and during most general movements.
- The SG90 servo latch will open at least 45 degrees to allow for a treat to be dispensed upon the microcontroller identifying a ‘caught’ state from the vibrational sensors.
- The EK1794 speaker module will play short and basic sounds when voice commands are properly received, and will repeat a sound when the ‘locate’ command is given until retrieved.

2.4.4 Power Subsystem (Battery, BMS, Voltage Regulator, Power Button)

- The TP4056 battery management system should provide overload, underload and short-circuit protection.
- The voltage regulator will ensure the lithium ion battery provides the correct voltages to each subsystem part. Illustratively, 3.3V ± 0.1V for the laser sensor and 5V for the other sensors and parts such as the microcontroller and motors.
- The power button should power on and off the system.

2.5 Subsystem Verifications

2.5.1 Sensor Subsystem (Laser, Vibration, Microphone)

Requirement	Verification
The toy should be able to detect any obstacles within the range of 1m and automatically move away from it to avoid collision.	1. Place an obstacle (like bottle, book. Etc.) on the right, left, or front of the toy within 1m. 2. Directly measure the sensor outputs around when obstacles present. 3. Outputs should measure below the 1,000mm threshold for turn state change.
The toy should be able to identify if it is caught.	1. Shake the toy first. 2. Directly measure the sensor output after shaking. 3. Outputs should measure HIGH for the part's threshold for caught state change.
The toy should be able to turn on/off through human speech. A recorded sound should be played after certain commands.	1. Directly measure the sensor output for a 40-50 dB threshold after speaking out some command. 2. Check if the power is on/off after speaking the corresponding command. 3. Check if the sound is played after speaking the corresponding command.

2.5.2 Microcontroller Subsystem

Requirement	Verification
Toy should begin in an audio state ready for user voice activation. Should react to commands “start” and “locate”	1. Say the word “locate” and ensure that the toy begins to make a noise. 2. Find the toy and ensure that the noise stops once found. 3. Say the word “start” and ensure that the toy begins to move.
The toy should stop either when the vibration sensor detects the cat caught it, or the command “stop” is given	1. While the toy is moving, say the word “stop”. 2. Ensure later behavior reflects what the toy should do after stopping. 3. While the toy is moving again, pick it up or start touching it to see if it stops. Repeat step 2.
The toy should only dispense a treat to the cat when the toy is caught successfully 3 times. Otherwise, it should stop momentarily but then continue to move. Once a treat is dispensed, it should continue moving along.	1. Stop the toy once by grabbing it, ensure that it momentarily stops and then continues again. 2. Repeat step 1 two more times, and see if on the last stop the toy dispenses a treat. 3. After dispensing a treat, make sure the toy continues to move.

2.5.3 Motors & Output Subsystem

Requirement	Verification
The toy should be able to move forward and backward at the average speed of 1m/s.	1. Find an empty hallway and make a 5m path. 2. Then measure the time needed for the toy to go through it. 3. Verify the speed is as specified.
The toy should be able to completely stop at its maximum forward speed of in 1s.	1. Write a program to accelerate the toy to its maximum speed. 2. Write a program to change its state from forward to Coast. 3. Record the time needed to stop the motor.
The toy should be able to turn 90 degrees right or left.	1. Find an empty place, then make a 90 degrees path. 2. Check if the toy can pass it.
The tail driven by the servo motor should be able to move up and down in different frequencies depending on states.	1. Write a program to generate PWM. 2. Check if the servo motor rotates properly.
The treat should be able to be dispensed after the toy is caught.	1. Write a program to generate PWM. 2. Check if the servo motor rotates properly.
The speaker should play a recorded sound properly upon command receival.	1. Write a program to trigger the speaker. 2. Check if the recorded sound is played. 3. Measure the dB of the sound played to be at least 40dB.

2.5.4 Power Subsystem (Battery, BMS, Voltage Regulator, Power Button)

Requirement	Verification
Must provide stable 5V, 3.3V and 1.8V voltage with 5% regulation.	1. Directly measure the voltage of input connections using a multimeter. 2. Use an oscilloscope to check if the voltage is clean enough and within the 5% regulation.
The battery should be able to power the toy for at least 20 minutes.	1. After the toy is assembled, we will put the mouse in a confined area to run freely. 2. Measure the maximum time before the toy stops moving.
The button should be able to turn the toy on/off	1. Press the button on to validate that movement has begun. 2. Press the button off to validate proper turning off. 3. Repeat 5 times to confirm continued functionality.

2.6 Supporting Material

2.6.1 Circuit Diagram

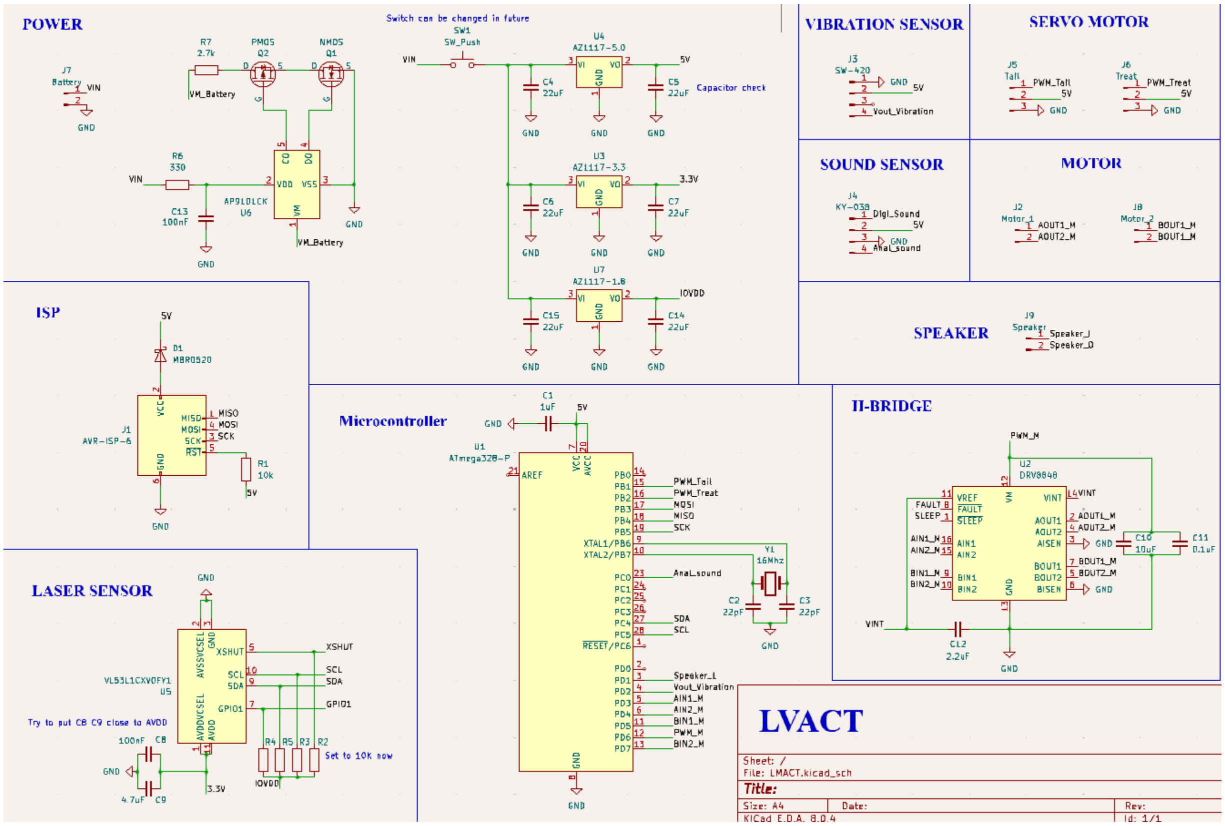

Figure 10. Overall schematic view for circuit connections.

2.6.2 State Diagram

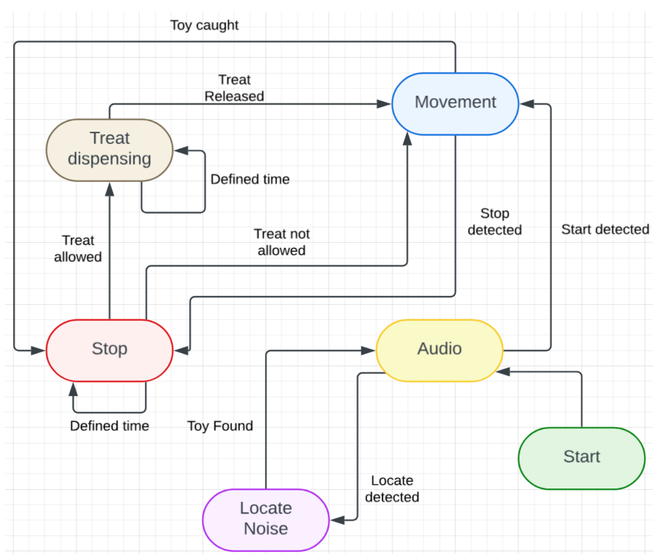

Figure 11. State diagram for microcontroller

2.7 Tolerance Analysis

While identifying potential issues for our cat toy, we were able to isolate the most important to the following two:

1. Identifying an optimal speed for the cat toy at which the range sensor can update and the toy can adjust its speed in time.
   - This will allow us to create more realistic movements as well as successfully avoid multiple collisions, both of which are goals outlined in Section 1.4.

In order to address the issue of optimal speed, we can take a look at the motor for our wheels itself, the N20 DC motor. Before we look into the specifications of the motor itself, we can mathematically model the speed of any given wheel of radius **r** millimeters with a motor rotating at a rate of ‘n’ revolutions/min.

$$velocity = \frac{n}{60} \times 2\pi r = \frac{n\pi r}{30}\ mm/s$$

For the N20 DC motor, the maximum RPM is 1000 and we have a chosen wheel diameter of 32 mm. Under these specifications, we get a value for the velocity of roughly 1.675 m/s. The range of our VL53L1X laser sensor is up to four meters in the best case, leaving our microcontroller roughly 2.388 seconds. In our case, we have chosen to utilize it up to one meter as we don’t require the full range. This would leave our microcontroller with about 0.597 seconds, which should still be more than enough to make movement alterations such as stopping or changing directions. Furthermore, we may decide to utilize half of this maximum velocity to allow for double the time in decision making if necessary.

2. Being able to simultaneously set states and functionalities of the cat toy based on the three sensors, enabling the speaker and treat dispensing either together or in discrete time periods.

In order to properly process the data from the multiple sensors to come to conclusions about the cat toy’s expected behavior, we plan to create a state diagram to implement a finite state machine (FSM). This implementation will allow us to handle the complex behavior that is required to be modeled based on the different sensors such as moving the tail, acceleration/deceleration, and dispensing treats through the drop-latch. As we receive input from our sensors, we can prioritize different signals through a FSM which can simplify the overall logic we will need to implement.

3. Cost and Schedule

3.1 Bill of Materials

The overall cost of this project is determined by the individual parts and the required quantity per part. The following table summarizes these part costs followed by a brief labor analysis and additional costs for 3D printing the toy shell and ordering the PCB design.

Part Name & Number	**Quantity Needed**	**Cost**	**Purchase Link**
7.4V 2600mAh Li-ion Battery	1	1 x $15.79ea = $15.79	7.4V Battery
AP9101CK BMS	1	1 x $0.59ea = $0.59	BMS
AZ1117 Voltage Regulator	3	3 x $0.38 = $1.14	Regulators
SW-420 Vibration Sensor	1	1 x $6.49 = $6.49 (5pk)	Vibr. Sensor
SG90 Servo Motor	2	2 x $6.99 = $13.98 (3pk)	Servos
KY-038 Sound Sensor	1	1 x $6.29 = $6.29 (5pk)	Sound Sensor
GA-12 N20 1000 RPM Motor	2	1 x $13.99 = $13.99 (2pk)	DC Motors
EK1794 Speakers	1	1 x $7.68 = $7.68 (2pk)	Speaker
ATmega328p Microcontroller	1	1 x $14.99 = $14.99 (2pk)	Arduino
DRV8848 H-Bridge	1	1 x $6.41 = $6.41 (2pk)	H-Bridge
VL53L1X Laser Sensor	1	1 x $17.99 = $17.99 (2pk)	Laser
CYT1091 Latching Button	1	1 x $6.99 = $6.99 (12pk)	Button
TOTAL	**N/A**	**$112.33**	**N/A**

The parts alone result in a cost of $112.33, which including a sales tax of approximately 9% results in a cost of $122.44.

Accounting for labor costs, we may assume a $48 hourly salary as the average in the United States for such engineering designs. In this project we assume three people are working on the project at a rate of about 8 hours each per week for 8 weeks. We therefore compute a total labor cost of $9,216.

In terms of additional costs, we may present the 3D printing of our shell as costing around $30, and the PCB order costing around $15. Accompanied by another $15 for potential replacement parts, we total the costs as such: $122.44 + $9,216.00 + $30.00 + $15.00 + $15.00 = $9,398.44.

3.2 Schedule

The following table contains our planned schedule for the remaining weeks and the work distribution alongside it.

Week	Overall Goals	Distribution
10/7	- Peer Review (10/8) - Design Review (10/9) - PCB Review (10/11)	1. Paul - Review microcontroller pins/start program 2. Yutong - PCB creation from circuit schematic 3. Rahul - Parts ordering and arrival estimation
10/14	- Team Evaluation - PCB Order Placement - Arduino Program Drafted	1. Paul - Full program outline completed 2. Yutong - PCB completed before order date 3. Rahul - Assist with PCB and analysis of it
10/21	- CAD Translation - Physical Model Planning	1. Paul - Begin moving 3D model to CAD 2. Yutong - Circuit placement planning for CAD 3. Rahul - Improve movement system in code
10/28	- Begin Parts Testing - Finish the Software - 3D Printing Prep	1. Paul - Finalize the software implementation 2. Yutong - Test arrived parts for functionality 3. Rahul - Research 3D print and best materials
11/4	- Toy Assembly - Shell Printing	1. Paul - Finish CAD model 2. Yutong - Assemble PCBs into toy's shell 3. Rahul - Get the model printed
11/11	- Revisions if Needed - Subsystems Testing	1. Paul - Code review if necessary 2. Yutong - Circuit review if necessary 3. Rahul - Subsystem testing/verification
11/18	- Mock Demo - Team Contract Fulfillment	1. Paul - Write team contract expectations/roles 2. Yutong - Prepare demo steps 3. Rahul - Write agenda/team issues
11/25	FALL BREAK	1. Paul - FALL BREAK 2. Yutong - FALL BREAK 3. Rahul - FALL BREAK
12/2	- Final Demo - Mock Presentation - Extra Credit Video	1. Paul - Presentation introduction/objective 2. Yutong - Presentation design slides (circuits) 3. Rahul - Presentation design slides (outputs)
12/9	- Final Presentation - Final Paper - Lab Checkout - Ensure Notebook Quality	1. Paul - Final report analysis/lab notebook check 2. Yutong - Final report drafting /lab notebook check 3. Rahul - Final report figures/lab notebook check

4. Ethics and Safety

4.1 Ethical Analysis

With regards to analyzing our project ethically, we are primarily following the IEEE Code of Ethics in order to ensure our design process goes smoothly[3]. We have also identified some codes from the ACM Code of Ethics in order to complement the IEEE codes, but also to provide more guidelines for us in technical considerations[4]. Our key foci revolve around upholding these specific codes, and notably the following sub-codes:

IEEE 1-5 To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors.

As we are doing a semester-long design project, we must note all problems and errors so we do not repeat mistakes, but also so we make our project the best it can be. Our time and budget is limited, so being open to acknowledging our errors will only help us get past them sooner. This will involve collecting and carefully monitoring real data during our development process. Furthermore, we must keep each team member's contributions clearly indicated so we can refer to them for help when errors arise, but also so we can give credit where credit is due. We intend to keep contributions and updates to our design in our notebooks, such that this process is well and easily streamlined.

IEEE 2 To treat all persons fairly and with respect, to not engage in harassment or discrimination…

We also must ensure each teammate feels welcomed with their ideas, and that we treat each other properly. All decisions should be discussed carefully and equally through either online Zoom meetings, text conversation, or in person. We have devised a group chat and have done well at coordinating any clarifications or changes we plan on making. This involves confirming with each member that our suggestions are okay and that they don't disrespect each other's work. This will avoid hiccups in our project and help us work more efficiently.

ACM 2-1 Strive to achieve high quality in both the processes and products of professional work.

We will furthermore maintain that our work is consistent with the best of our abilities, and that we respect any subjects or people involved with our work. This includes when we will playtest the toy with a cat, as our work quality will matter significantly to the safety of the cat. We can't leave any electrical connections exposed and should also make sure no parts can individually harm the subject.

ACM 2-6 Perform work only in areas of competence.

Finally, we are going to distribute work in a way that ensures the best odds of success. This means that if someone is more qualified than the rest of us at something specific, then they will take the lead and guide the rest. This is important for group safety and for the product's safety, but will also be more efficient in the long run as they can mentor the other groupmates, and we may learn faster. Important judgements should also only be made in competence, as they could lead to failures or setbacks in our project.

4.2 Safety Concerns

4.2.1 CPSC Regulations

The Consumer Product Safety Commission (CPSC) defines guidelines for toys and products in the United States. While these guidelines are for children's toys, we are choosing to follow some of them for our cat toy; pet toys are not federally regulated for safety[7]. Primarily speaking, we are following the toxicology guidelines, which state we must avoid using unsafe materials in terms of biohazards and bacteria[5]. This applies in our choice of filament for printing our toy shell, but also for our individual parts such as the lithium ion battery - as will be discussed in one of the next sections. The CPSC also defines electrical requirements which we will easily follow as we only have a maximum voltage of 5V being distributed to our parts[6]. Sound producing toys are addressed in these guidelines as well, wherein we will avoid high frequencies and volumes that may pose a danger to cats and people[5].

4.2.2 UIUC Lab Safety Guide

Our campus additionally provides a laboratory safety guide, which we have reviewed especially in the electrical domain. Since we are using a laser we had to identify the class of laser we are going to use, which is evidently a class 1 laser. This will be harmless and is the lowest class in terms of danger as it cannot produce any hazardous exposure effects[8]. All other electrical safety procedures were also looked over, and they provide general guidance for safely plugging in any devices we may need to use during our development.

4.2.3 Safety Practices

Since no federal or industry standards are set in place for pet toys specifically, we must also address the most dangerous parts of our toy by ourselves[7]. In our project, the lithium ion battery would pose the greatest safety danger to cats and people. Improperly charging our battery could create hazardous toxic gas, flames, or even an explosion. Thus, we are taking extra care to work with competence when installing our battery management system, which should handle the charging regulations for us. This will involve reviewing battery documentation and doing extensive research before soldering or powering any related components.

To prevent any dangers in general, all group members are going to follow the Safety Practice Guidelines provided on the course website as well. This includes following proper lab etiquette, and only working when someone else is present in the room. The proper way to charge and discharge batteries will also be reviewed from the lithium battery part datasheet.

References

[1] PetsRadar, "Best automated cat toys 2024 to keep your cat entertained," [Online]. Available: https://www.petsradar.com/buying-guide/best-automated-cat-toys. [Accessed: 19-Sep-2024].

[2] Pros Versus Cons, "Self-driving toy for your cat: Pros and Cons," [Online]. Available: https://pros-versus-cons.com/pets/pet-products/self-driving-toy-for-your-cat/. [Accessed: 19-Sep-2024].

[3] IEEE, "IEEE Policies - Section 7-8 - IEEE Code of Ethics," [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 19-Sep-2024].

[4] ACM, "ACM Code of Ethics and Professional Conduct," [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: 19-Sep-2024].

[5] U.S. Consumer Product Safety Commission, "Toy Safety Business Education," [Online]. Available: https://www.cpsc.gov/Business--Manufacturing/Business-Education/Toy-Safety. [Accessed: 19-Sep-2024].

[6] U.S. Government, "Requirements for Electrically Operated Toys or Other Electrically Operated Articles Intended for Use by Children," Electronic Code of Federal Regulations, [Online]. Available: https://www.ecfr.gov/current/title-16/chapter-II/subchapter-C/part-1505. [Accessed: 19-Sep-2024].

[7] Center for Pet Safety, "CPS Certified Products," [Online]. Available: https://www.centerforpetsafety.org/pet-parents/cps-certified/#:~:text=Pet%20Products%20are%20not%20classified,mandated%20to%20test%20their%20products. [Accessed: 19-Sep-2024].

[8] University of Illinois at Urbana-Champaign, "Laboratory Safety Guide," [Online]. Available: https://drs.illinois.edu/site-documents/LaboratorySafetyGuide.pdf. [Accessed: 19-Sep-2024].

Datasheets

VL53L1X Laser Sensor
Information:
https://www.st.com/en/imaging-and-photonics-solutions/vl53l1x.html
Datasheet:
https://www.st.com/resource/en/datasheet/vl53l1x.pdf

SW-420 Vibration Sensor
Information:
https://wiki.seeedstudio.com/Grove-Vibration_Sensor_SW-420/
https://docs.sunfounder.com/projects/ultimate-sensor-kit/en/latest/components_basic/04-component_vibration.html
Datasheet:
https://media.digikey.com/pdf/Data%20Sheets/Seeed%20Technology/Grove_Vibration_Sensor_SW-420_Web.pdf

KY-038 Microphone
Information:
https://sensorkit.joy-it.net/en/sensors/ky-038
Datasheet:
https://kirig.ph/wp-content/uploads/2020/08/KY-038-Joy-IT.pdf

ATmega328P Microcontroller
Information:
https://www.microchip.com/en-us/product/atmega328p
Datasheet:
https://ww1.microchip.com/downloads/Microcontrollers-ATmega328P_Datasheet.pdf

N20 DC Motors
Datasheet:
https://www.handsontec.com/dataspecs/GA12-N20.pdf

SG90 Servo Motors
Datasheet:
http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf

EK1794 Speaker
Information:
https://www.amazon.com/Gikfun-Speaker-Stereo-Loudspeaker-Arduino/dp/B01LN8ONG4

TP4056 Battery Management System
Datasheet:
https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf