



UNIVERSITY OF
ILLINOIS
URBANA-CHAMPAIGN

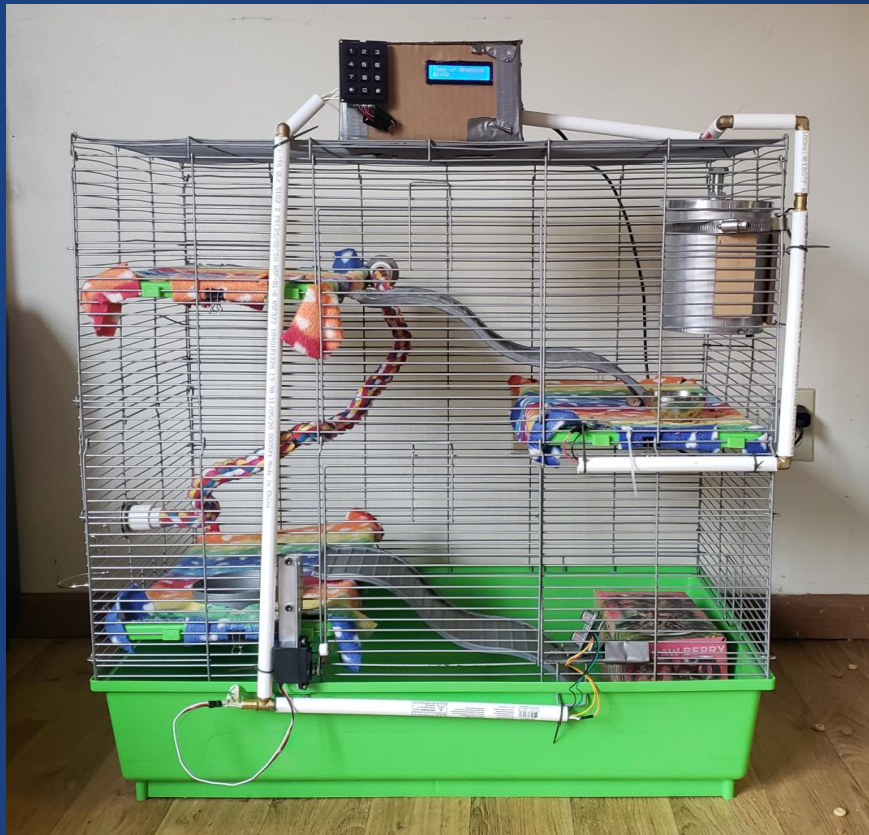
Team 16: Automated Pet Cage

Department of Electrical and Computer Engineering

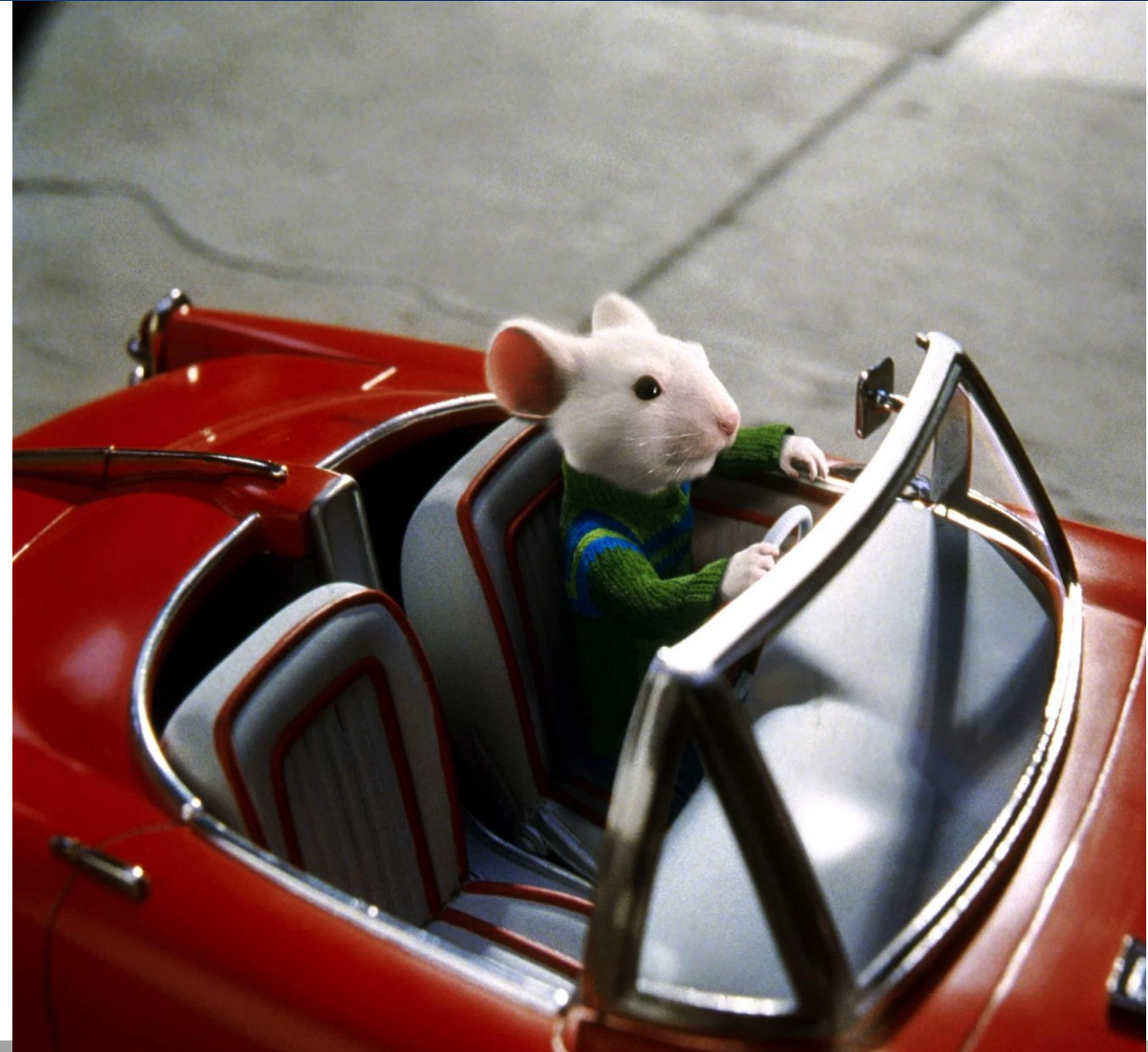
By: Avram Fouad, Christina Hejny, and Saurav Kumar

May 2nd 2022

Introduction



- Pet rodents typically need one to two hours of daily enrichment time.
- One of the main problems people with pet rodent face is that most of the time there is no one home to open the cage for the pet to have its daily enrichment time.





Objective

- To allow pet owners to set specific times during the day for the cage door to open.
- To create an incentive for the pet to come back to the cage, when predetermined enrichment time is over.
- To create a safe environment for the pet to use.

High Level Requirements



The device must be able to properly control the orientation of the cage door.

If the cage has been open for at least 2 hours [\pm 1 minute], the door must close within 3 seconds of the thin film pressure sensor activating.

The cage should not close during the designated 2 hour open-cage period under any circumstance unless the emergency “close” button is pressed.

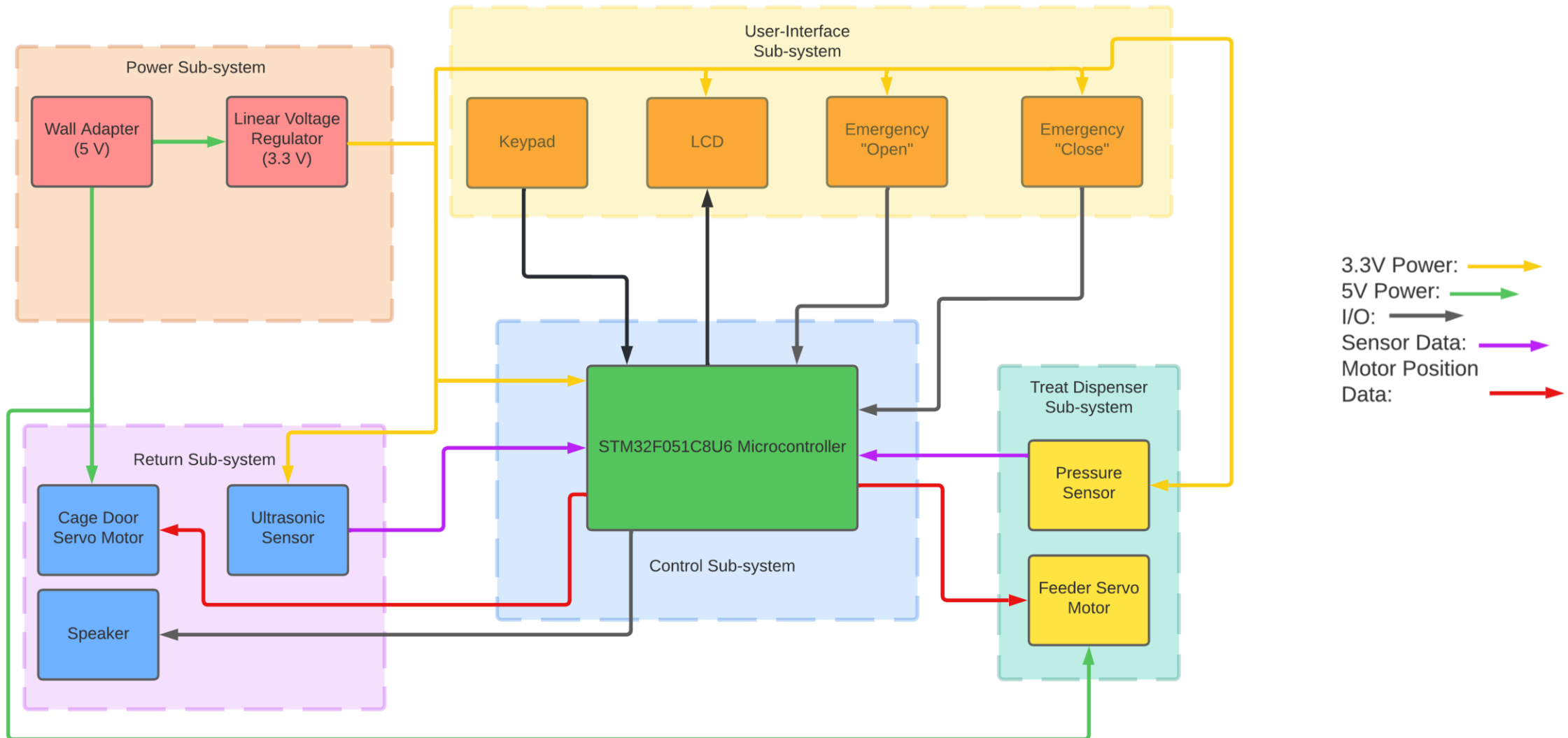






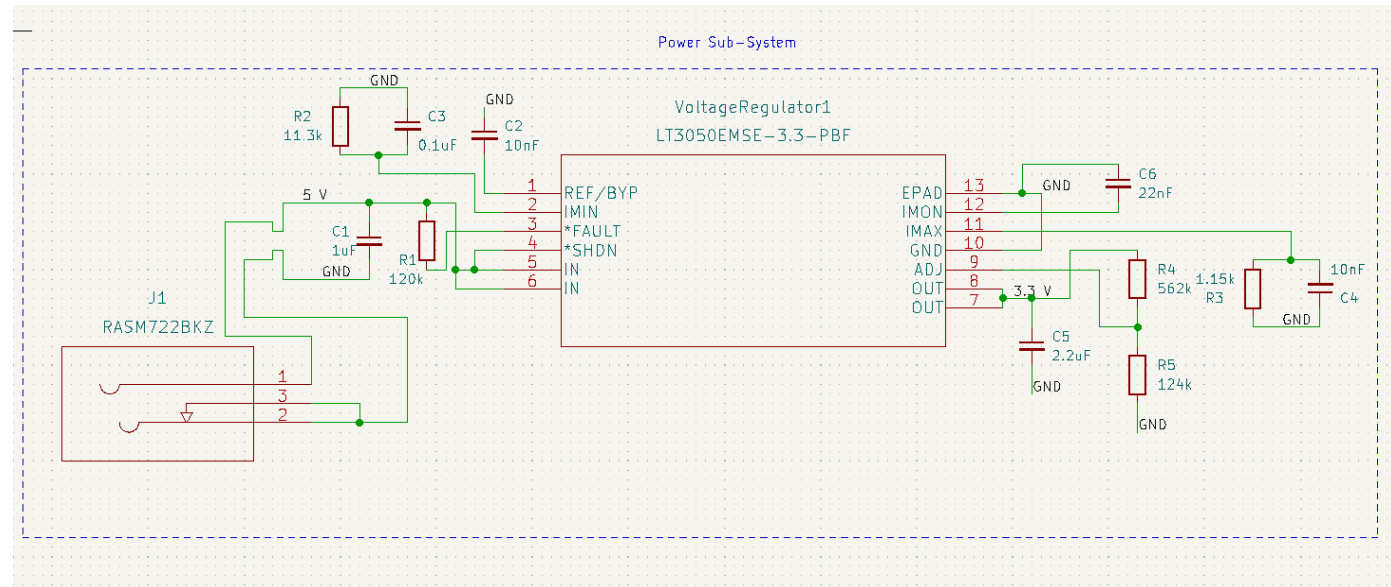
Design & Verifications

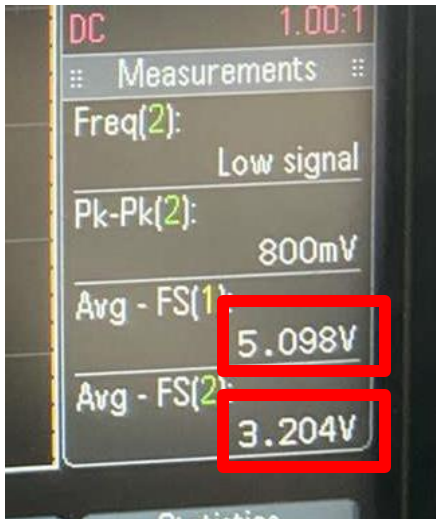
Block Diagram



Requirements:

- Must supply a 5 V [± 0.4 V] power line to the servo motors for currents up to 3 A
- The 5 V input source must be stepped down to 3.3 V [± 0.3 V] for currents up to 200 mA



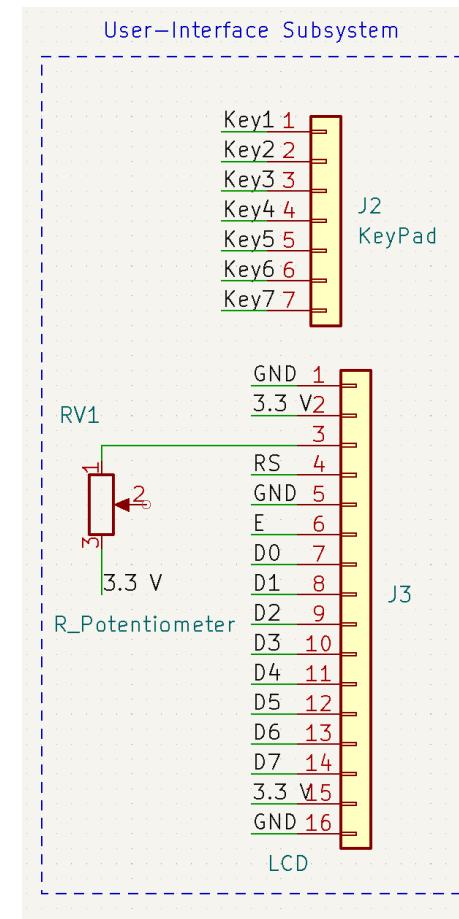


The Power Subsystem showed complete functionality

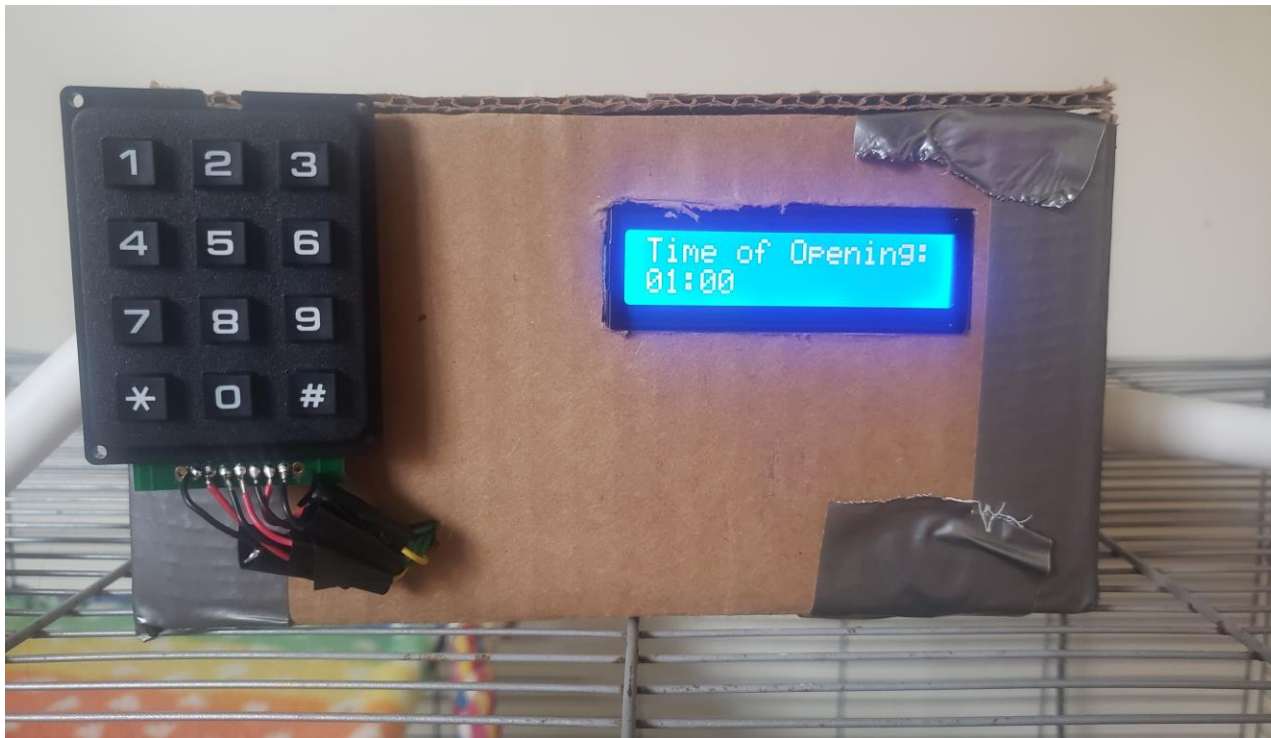
- The power subsystem successfully provided 3.3 V +/- 6% as well as 5 V +/- 8%.
- The 3.3 V power line can provide up to 200 mA to the rest of the subsystems.
- The power subsystem functioned as designed to fulfill the power requirements for all the other subsystems.

Requirements:

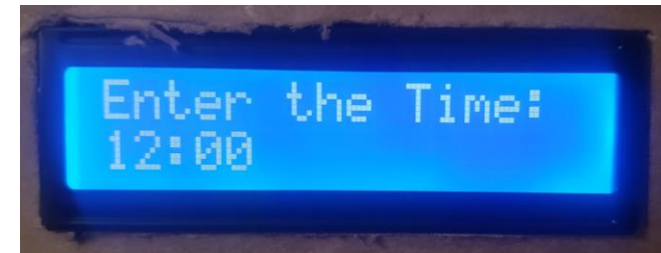
- The LCD display must correctly display the desired open time that is entered by the user on the keypad
- The user should be able to manually open/close the cage door immediately in case of any emergency events.



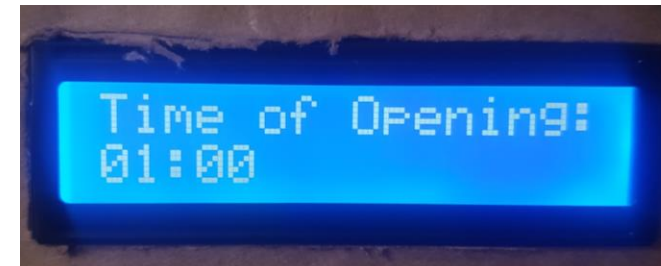
The User-Interface Subsystem showed complete functionality



At start-up



After setting clock
to 12:00



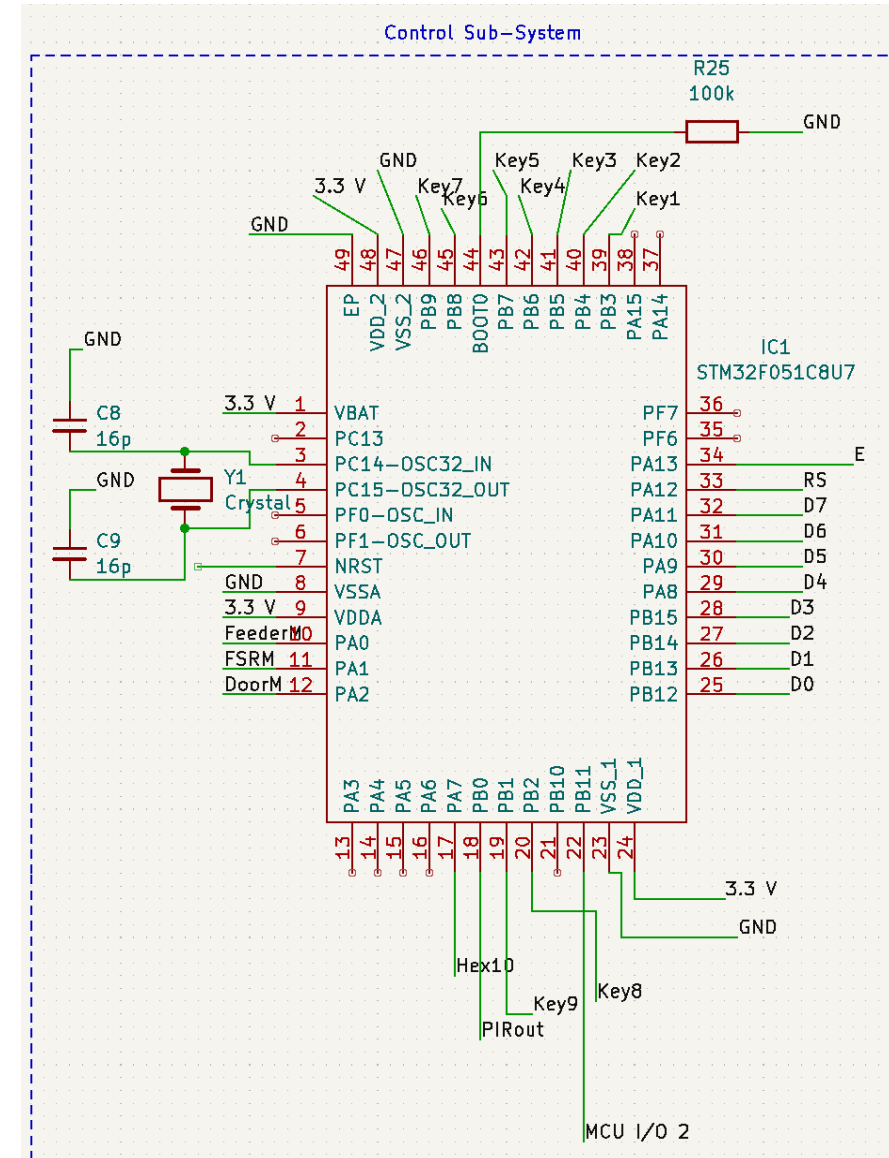
After setting opening
time to 1:00



After clearing
opening time

Requirements:

- Must keep track of real time [± 30 seconds]
- Must be able to send correct I/O signals to LCD and Feeder/Door servo motors
- Must be able to correctly process user input from keypad



The Control Subsystem did not show complete functionality

- The control subsystem did not function properly.
- We were unable to program the microcontroller due to incorrect pin assignments.
- However, we were able to show complete functionality using an Arduino dev board, which verifies the software component functions properly.

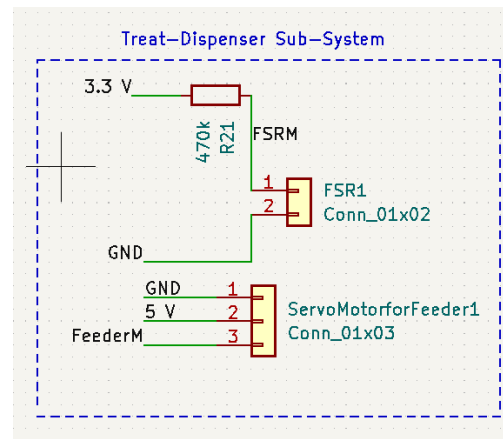
COM8

```
+2 hours { 21:52:35.340 -> The cage is now open!  
+15 min { 23:52:30.449 -> Beep! Come Back!  
          { 00:07:29.837 -> Beep! Come Back!  
          { 00:07:37.573 -> Pressure Sensor Triggered29  
          { 00:07:37.618 -> 29  
          { 00:07:37.618 -> 29  
          { 00:07:37.618 -> 30
```



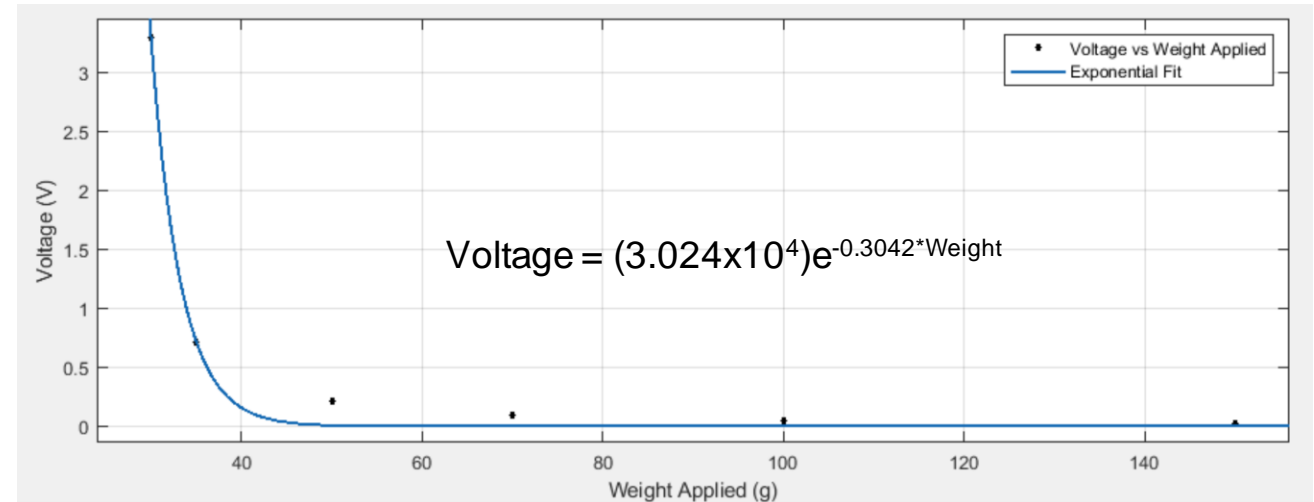
Requirements:

- The pet rodent must be able to activate the feeder and close the door by stepping on the pressure sensor once it returns to the cage after being outside for 2 hours
- The feeder should not dispense food and the door should not close for any weight applied to the pressure sensor below 50 g
- The feeder should not dispense food if the emergency close key is pressed



The Treat Dispenser Subsystem showed complete functionality

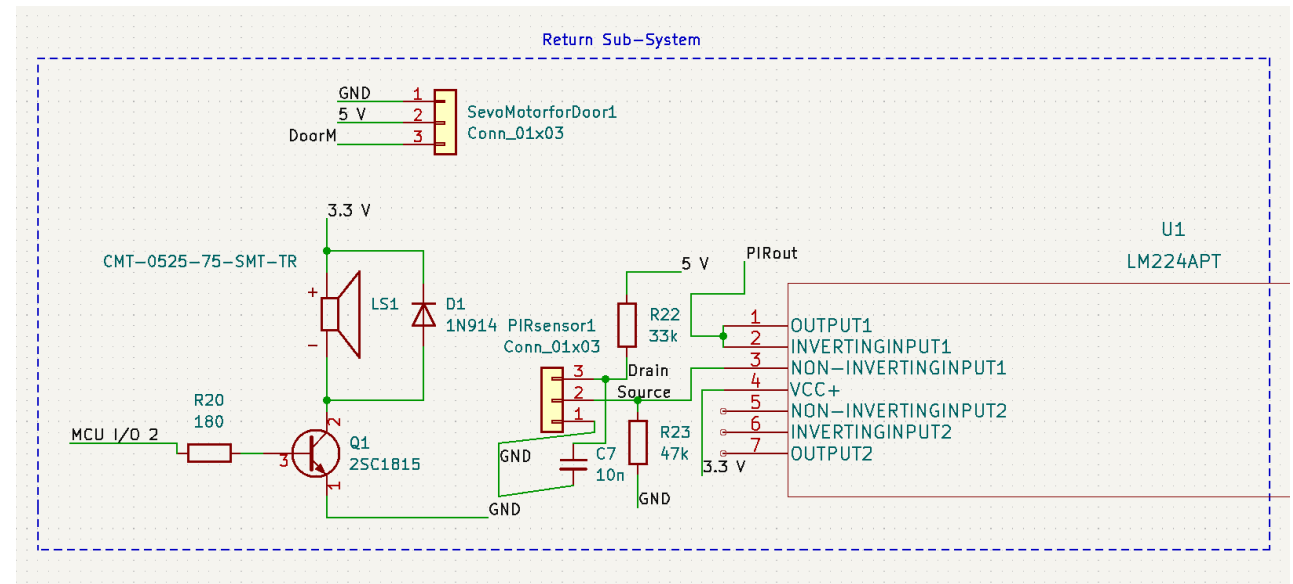
- The pressure sensor is consistently triggered by a pet rodent and not triggered by debris inside the cage.
- The feeder correctly dispenses food and the door closes once the rodent activates the pressure sensor
- Any weight below 50 grams applied to the pressure sensor does not activate the feeder nor the cage door

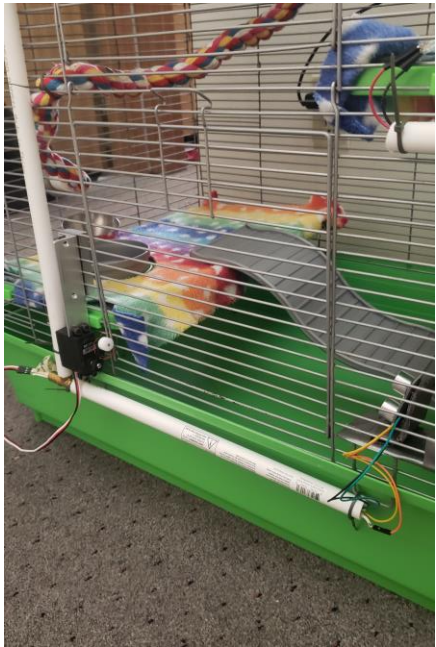


Weight Applied (g)	FSR Output Voltage (V)	Feeder Dispensed/Door Closed?
30	3.29	No
35	0.71	No
50	0.21	No
70	0.10	Yes
100	0.045	Yes
150	0.026	Yes

Requirements:

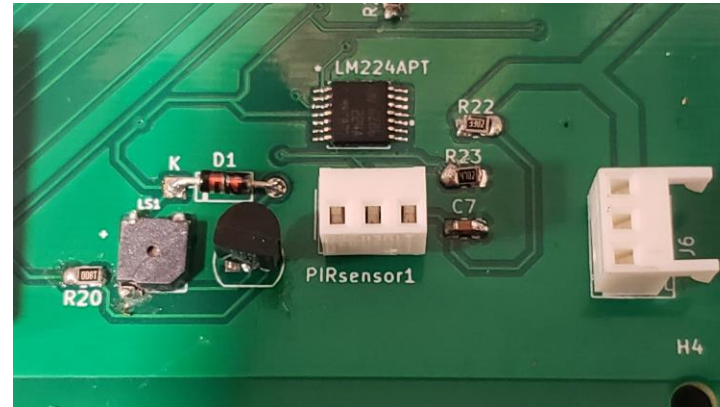
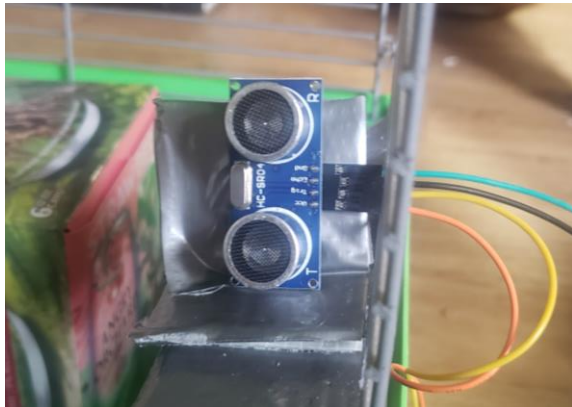
- The cage door servo motor must be able to fully open and close the cage door
- The speaker must output a "beep" 2 hours $[+/- 30 \text{ s}]$ after the door opens. It should then play this every 15 minutes until the pet comes back
- The ultrasonic sensor must be able to detect any movement near the door so that the cage door doesn't close on the pet



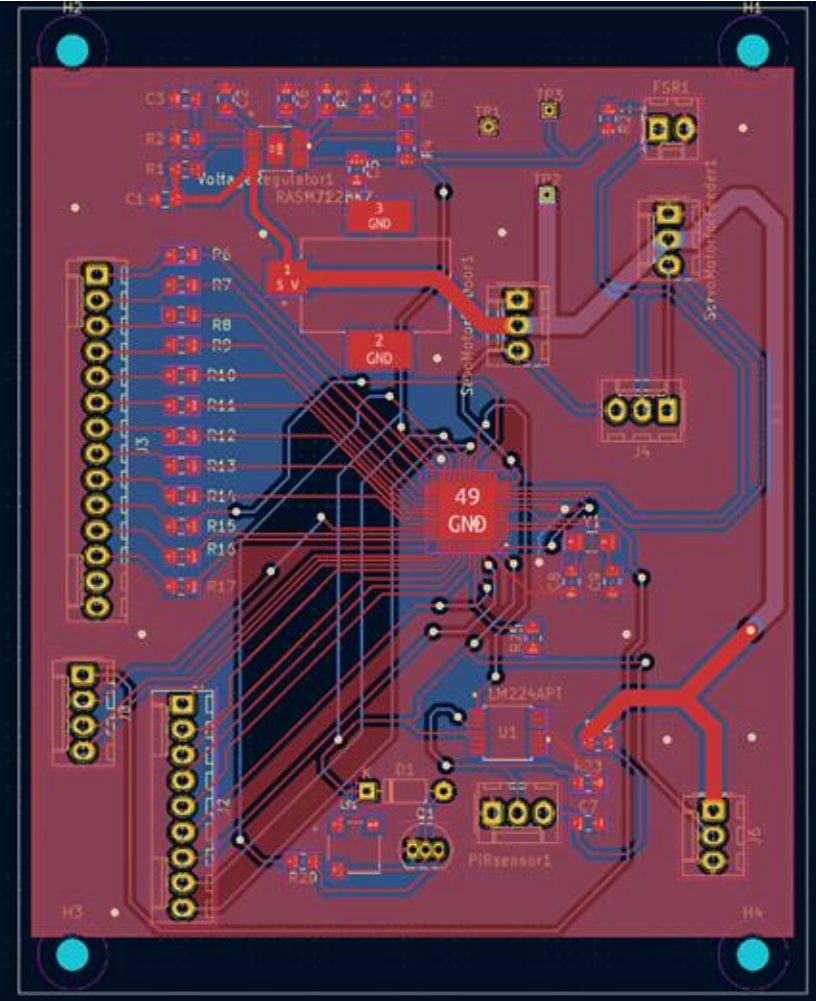
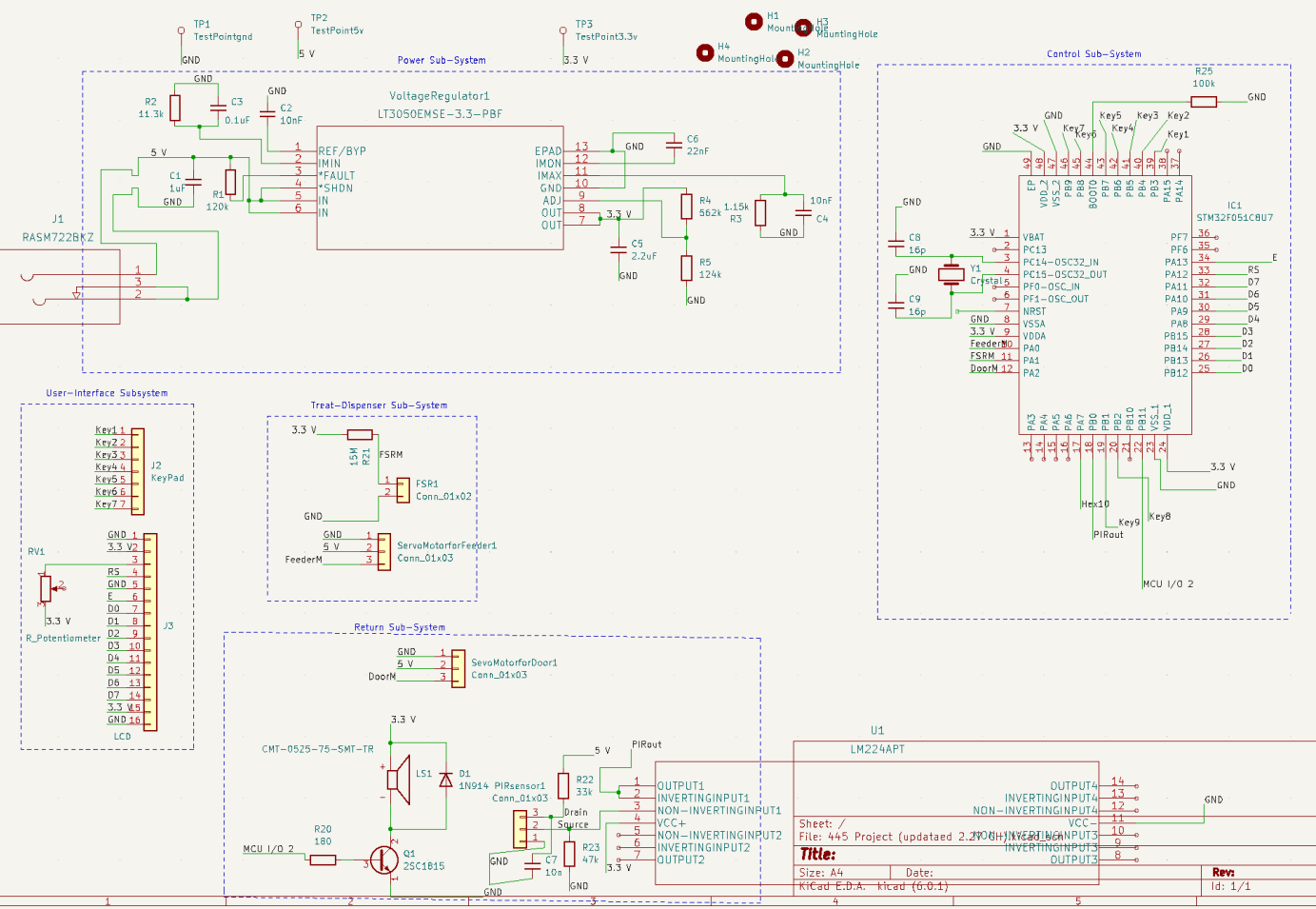


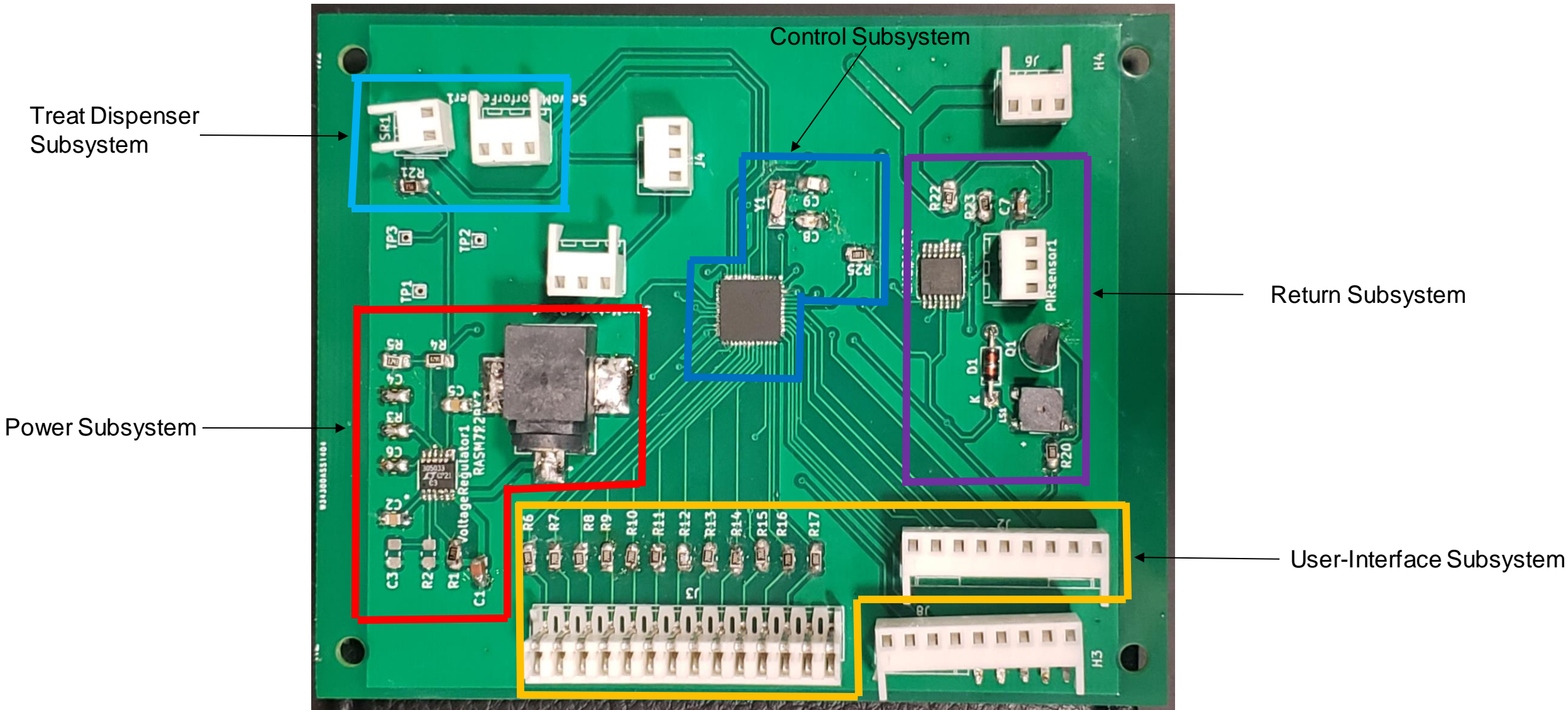
The Return Subsystem showed complete functionality

- The door opens and closes properly
- The speaker correctly beeps 2 hours after the cage opens as well as every 15 minutes after.
- The Ultra Sonic sensor detects objects in the doorway and stops the motor without fail.



Circuit Schematic and PCB Design



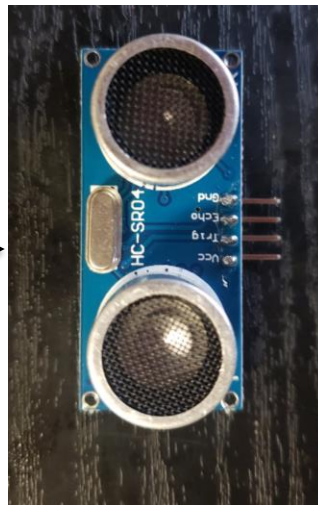
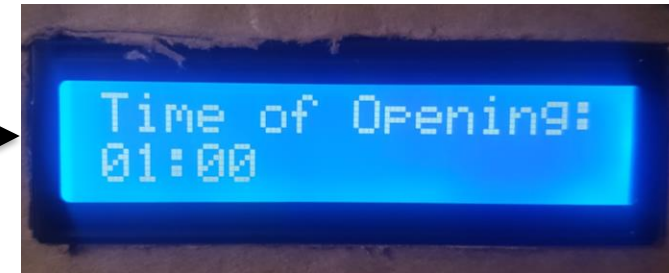




Challenges, Successes, and Failures

Our initial design used a PIR sensor to detect objects in the cage doorway.

- However, the PIR sensor did not perform reliably.
- To overcome this, we used an Ultra Sonic Sensor instead which proved to reliably work.



Our initial design used a 7-segment display to display the time.

- The microcontroller could not refresh the display while performing any other function.
- To overcome this, we used an LCD instead.
- This became a benefit since we could now also display text.

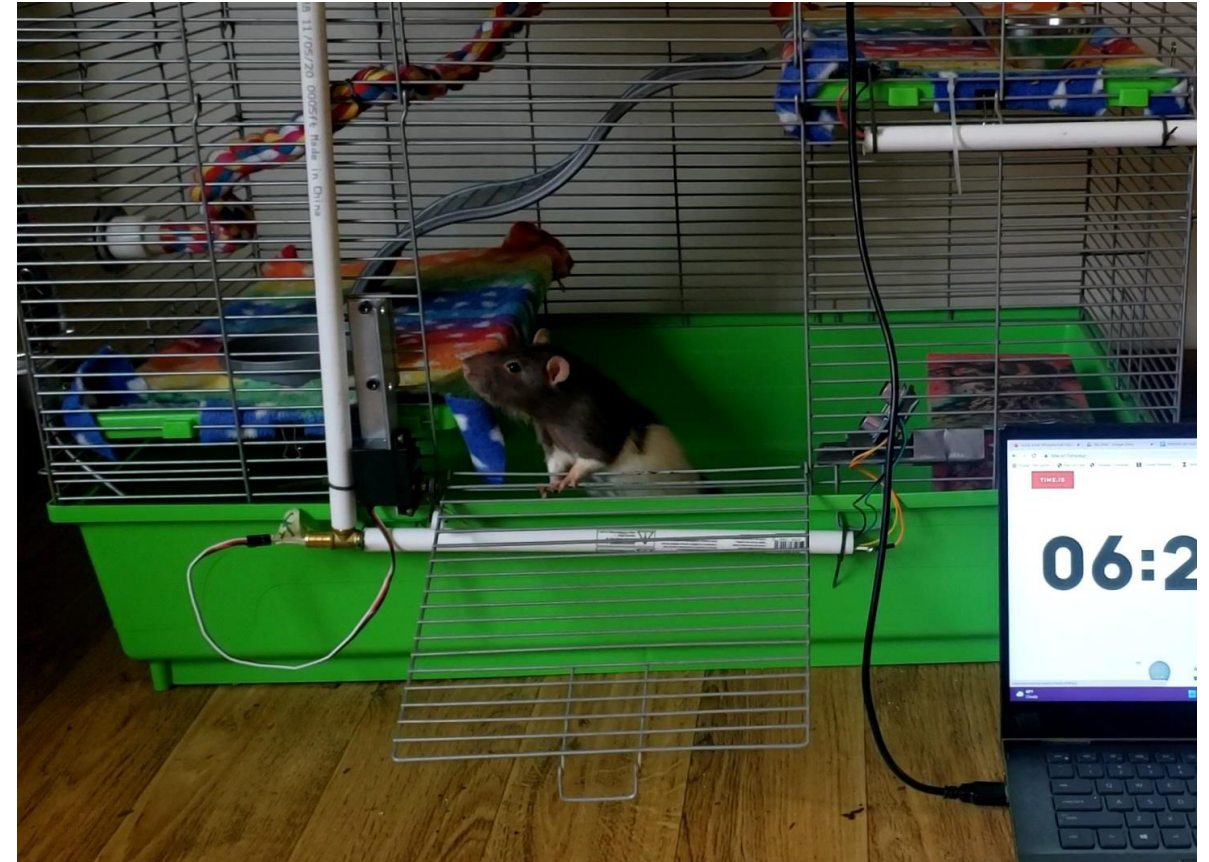
We had issues with our software not being able to perform all the functions simultaneously.

- We determined the problem was the many “delay();” functions used.
- Rewriting the code to instead use “millis();” functions solved this issue.

```
long timer = millis();  
tone(piezopin, 800, 2000);  
  
while (analogRead(A3) > 150)  
{  
    long timer1 = (millis());  
  
    if((timer1) > timer + 60000*15){  
        Serial.println("Beep! Come Back!");  
        tone(piezopin, 800, 2000);  
        timer = abs(millis());  
    }  
}
```

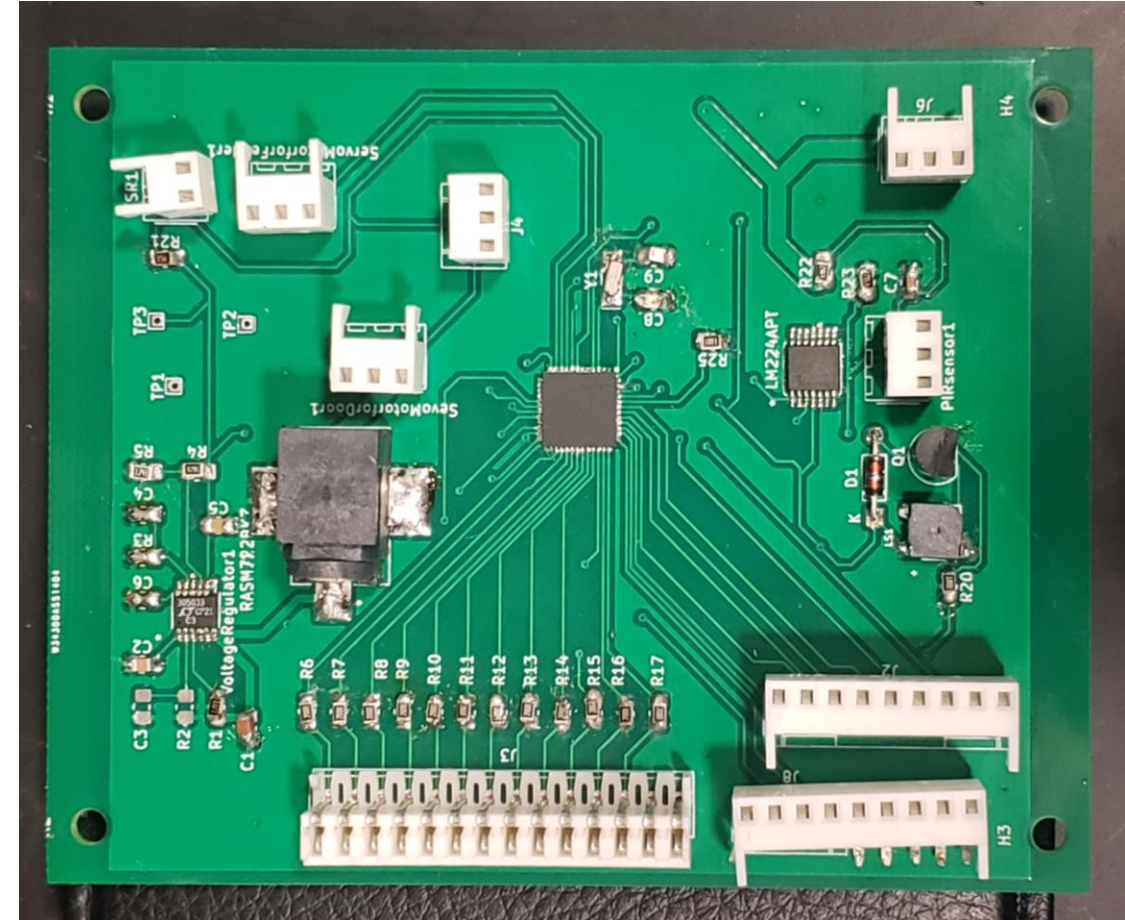

The device successfully performs all main functions

- Keeps track of real time
- Beeps at 15-minute intervals
- Closes cage when pressure sensor is triggered
- Stops closing cage door if doorway is obstructed by an object
- Dispenses food via feeder after cage door is closed



Our project failed to utilize a working PCB for the Final Demonstration.

- We designed the PCB to program the microcontroller using SPI pins instead of the Serial Debug Wire.
- We fixed this issue on the 2nd PCB order, but we were unable to find an ST-Link in time to program the microcontroller.
- The microcontroller, STM32F0, cannot be programmed with the Arduino IDE.



There are multiple things we would do differently now

- We would choose a different microcontroller that can be programmed with Arduino IDE. (ATMEGA)
- We would learn how to properly program it before designing the PCB.
- Preferably pick a microcontroller that can be programmed using JTAG.
- We gained a lot of experience in project management, designing circuits and PCBs, SMD soldering, writing software for a microcontroller.

- Use RFID tags for multiple pets in the same cage
- A sturdier enclosure (steel or plastic)
- Ability to monitor the cage remotely
- Make applicable to different kinds of pets



Acknowledgements

- Special thanks to our TA Hanyin, Gregg and Skee from the Machine Shop, and Professor Fliflet and the rest of the ECE 445 Team
- Questions?