Automatic Wet Food Cat Feeder

Damera Gebissa, Christian Pellerito, Alaina Wayne, Julia Carroll, Jacob Morrison

Department of Mechanical Engineering, Kettering University

Dr. Craig Hoff

December 23rd, 2024

Automatic Wet Food Cat Feeder

MECH 495: Team 3 Senior Design Project Final Report

Cats are beloved companions, but their feeding schedules often clash with the busy lives of their human counterparts. Whether it's an early morning demand for breakfast or a long day at work leaving a furry friend unfed, managing a cat's diet can become a challenging task. The problem isn't just about convenience — it's about ensuring consistent portions, proper timing, and a stress-free feeding experience for both owner and pet. While automated feeders for dry food are widely available, they fail to address the needs of cats that thrive on wet food diets. Wet food provides essential hydration and nutrients but poses challenges in storage, portioning, and freshness. This prototype tackles this gap by developing an innovative automated feeder specifically designed for wet food, combining reliability and functionality to keep meals fresh and portions precise. All with the aim to enhance the quality of care for cats and simplify life for their owners.

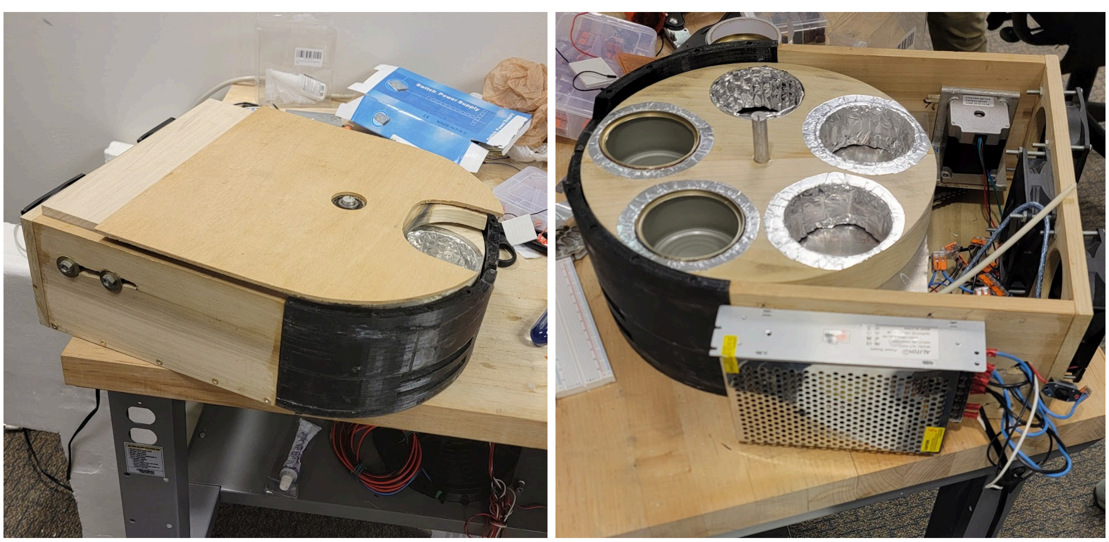


Figure 1: (a) Fully assembled feeder with lid and (b) without lid

Initial Design Concepts and Design Goals

This prototype aimed to use the whole can of cat food so the owner wouldn't have to scoop the food out of the can. This use of the can was a cornerstone of the design that the rest of the project was built around. The next issue was the rotating of the cans and how to attach the Peltier devices. It was settled on using a stationary plate that the cans rotate atop of - this eliminated the need for slip rings, which would have been required had the Peltier device coolers also spun. The rest of the design was to maximize airflow to cool the Peltier devices. There were two primary issues with the design. One of these issues was that the shell was too small for the electrical components, specifically the power supply. The second issue was the placement of the drive motor. This was solved by attaching the motor to the side of the shell upside down so the chain wouldn't interfere with the Peltier devices. The goal was to make a cat feeder that dispensed and kept the wet cat food cold, which was ultimately successful.

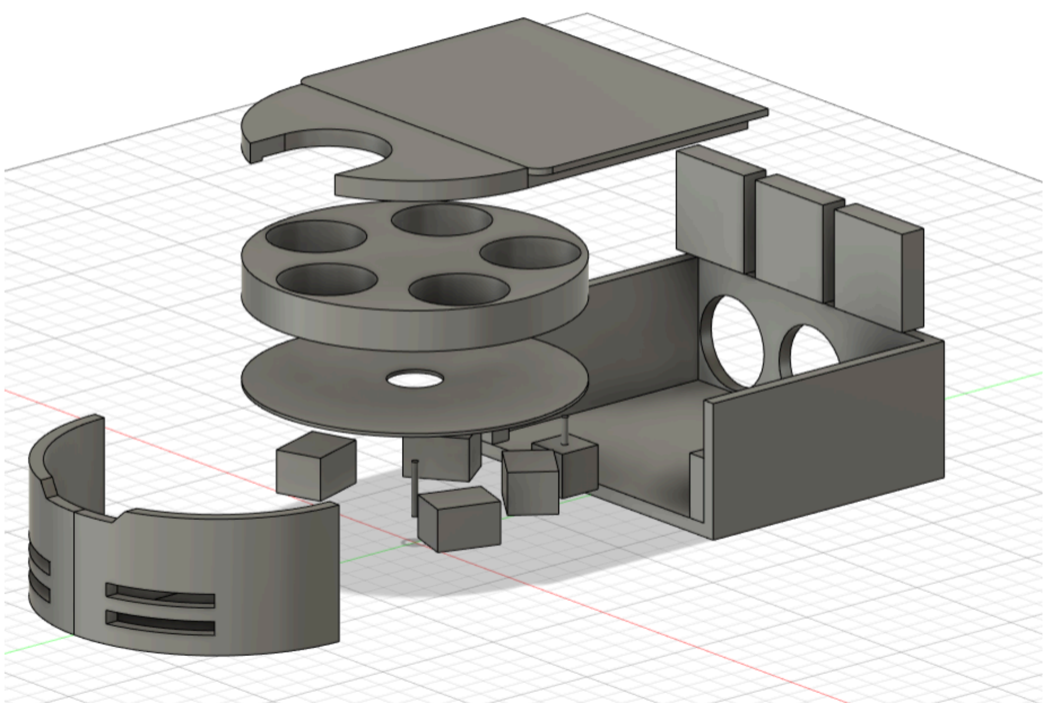


Figure 2: Exploded view showing main components.

Design Goals:

- **Effective Cooling**: Maintain a low temperature to preserve wet food freshness.
- **Rotational Accuracy**: Ensure precise rotation to dispense one portion of food at a time.
- **Power Efficiency**: Power supply optimized for required loads; ensure economic power draw over time.
- **User-Friendly Operation**: Includes simple interface for dispensing and easy to refill feeder.
- **Durable and Safe Construction**: Use food-safe materials and ensure all components are secure.
- **Compact and Aesthetic Design**: Create a visually appealing and space-efficient feeder.
- **Ease of Maintenance**: Ensure the system is easy to disassemble for cleaning and repairs.
- **Thermal Efficiency**: Isolate hot and cold zones effectively to maximize cooling performance.
- **Mechanical Reliability**: Design a robust dispensing mechanism to prevent wear over time.
- **Future Expandability**: Allow for potential integration of smart features.

Components and Materials

The construction of the automatic wet food cat feeder relied on a combination of repurposed components, group-owned materials, and custom-manufactured parts. This resourceful approach allowed the team to improve sustainability while still meeting the project's design requirements. Below is a comprehensive list of quantities of materials and devices used in the construction of the cat feeder prototype:

Table 1: Bill of materials (items marked N/A are already owned, or being recycled from prior projects)

Material/Component	**Amount (#)**	**Projected Cost (USD)**	**Actual Cost (USD)**
3.5-in Fans	3 pcs	$15.27	N/A
Peltier Device	4 pcs	$14.99	N/A
Heat Sinks	4 pcs	$8.99	N/A
Wood	1in.x 12 in.x 8 ft.	N/A	N/A
Arduino Uno Controller	1 pc	$16.99	N/A
TB6600 Stepper Motor Driver	1 pc	$10.89	$10.89
Stepper Motor 23HS22-2804S	1 pc	$25.99	N/A
Assorted Wires	-	N/A	N/A
Thermal Cutoff Switch	1 pc	$8.99	N/A
Silicone Dielectric Compound	.05 oz	N/A	N/A
Double Reflective Insulation	1.5x11in x 5 pc	$0.34	$0.34
Reflective Foil Tape	½ Roll	$2.49	$2.49
Spray Foam Insulation	2 oz	$0.74	$0.74
Loctite	½ tube	$3.29	$3.29
Liquid Nails	1 oz	$0.99	$0.99
CA Glue	-	N/A	N/A
2-part Epoxy	⅛ tube	$1.53	$1.53

PETG Filament	1 kg	$14.99	$14.99
6x2mm Neodymium Magnets	6 pcs	$0.24	$0.24
120V AC to 12V DC Power Supply	1 pc	$19.49	$19.49
Hall-Effect Sensors	3 pcs	$1.05	$1.05
Lever Wire Connectors	8 pcs	$3.20	$3.20
Dev Button	1 pc	N/A	N/A
Power Switch	1 pc	N/A	N/A
9V Battery	1 pc	N/A	N/A
AC Power Cable	1 pc	N/A	N/A
Aluminum Sheet Metal	128.5 sq in	N/A	N/A
Aluminum Rod	1 pc	N/A	N/A
¼” Machine Screws	16 pc	$6.08	N/A
¼” Nuts	16 pc	$2.19	N/A
¼” Washers	16 pc	$0.96	N/A
Brass Rod	1 pc	N/A	N/A
25 Pitch Roller Chain	1.5 ft	$2.54	N/A
Martin 25B9	1 pc	$12.97	N/A
Martin 25BS14-1/2	1 pc	$5.40	N/A
Brass Plated Wood Screws	19	$1.82	N/A
Wood Glue	-	N/A	N/A
Ball Bearings	2 pcs	$1.09	N/A
1.5” Wooden Dowels	8 pcs	N/A	N/A
Total Cost	**-**	**$183.51**	**$59.24**

Manufacturing Process

The manufacturing of the automatic wet food cat feeder involved a combination of advanced CNC machining and hands-on craftsmanship in the woodshop. This hybrid approach allowed for precise fabrication while leveraging the team's practical skills and available resources.

Fabrication of the Wooden Box:

- **CNC Machining**:
  - The primary structure of the wooden box was manufactured using a Laguna IQ CNC machine. This ensured precise cuts and dimensions for the panels, creating a robust and accurately assembled structure.
  - CNC programming was based on CAD models designed earlier in the project, ensuring the panels fit seamlessly during assembly.
- **Woodshop Tools**:
  - Additional adjustments and fine-tuning of the wooden panels were made using tools in the woodshop, such as saws, sanders, and drills. This allowed the team to address minor discrepancies and ensure a perfect fit for all components.

Fabrication of the Aluminum Sheet:

- **Material Preparation**:
  - The aluminum sheet, central to the cooling system, was carved to the correct dimensions using a combination of tin-snips and Dremel sanding. The sheet in

particular required work to ensure proper alignment with the Peltier devices and surrounding structure.

- Sanding and polishing were performed on the Peltier device contact surfaces to create a clean medium for heat conduction. This was further assisted by the presence of thermal compound between heat transfer materials.

Assembly of the Box:

- **Joining Techniques**:
    - The box was assembled using a combination of nuts, bolts, glue, and epoxy:
        - **Nuts and Bolts**: Used to secure removable parts, providing structural integrity while allowing for future disassembly.
        - **Glue and Epoxy**: Applied to permanently bond fixed sections of the box, especially where strong, airtight seals were required.

All joints were inspected and tested for stability before moving to the next phase.

Heat Transfer Considerations for the 3D-Printed Housing:

1. **Objective:**

   To optimize the thermal performance of the 3D-printed housing, ensuring sufficient heat dissipation and insulation to maintain operational stability and preserve food freshness.

2. **Methodology**:

   The design of the housing prioritized thermal resistance, cooling efficiency, and structural integrity. Key considerations included:

3. **Material Selection:**
    - Black PETG 3D printing filament was chosen for its superior thermal resistance and mechanical strength compared to alternatives like PLA.
    - White filament was considered for better heat reflection but deprioritized in favor of strength, durability, and heat absorption of black PETG.
4. **Structural Design:**
    - Increased wall thickness
    - Ultimaker Cura Gyroid infill pattern was chosen for the walls to maximize heat dissipation
    - Ultimaker Cura Tri-hexagon infill pattern was chosen for the bottom wall to improve heat dissipation as well as increase structural integrity
    - A 20-30% infill density struck the optimal balance between material usage and thermal performance.
    - Ventilation holes, in combination with active cooling, improved thermal management without compromising the structural strength of the housing.

Integration of Components:

- **Cooling System Installation**:
    - The aluminum sheet was mounted inside the box with Peltier devices, heat sinks, and fans attached. Proper alignment and thermal paste ensured effective heat transfer.
    - Wiring for the fans and Peltier devices was routed through pre-drilled holes, secured with clamps to prevent tangling or disconnection.
- **Dispenser Mechanism**:
    - The wooden food holder was mounted on a chain-driven rotating system, powered by the motor. Alignment was critical to ensure smooth rotation without jamming.
    - The 3D-printed front cover was carefully positioned to allow only one food portion to be visible at a time.

Challenges and Solutions:

- **Dimensional Accuracy**:
  - Minor adjustments to the CNC-machined panels were made manually using woodshop tools to ensure a good quality of fit.
- **Adhesion Strength**:
  - Additional layers of epoxy were applied to strengthen joints in high-stress areas, such as the base of the housing where components were mounted.
- **Aluminum Sheet Fitment**:
  - Precise carving and sanding of the aluminum sheet ensured compatibility with the wooden box and thermal components.

There were a series of issues pertaining to the construction of the cat feeder prototype. The 3D printed housing components required multiple redesigns, reprints, and reassemblies in order to achieve the desired build quality. Achieving good quality fitment between the CNC produced wood housing and the 3D printed housing sections was met with great difficulty, and required a significant amount of care with hand tools to modify and improve the parts. In addition, properly mounting the motor and tensioning the chain was met with difficulty. There was a lack of foresight in regards to tensioning said chain. It was determined that instead of using a specific idler sprocket on the chain assembly for tensioning, it would be best to instead mount the motor using a slot. By mounting the motor with a slot, the chain tension could be easily adjusted by moving the motor. In the future, it would be prudent to spend more time in the design phase of development.

Applying what was learned to a more thorough design process could aid in the construction of a second prototype immensely.

Calculations and Testing

Heat Transfer Calculations:

To ensure that the insulation used in the cooling system effectively maintained a low temperature, heat transfer calculations were performed. These calculations focused on minimizing heat ingress into the cooling chamber while ensuring efficient heat dissipation from the Peltier devices.

1. **Objective**:
   - To calculate the thermal resistance of the insulation materials and determine the heat loss through the walls of the cooling chamber.
2. **Methodology**:
   - The heat transfer equation was applied, where Q is the heat loss, ΔT is the temperature difference, and Rtotal is the total thermal resistance of the system.
   - The thermal resistance of each layer (e.g., wooden box, spray insulation, aluminum sheet) was calculated using:

$$R = \frac{Thickness}{Thermal\ Conductivity} * Surface\ Area$$

   - The calculated Q was compared with the cooling capacity of the Peltier devices to ensure the system could maintain a desired temperature range.

3. **Results**:
   - Preliminary calculations indicated that the insulation effectively minimized heat ingress, reducing the load on the Peltier devices.
   - Final adjustments to insulation thickness and material placement were made based on these results.

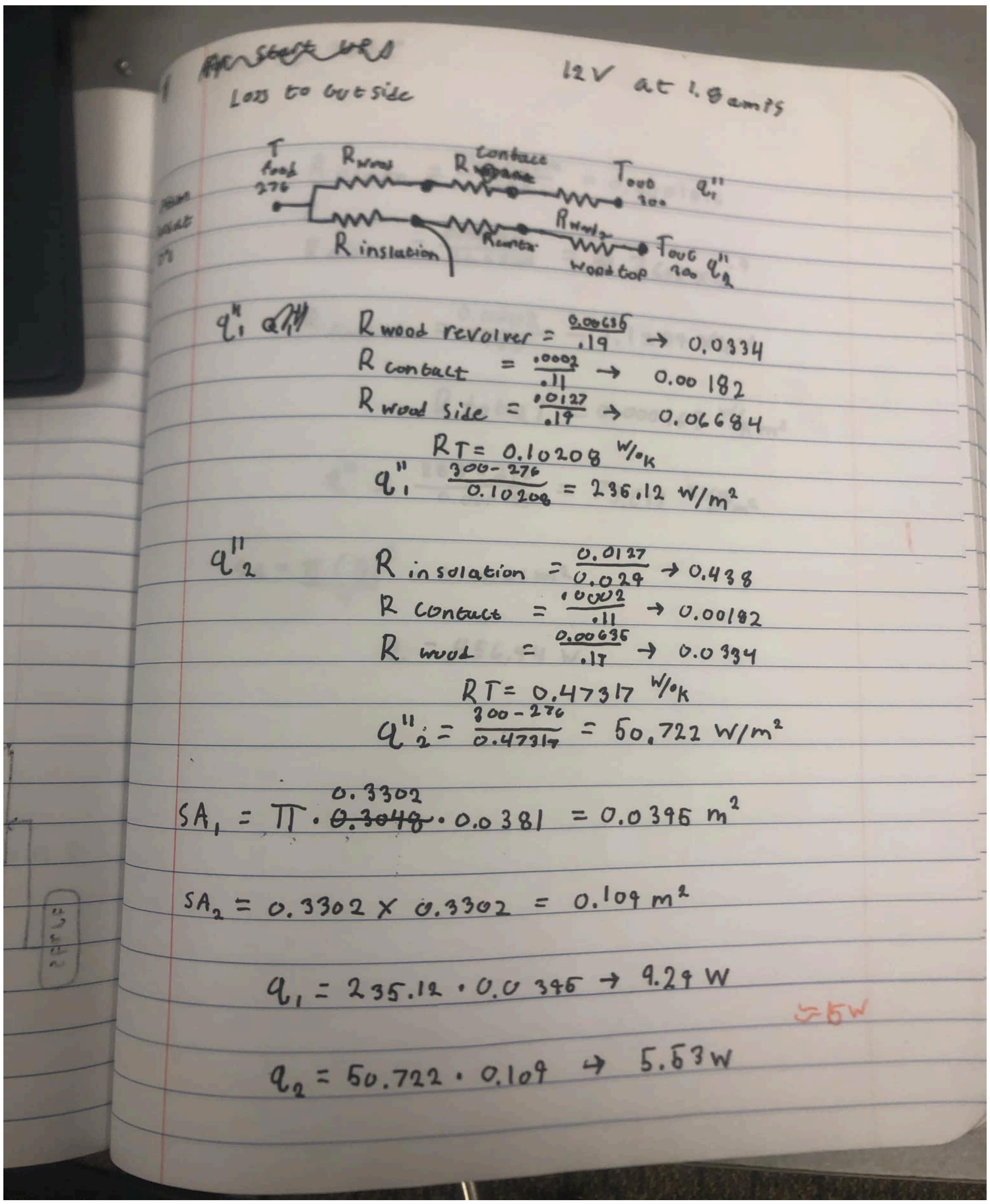


Figure 3: Thermal circuit depicting X-Y heat loss of the food can, along with surface area calculations.

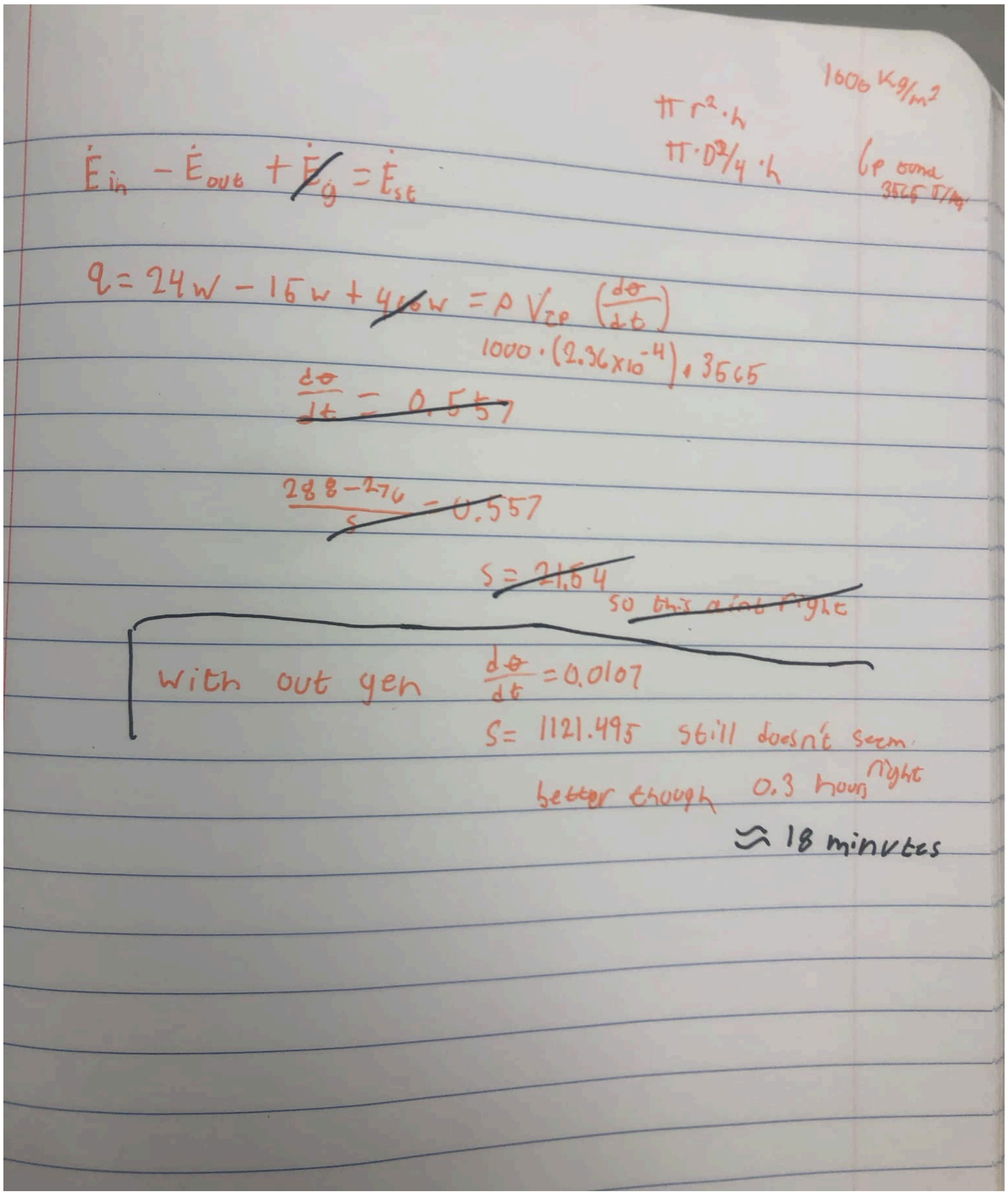


Figure 4: Time to cool calculations for the food to get from 70°F to 37°F. This is a basic calculation, and a more in-depth one will follow based on information from the current lecture in MECH-420 class.

Motor Calculations:

1. **Objective:**

   To select and validate a stepper motor capable of providing sufficient torque and precision to rotate the turntable accurately while maintaining durability and efficiency.

2. **Methodology:**

   Motor selection involved calculating the required holding torque to rotate the turntable assembly. Factors such as weight distribution, rotational inertia, and resistance were considered. Key parameters included:

   - Dimensions and weight of the turntable
   - Required angular displacement
   - Friction coefficients of the rotating components
   - Safety factor for operational reliability
   - Potential gear ratios

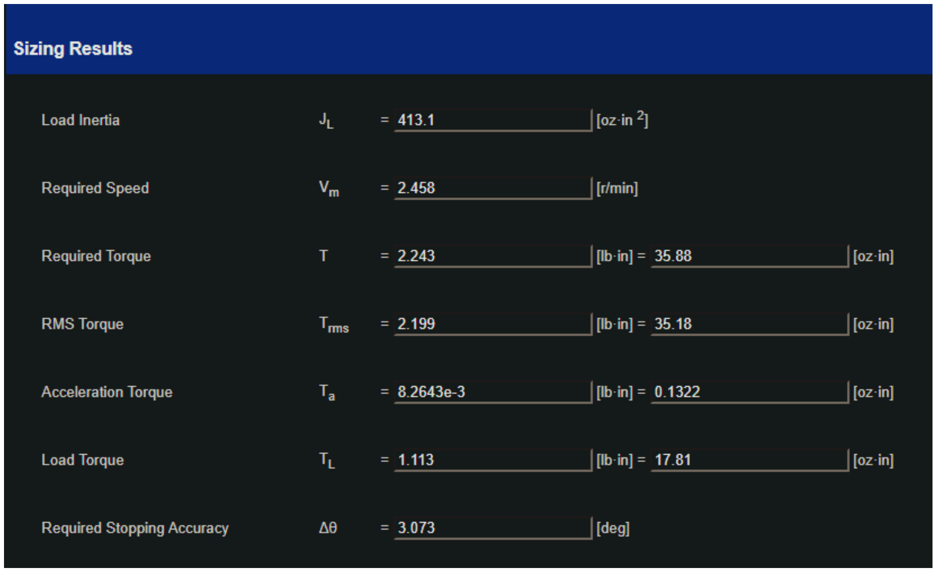
   Calculations were performed using the Motor Sizing Tool from Oriental Motor. Based on these evaluations, the 23HS22-2804S stepper motor was chosen for:

   - Its torque rating of 170 oz.in (0.89 ft-lbs.), which exceeded the calculated requirements.
   - Compatibility with the system's 12V power supply.

3. **Results:**

   The selected motor provided:

   - A sufficient safety margin, ensuring reliable operation without stalling or excessive power consumption.
   - Compatibility with the calculated torque demands and power supply constraints.
   - Efficient performance, maintaining durability and precision for the dispensing mechanism.



Sizing Results

Load Inertia	J_L	= 413.1	[oz·in ^{2}]		
Required Speed	V_m	= 2.458	[r/min]		
Required Torque	T	= 2.243	[lb·in] =	35.88	[oz·in]
RMS Torque	T_{rms}	= 2.199	[lb·in] =	35.18	[oz·in]
Acceleration Torque	T_a	= 8.2643e-3	[lb·in] =	0.1322	[oz·in]
Load Torque	T_L	= 1.113	[lb·in] =	17.81	[oz·in]
Required Stopping Accuracy	Δθ	= 3.073	[deg]		

Figure 5: Results of motor calculations, including required torque

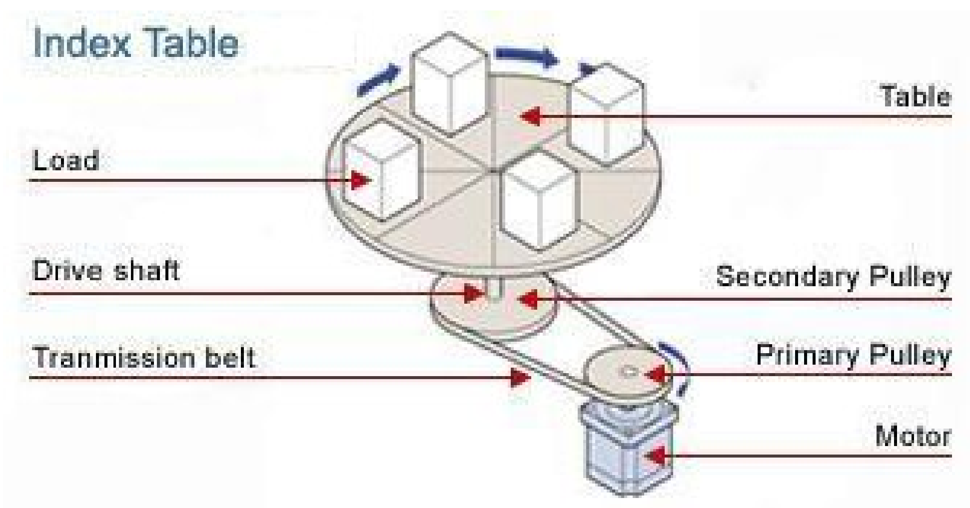


Figure 6: Diagram of the index table used for torque and inertia calculations

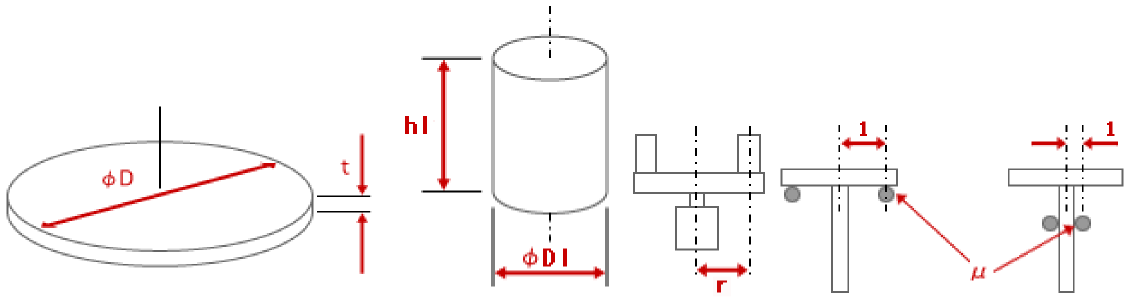


Figure 7: Parameter diagrams illustrating inputs for the calculations

Testing Ideal Height for Cat Feeder:

1. **Objective:**

   To determine the optimal height for the food dispenser that ensures comfort for cats while maximizing the available space for internal electronics within the feeder.

2. **Setup**:

   A simple testing rig was constructed using variable-height cardboard boxes to simulate different feeder heights. The setup included:

   - Feeding bowls regularly used by the test subjects.
   - Two test subjects representing the 75th and 95th weight percentiles of domestic cats were used.
3. **Testing Procedure**:

   The cats were observed while eating from bowls placed at two different heights:

   - 8 inches: Comfort and ease of access were assessed for both cats.
   - 10.5 inches: The upper height limit was evaluated for universal usability. Posture and accessibility were used as indicators of comfort.
4. **Results:**
   - At 8 inches: Both test subjects comfortably accessed the food without strain.
   - At 10.5 inches: The 75th percentile cat did not attempt to eat. The 95th percentile cat managed but showed noticeable discomfort, stretching to reach the bowl.
5. **Conclusions:**

   An optimal height of 8 inches was determined to provide maximum comfort for cats. A maximum height of 10 inches is acceptable for larger cats but may compromise usability for smaller ones.

Figure 8: At 8 inches, both test subjects comfortably access the food without strain

Figure 9: At 10.5 inches, the 95th percentile cat eats with some exertion, while the 75th percentile cat does not attempt to eat

Testing Cooling Rate of Pre-Built Peltier Fridge:

1. **Objective:**

   To evaluate the cooling rate inside of a pre-built Peltier fridge, optimized for cooling in a small space, to ensure that the Peltier devices could be an effective method of cooling wet food to a food safe temperature and maintain said temperature with minimal variability.

2. **Setup**:

   The pre-built Peltier fridge (Figure 10) was equipped with:

   - A single Peltier device integrated with a heatsink and fan, and powered by a 120V AC power supply at 38W.
   - A temperature and humidity sensor placed inside the fridge to monitor performance over time (Figures 11 and 12).

3. **Testing Procedure**:

   The system was activated at ambient conditions (78.1°F, 44% RH). Temperature and humidity changes were recorded over a 3-hour duration to assess cooling performance.

4. **Results:**
   - Initial Conditions: 78.1°F, 44% RH
   - Final Conditions: 44.1°F, 66% RH
   - Temperature Drop: ΔT, 34.0°F
   - Humidity Increase: ΔRH, 22%

5. **Conclusions:**

   Although the system cooled quite slowly and did not reach food-safe temperatures during this test, the results confirmed the feasibility of using Peltier devices for direct cooling of individual cans. This finding led to the design choice of incorporating four Peltier devices in the final system to improve cooling capacity and uniformity. Using pre-refrigerated cans would improve efficiency. Further testing is required to validate the effectiveness of this approach for maintaining safe food temperatures over extended periods.

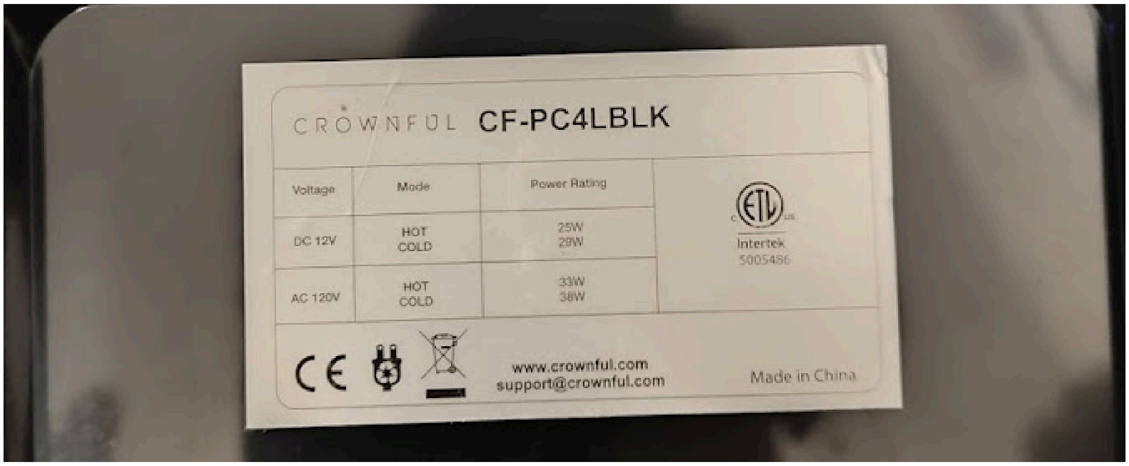


Figure 10: Model and specifications of the pre-built Peltier fridge.

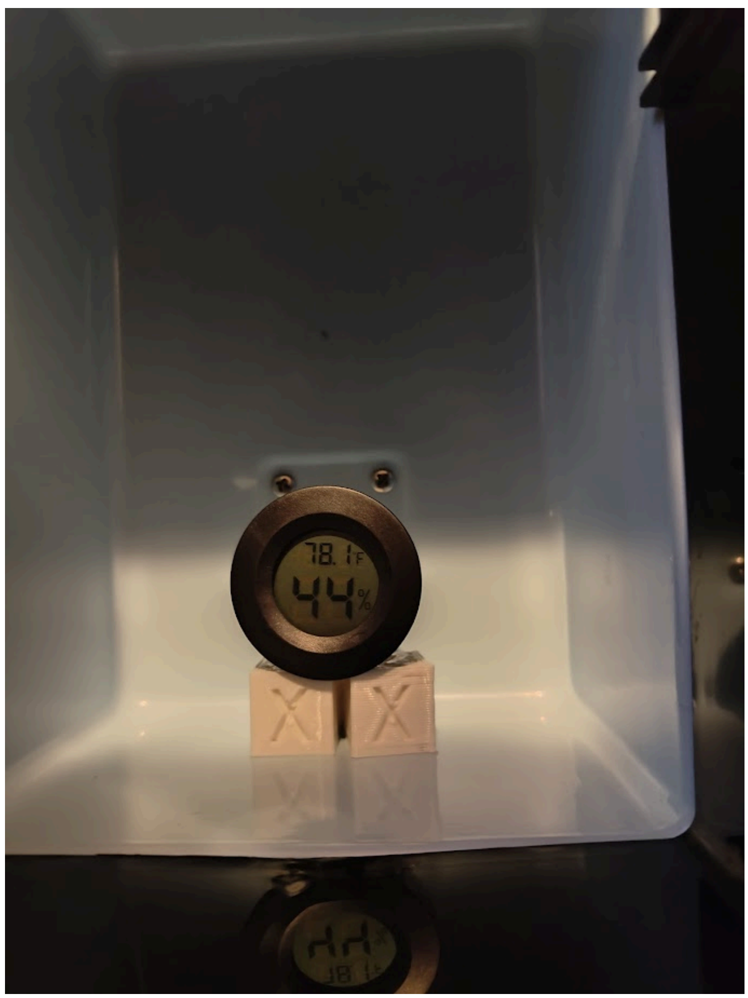


Figure 11: Testing setup and initial measurements

Figure 12: Final measurements after a duration of 3 hours

Testing the Peltier Devices:

The Peltier devices were tested on a separate circuit to verify their functionality and cooling performance before integration into the system. This testing ensured they could create the required temperature differential under expected operating conditions.

1. **Setup**:
    - The circuit was powered by a regulated DC power supply, allowing precise control over voltage and current.
    - Each Peltier device was connected to the circuit with proper heat sinks and fans on the hot side to dissipate heat.
    - Temperature probes were placed on both the hot and cold sides of the device to measure the temperature gradient.
2. **Testing Procedure**:
    - The circuit was powered with a constant voltage and current matching the expected operating conditions in the final design.
    - Temperature readings were taken at regular intervals to monitor the performance of the devices.
    - The devices were tested under varying load conditions to observe their stability and cooling capacity.
3. **Results**:
    - The Peltier devices achieved a significant temperature drop on the cold side while maintaining thermal stability on the hot side with proper heat dissipation.
    - This validated their suitability for the project, ensuring they could meet the cooling requirements when integrated into the system.

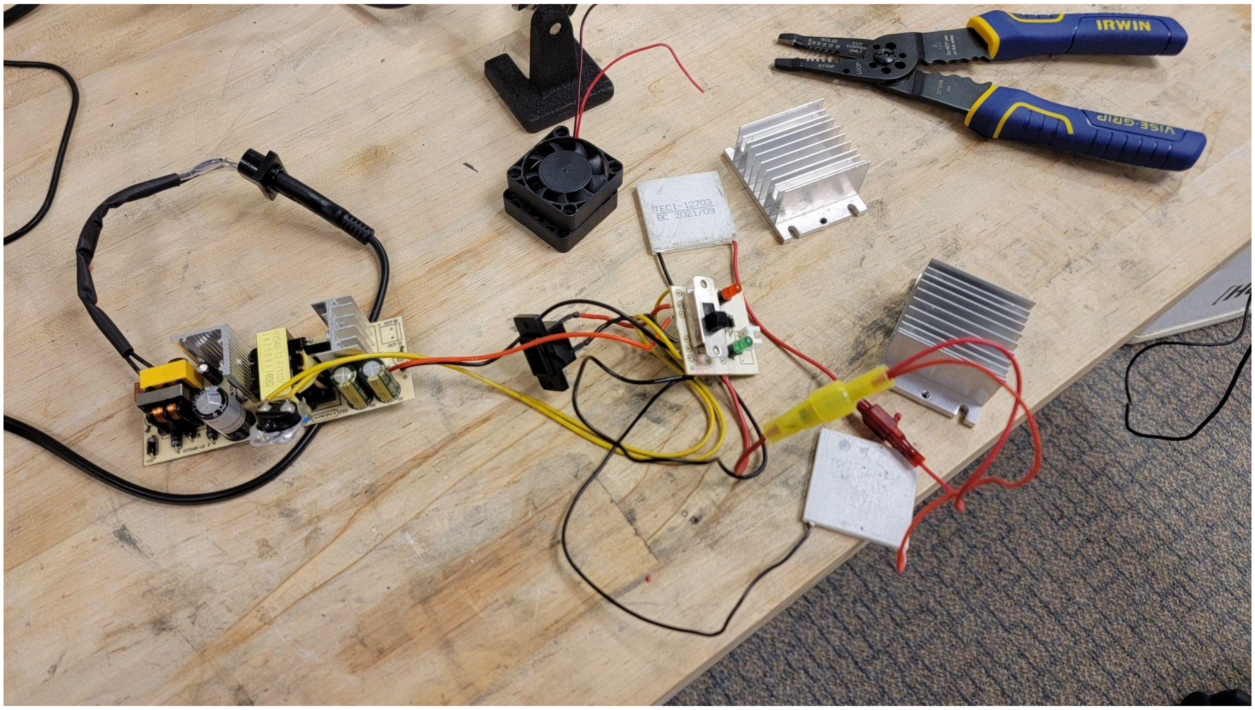


Figure 13: Peltier thermoelectric device testing circuit

Electrical:

- **Power:**
  - 12V Power supply for most components
  - 9V battery for Arduino
- **Fans:**
  - 3 large fans
  - 4 small fans on heatsink
  - All wired in parallel
- **Peltiers:**
  - 4 Peltiers
  - Wired in parallel
  - A switch in series
  - A thermal fuse in series for safety
- **Motor:**
  - Stepper motor (Bipolar)
  - Stepper motor driver (12V)
- **Hall Effect Sensors:**
  - 2 side by side at same radial distance (on a ring with 5 evenly spaced magnets) this is to detect rotation direction and change in position
  - 1 offset radial distance (with only 1 magnets) to detect “home” position

- **Arduino:**
  - Connected to hall effect sensors
  - Connected to stepper motor driver inputs
  - Added dev button for demo purposes

Schematics:

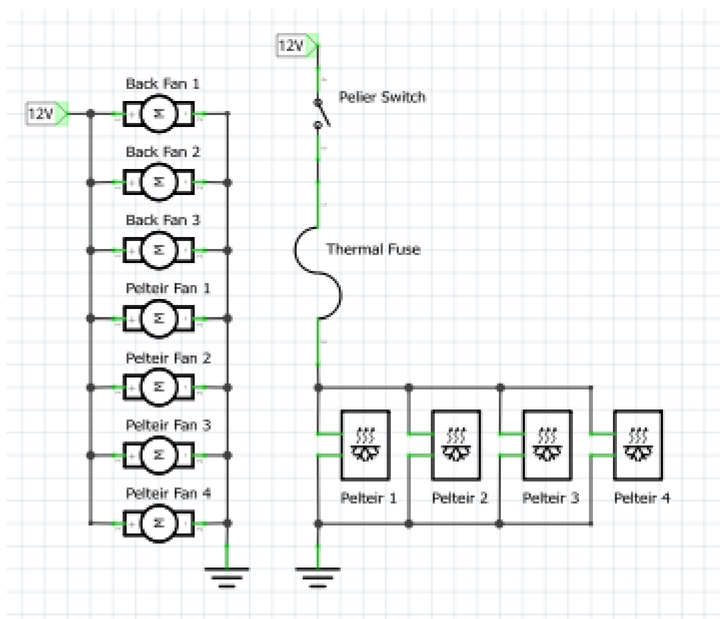


Figure 14: Schematic for fans and peltiers. Peltiers have a switch and a thermal fuse for safety.

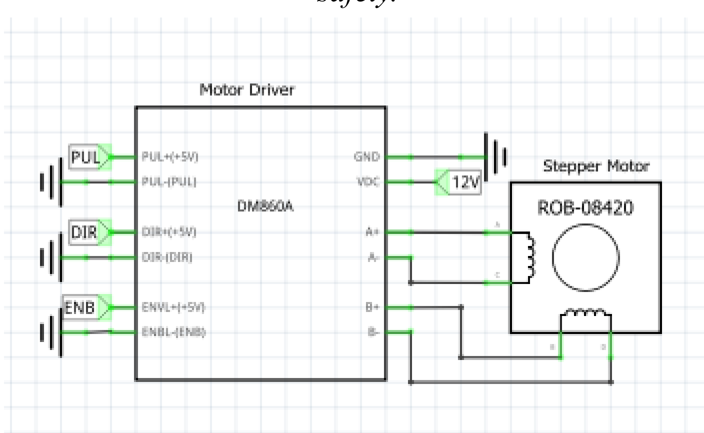


Figure 15: Schematic for motor driver and stepper motor.

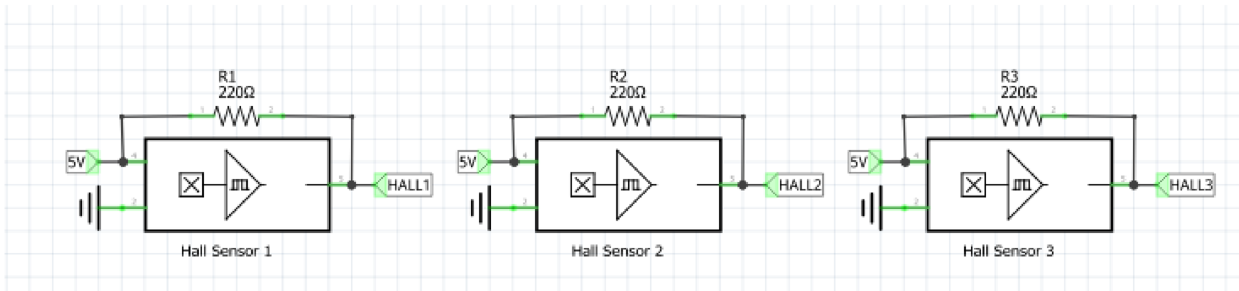


Figure 16: Schematic for hall effect sensors.

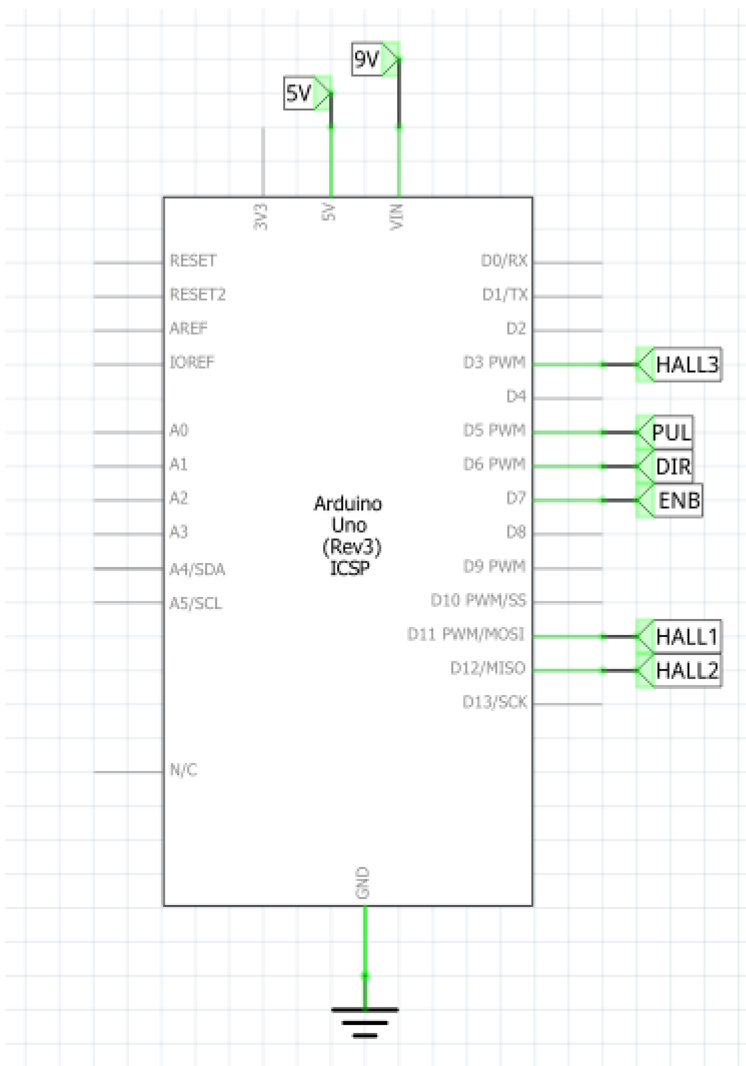


Figure 17: Schematic for connection to Arduino and power. Note, the DevButton is only there for demo purposes.

Software:

The plan for the code is relatively simple. All constants and variables are declared at the start to avoid confusion later. All pinmodes settings are made in the startup routine. Then for the main bulk of the code, a state machine is used.

There will be four states: **WaitHome**, **MoveFeed**, **WaitFeed**, and **MoveHome**.

WaitHome is the state where the feeding is at the "home" position, which is the position when food isn't available. In this state, the machine does nothing other than wait for an input. For the demo, the input will be a button. But in real operation the input would be a timed schedule. Once the state receives an input, the state machine goes to the state **MoveFeed.**

MoveFeed is the state where the feeder moves to the next available can. To do this, the code tracks which cans have been used and which have not, and will move in the rotation that is the fastest. To know if the machine has reached the can, the system used magnets and 2 hall effect sensors to behave as an encoder to track direction and change in position. Once the machine has reached the desired can, the state machine goes to the state **WaitFeed**.

WaitFeed is nearly identical to WaitHome, with the main change being that once this state receives the proper input, the state machine goes to the state **MoveHome**.

MoveHome is a state where the feeder returns to the "home" position. To do this, it looks at its current location, and rotates in the direction of the "position" that would be faster. While rotating, the stem is constantly checking the offset hall effect sensor that can only detect 1 magnet. This magnet and sensor is placed in such a way that whenever this sensor detects the magnet, that means the system is in the "home" position. Once the system is homed, the state machine goes to the state **WaitHome**.

Code:

```
const bool HALL = 0;
const int pulsePerRev = 1300;
const float gearRatio = 2;
int motorDelay = 2000;


void updateEncode();
void stepMotor();

int state = 0;
int pos = 0;
int prevEncode[] = { 0, 0 };
int currEncode[] = { 0, 0 };
int canLocation;
int pulseMove;
int i;
int canDistance;

const int hallPin[] = { 11, 12, 3 };
const int enbPin = 7;
const int dirPin = 6;
const int pulPin = 5;

const int devButPin = 9;

bool homeDir = HIGH;
bool usedCans[] = { 0, 0, 0, 0 };
int numSlots = 5;
int numCans = numSlots - 1;
```

Figure 18: Declaration stage of code.

```
void setup() {
  Serial.begin(9600);

  pinMode(enbPin, OUTPUT);
  pinMode(dirPin, OUTPUT);
  pinMode(pulPin, OUTPUT);
  pinMode(hallPin[0], INPUT);
  pinMode(hallPin[1], INPUT);
  pinMode(hallPin[2], INPUT);
  pinMode(devButPin, INPUT_PULLUP);

  // Set the spinning direction CW/CCW:
  while (digitalRead(devButPin) == HIGH) {}
  digitalWrite(dirPin, HIGH);
  while (digitalRead(hallPin[2]) != LOW) { stepMotor(); }
  updateEncode();
}
```

Figure 19: Initialization stage of code.

```
void stepMotor() {
  digitalWrite(pulPin, HIGH);
  delayMicroseconds(motorDelay);
  digitalWrite(pulPin, LOW);
  delayMicroseconds(motorDelay);
}
```

Figure 20: Stepper motor code

```
void loop() {
  switch (state) {
    case 0:
      Serial.print("\nWaitHome");

      while (digitalRead(devButPin) != HIGH) {}
      state = 1;  //movefeed
      break;
    case 1:  //movefeed
      Serial.print("\nMoveFeed");

      canLocation = 0;
      for (i = 0; i < numCans; i++) {
        if (usedCans[i] == 0) {
          canLocation = i + 1;
          usedCans[i] = 1;

          break;
        }
      }
      if (usedCans[numCans - 1] == 1) {
        for (i = 0; i < numCans; i++) {
          usedCans[i] = 0;
        }
      }
      if (canLocation > (numCans / 2)) {
        digitalWrite(dirPin, LOW);
        homeDir = HIGH;
        canDistance = canLocation - (numCans / 2);
      } else {
        digitalWrite(dirPin, HIGH);
        homeDir = LOW;
        canDistance = canLocation;
      }

      if (HALL == 0) {
        pulseMove = canDistance * ((pulsePerRev * gearRatio) / numSlots);
        Serial.println(pulseMove);
        for (i = 0; i < pulseMove; i++) {
          stepMotor();
        }
      }
```

Figure 21: Loop for state machine code, first half up to move for feed without hall sensor.

```
    else {
      countDelay = 0;

      while (pos != canLocation) {
        stepMotor();
        countDelay = countDelay + 1;
        prevEncode[0] = currEncode[0];
        prevEncode[1] = currEncode[1];
        currEncode[0] = digitalRead(hallPin[0]);
        currEncode[1] = digitalRead(hallPin[1]);

        if ((prevEncode[0] == 1) && (prevEncode[0] == 1) && (countDelay >= 200)) {
          if (currEncode[0] == 0) {
            pos = pos + 1;

            if (pos > numCans) pos = 0;
            Serial.println(pos);
            countDelay = 0;
          }
          else if (currEncode[0] == 0) {
            pos = pos - 1;
            if (pos < 0) pos = numCans;
            Serial.println(pos);
            countDelay = 0;
          }
        }
      }
    }
    state = 2;  //waitfeed
    break;
  case 2:  //waitfeed
    Serial.print("\nWaitFeed");
    while (digitalRead(devButPin) != HIGH) {}
    state = 3;  //movehome
    break;
  case 3:  //movehome
    Serial.print("\nMoveHome");

    digitalWrite(dirPin, homeDir);

    while (digitalRead(hallPin[2]) != LOW) {
      stepMotor();
    }
    state = 0;  //waithome
    break;
  default:
    Serial.print("\nbad");
    state = 0;  //waithome
    break;
  }
  delay(1000);
}
```

Figure 22: Loop for state machine code, second half from move for feed with hall sensor.

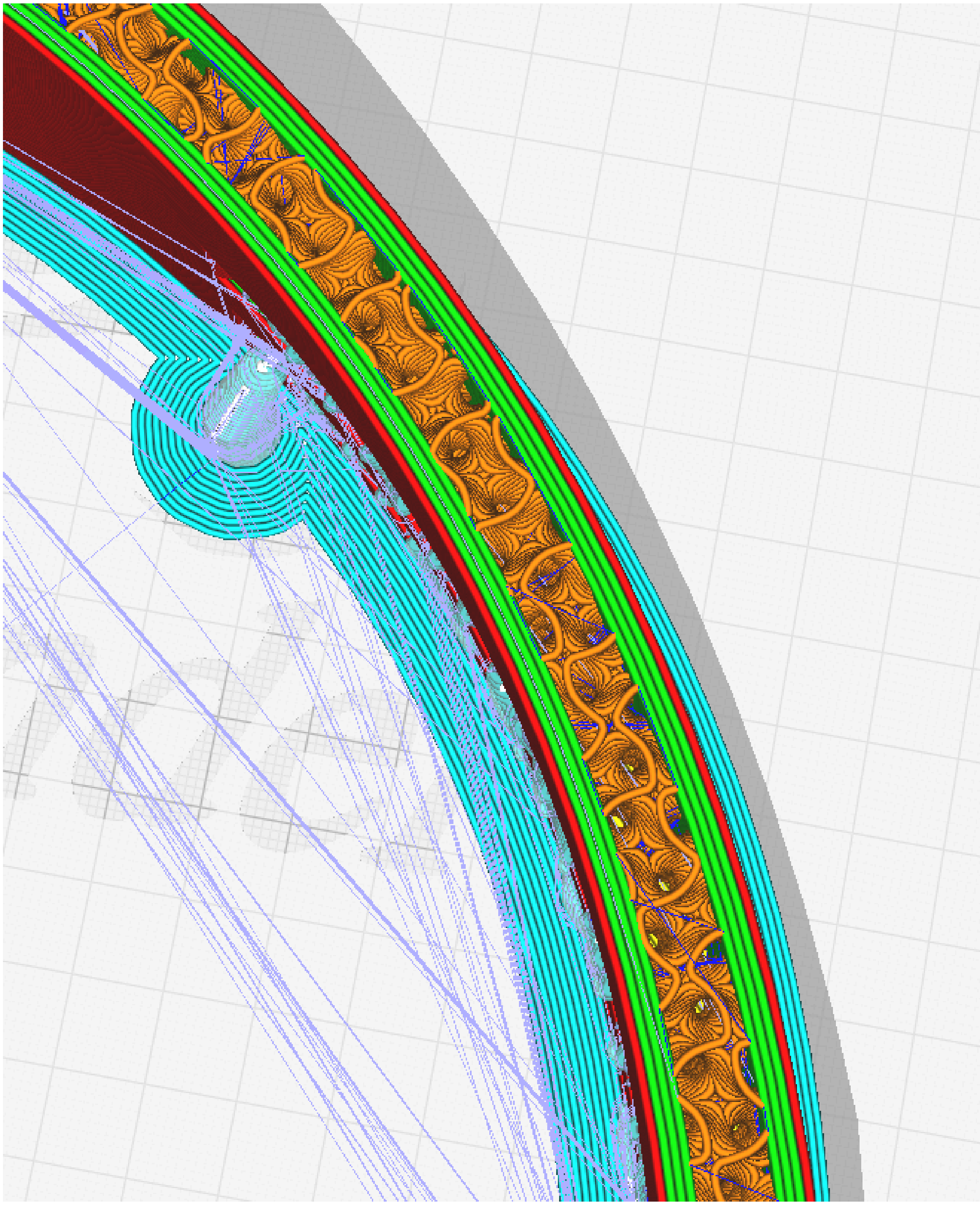
Conclusion

The automatic wet food cat feeder project was able to *mostly* address the challenges of preserving wet cat food while automating its dispensing process. By integrating a cooling system using Peltier devices, aluminum heat sinks, and a fan-assisted airflow design, the feeder was able to effectively maintain a colder than ambient environment. However, due to time limitations, the long term ability of the device to keep cat food at a safe temperature was not able to be tested. More validation is required to ensure the Peltier thermoelectric cooler devices are working as intended. The rotating dispenser mechanism, driven by a chain-connected motor and controlled via a button interface, ensured precise and user-friendly food dispensing.

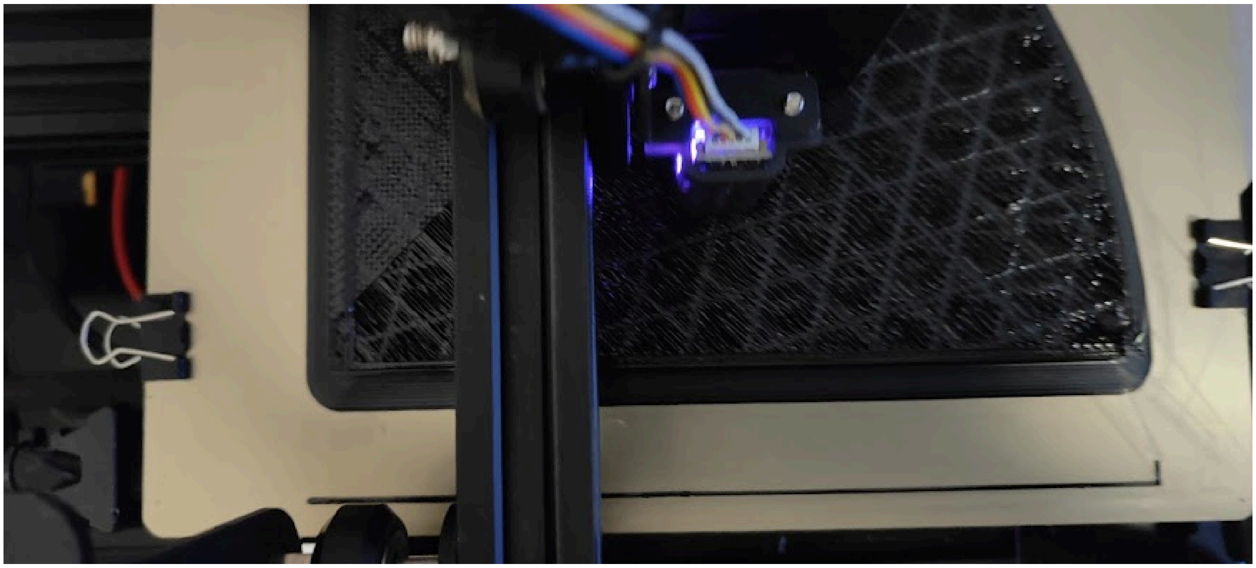
The design process encompassed multiple engineering disciplines - CAD modeling, heat transfer analysis, electronics integration, and the use of manufacturing techniques. Despite encountering challenges with precise component alignment, power supply limitations, and Peltier device performance, a functional prototype was able to be built within the time constraints given.

In its final state, the feeder prototype did meet the majority of the set primary objectives: it reliably cooled and dispensed wet cat food, providing a practical solution for pet owners seeking convenience and efficiency - though more testing is required to ascertain the extent of the cooling capability of the device. This project not only showcased the team’s technical expertise but also their ability to collaborate and overcome real-world engineering challenges. Moving forward, this design can be refined and scaled into a market-ready product, offering a convenient and contemporary solution to pet care.

Appendix



Appendix A: Gyroid infill



Appendix B: Tri-Hexagon infill

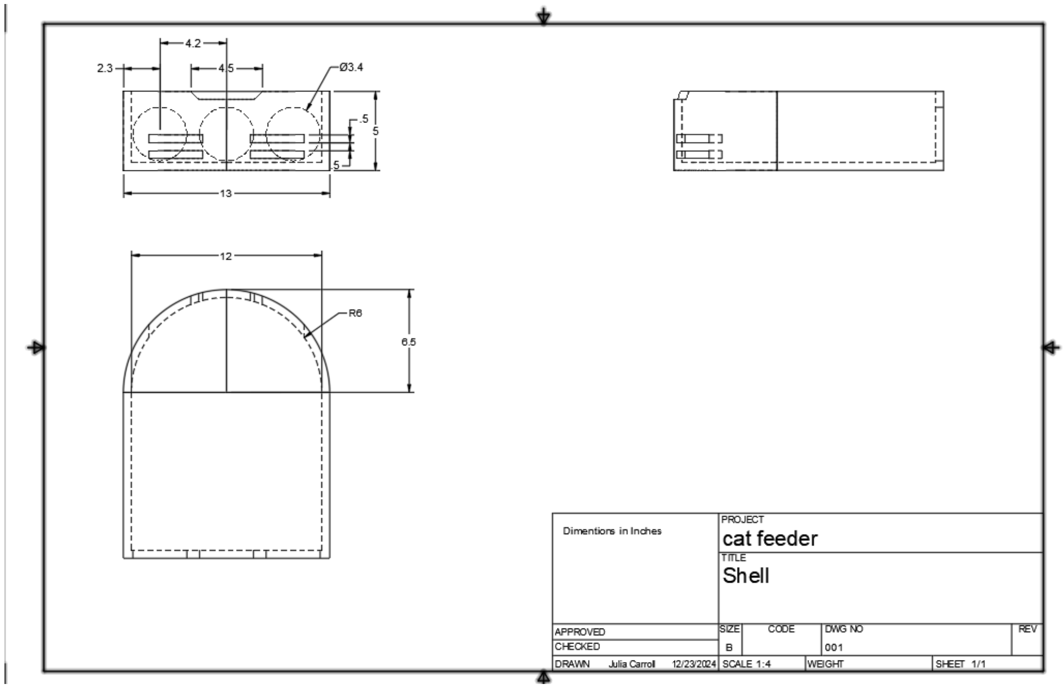
Appendix C: Assembled 3D printed baseplates

Appendix D: Outcome of gyroid infill walls

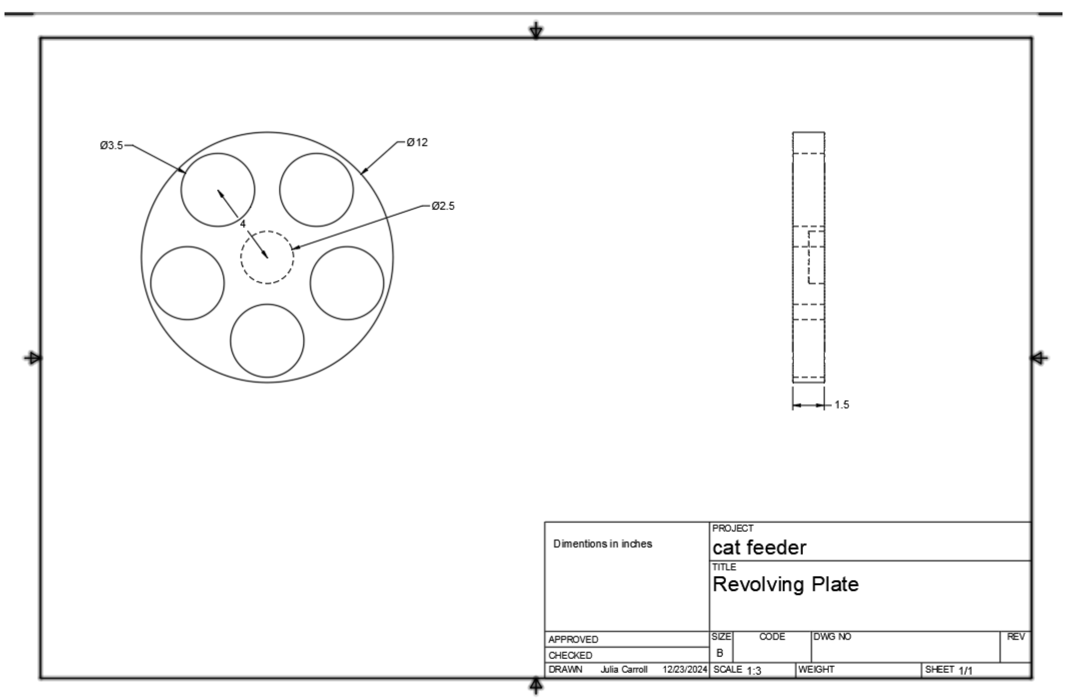
Appendix E: Slot used for motor placement and chain tensioning

Appendix F: Rear of housing with mounted flowthrough fans

Appendix G: Cat feeder prototype lid with foam insulation



Appendix H1 : Dimensional drawing of outer shell components



Appendix H2 : Dimensional drawing of revolving plate

Appendix I1: Motor Calculations

Appendix I2: Motor Calculations

Required Speed

V_m $(\theta / 360) \times (60 / (t_0 - t_1)) \times (D_{p2} / D_{p1})$

$(\ 72\ / 360)\ ) \times (60 / (\ 10\ -\ 2.5\ )) \times (\ 1.123\ /\ 0.731\ )$ $= 2.458$ [r/min]

Required Torque

$T = (T_a + T_L)$ (Safety Factor)

$= (\ 8.2643e\text{-}3\ +\ 1.113\ ) \times\ 2$ $= 2.243$ [lb·in]

$= 35.88$ [oz·in]

RMS Torque

$T_{rms} = \sqrt{\{((T_a + T_L)^2 \times t_1) + (T_L^2 \times (t_0 - 2 \times t_1)) + ((T_a - T_L)^2 \times t_1)\} / (t_0 + t_s)} \times$ (Safety Factor)

$= \sqrt{\{((\ 8.2643e\text{-}3\ +\ 1.113\ )^2 \times\ 2.5\ ) + (\ 1.113\ ^2 \times (\ 10\ - 2 \times\ 2.5\ )) + ((\ 8.2643e\text{-}3\ -\ 1.113\ )^2 \times\ 2.5\ )\} / (\ 10\ +\ 0.25\ )} \times\ 2$ $= 2.199$ [lb·in]

$= 35.18$ [oz·in]

Acceleration Torque

$T_a = ((1.2 \times J_L) / 386) \times (V_m / (9.55 \times t_1)) \times (1 / 16)$

$= ((1.2 \times\ 413.1\ / 386) \times (\ 2.458\ / (9.55 \times\ 2.5\ )) \times (1 / 16)$ $= 8.2643e\text{-}3$ [lb·in]

$= 0.1322$ [oz·in]

Load Torque

$W_T = W$

$= 2$ $= 2$ [lb]

$W_1 = w_1 \times n$

$= 0.355 \times 4$ $= 1.420$ [lb]

$T_L = (W_T + W_1)\,\mu\,(1 / (\eta \times 0.01))\,(D_{p1} / D_{p2})$

$= (\ 2\ +\ 1.420\ ) \times\ 0.5\ \times\ 0.5\ \times (1 / (\ 50\ \times 0.01)) \times (\ 0.731\ /\ 1.123\ )$ $= 1.113$ [lb·in]

Appendix I3: Motor Calculations

$= 17.81$ [oz·in]

Required Stopping Accuracy

$\Delta\theta = \Delta\theta \, (D_{p2} / D_{p1})$

$= 2 \times (1.123 / 0.731) = 3.073$ [deg]

Other requirement(s)

Holding Torque

Appendix I4: Motor Calculations