SMART MEAT DEFROSTER
Final Report

By

Ben Civjan

Brad Palagi

Payton Thompson

Final Report for ECE 445, Senior Design, Spring 2023

TA: Prannoy Kathiresan

29 March 2023

Project No. 49

Abstract

This report outlines the design and implementation of our Smart Meat Defroster. It provides an overview of the individual subsystems, specific design considerations taken, the requirements needed for the subsystems to be successful, and how these requirements were verified. It outlines the costs incurred by the project, uncertainties and challenges that our team encountered, and future plans and goals that we have for this project to make it better and more feasible for production.

Contents

1. Introduction

The task of defrosting frozen meat has had a long span of lacking innovation. Our solution is a meat defrosting device that warms meat within a container and tracks the internal temperature of the meat which will give user feedback once complete. Our product shortens defrost time, is hands-off, provides an even thaw, and is easily washable.

In the following chapters, we will discuss in finer detail the problem we look to solve and our solution. Including the different subsystems of our device, self-requirements we made for these subsystems, and how we tested them, as well as a rough estimate on the financial expenditure we would need to make this product into a business.

1.1 Problem

Defrosting frozen meat is a very tedious process. There are a few commonly used thawing techniques, each with its own issues. First, one can leave their frozen meat in the fridge to thaw. Although this takes around two days so if one is not prepared in advance their food will remain frozen. Next, you could try heating frozen meat in the microwave. This results in extremely uneven thawing where some thin parts of the meat begin cooking while thicker parts remain frozen. Simply leaving meat out on the counter is quite slow but also unsafe and can allow for potentially rapid bacterial growth. Additionally, one can run water over the meat to help defrost, but it is a hands-on, tedious process of changing the water periodically and still takes extensive time to fully defrost. Lastly, one can leave the meat on a defroster plate [1,2], but the length of time varies with the quality of the plate and still takes a significant amount of time and interaction. Ultimately, there are no solutions to evenly thaw frozen meat when trying to make a meal without planning to defrost well in advance.

1.2 Solution & Functionality

We propose a meat-defrosting container that will defrost a frozen chicken breast both quickly and evenly. Using an insulated container with a heating element to warm the air we will use heat conduction to thaw the meat efficiently without cooking it. This device will be extremely easy to use and provide a hands-off experience between placing the meat inside the container and removing it once it has thawed. Our insulated container will be food safe and use a heating element with a fan, and a conductive defrosting plate beneath the meat. This will allow the water frozen in the meat to melt and run off the sides of the plate. The top of the container will feature our electronics, where a PCB with a microcontroller will use temperature data from a thermal sensor to control the heating element. The meat's current temperature will be shown on an LCD display on the container, and a buzzer will sound when the meat has completed defrosting to alert the user. This will allow for an efficient and versatile hands-off method for people to defrost meat. Shown below in Figure 1 and Figure 2 are an illustrated plan, and the final photographs of our completed device. A successful project will be determined through our high-level requirements, namely:

- The device will power a heating element and display the current temperature until the frozen meat has reached an internal temperature of 2°C with an accuracy of +/- 2°C, then it will cut power from the heater and sound an alarm for the user.

- A frozen chicken breast will thaw faster in our defrosting container than standard techniques, such as placing the meat in water or on a defrosting plate. We expect to achieve at least a 20% reduction in defrost time.
- Defrosted meat will be evenly defrosted, with less than 10% of the meat being frozen or cooked.
- The device is easily washable and reusable without damaging the electronics or heating system.

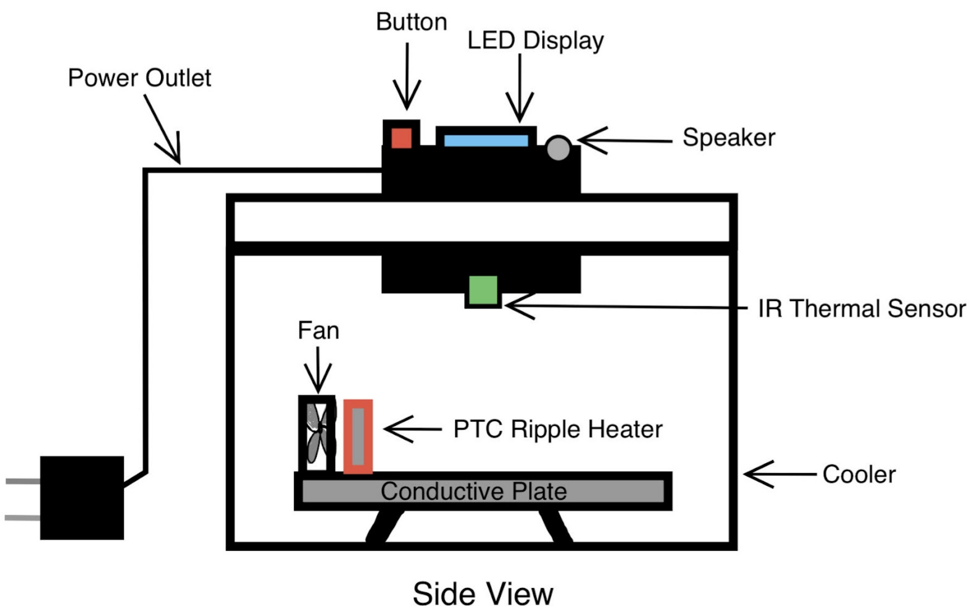

Figure 1: Drawn Diagram of the Meat Defrosting Device.

1.3 Visual Aid

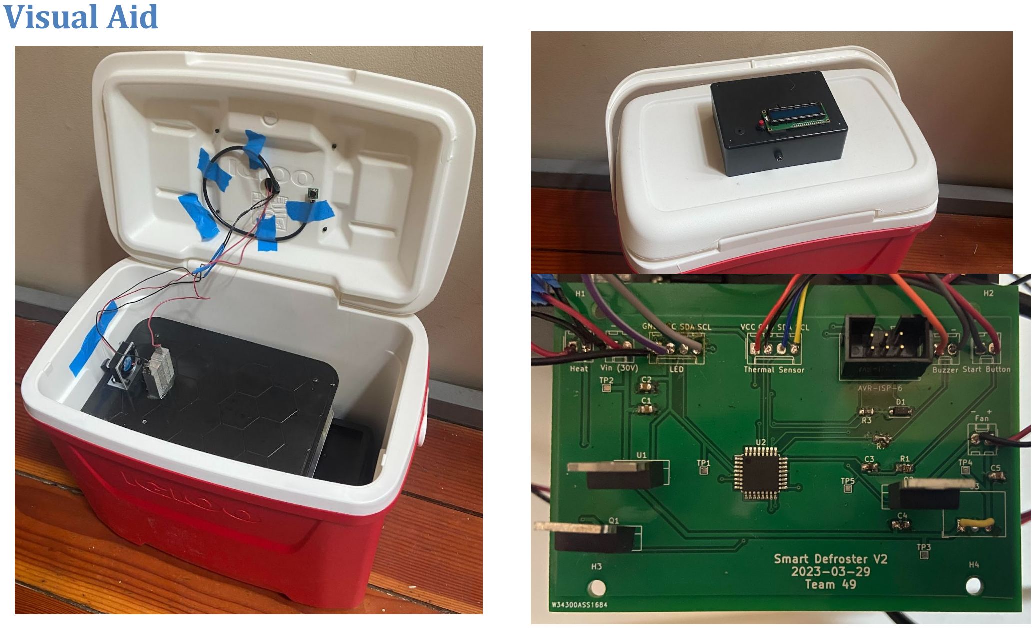

Figure 2: Photos of completed device, internal, external and PCB which is enclosed in the top mounted Polycase.

1.4 Subsystem Overview

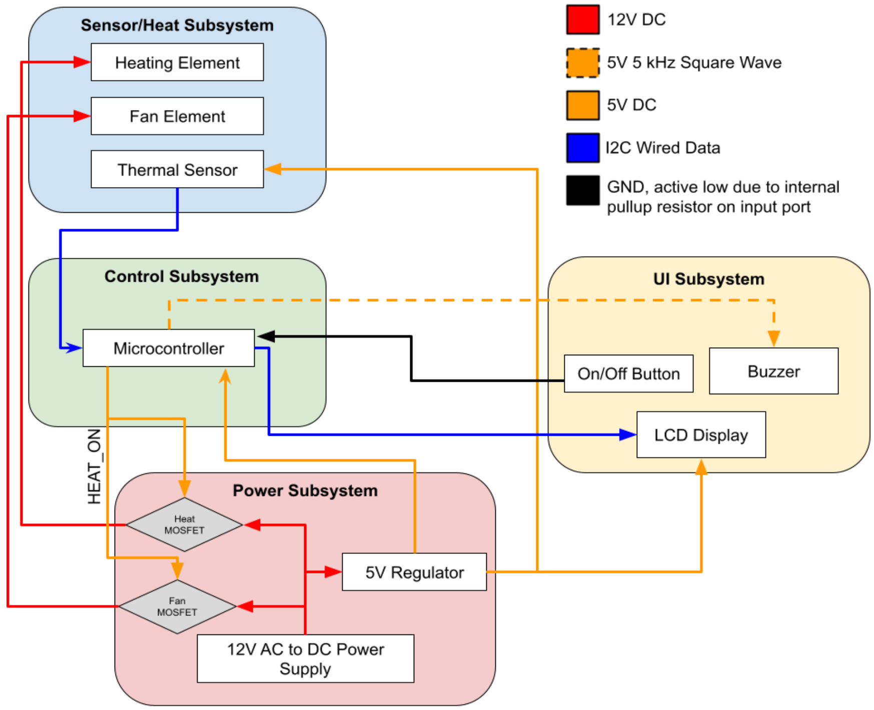

Figure 3: Block Diagram illustrating our final design subsystems and their connections.

1.4.1 Control Subsystem

The control subsystem featured in our block diagram, Figure 3, constraints our microcontroller which is programmed to receive input from the button and thermal sensor, and provide output to control the heat and UI subsystems. The microcontroller will send the current internal temperature estimation to the LCD display, and will also enable and disable the heating element depending on the current button position as well as defrosting completion. Finally, upon successful defrosting completion a buzzer will be activated to alert the user.

1.4.2 Sensor Subsystem

The sensor subsystem is used to detect whether the meat has been defrosted. It consists of a single infrared thermal sensor that can read the temperature of the meat without contact. The sensor is placed on the top of the defroster, away from the heating element. It is located a distance of 9 cm away from the internal defrost plate to ensure that the entirety of the detection circle is focused on the meat itself. It is connected via an Inter-Integrated Circuit (I2C) data bus to the ATMEGA328PB microcontroller. The readings from the sensor are sent via this bus. The microcontroller uses this data to determine which control signals to send to the other subsystems.

1.4.3 Heat Subsystem

The heat subsystem is made up of the PTC ripple heater, fan, and defrost tray. Once the button is pressed, the heater and fan are turned on by sending a signal to their individual MOSFETs from the microcontroller. When defrost is complete the microcontroller sends a low signal to the

MOSFETs to turn off the two devices. This creates heating through air convection. The meat product sits on the defrost plate causing it to defrost through conduction.

1.4.4 UI Subsystem

The main piece of our UI subsystem is a display that shows the current status of the device. The status shows whether the device is off, defrosting is completed, or, while the device is active, displays the estimated internal temperature. This display works well with our control subsystem and power subsystem as it operates on 5V and requires I2C data. We also have a simple button to activate our device, and a piezoelectric buzzer to sound an audible alarm to alert the user defrosting has been completed.

1.4.5 Power Subsystem

The power subsystem is used to provide the appropriate voltage to each of the other subsystems. It outputs 5 VDC to the microcontroller, LCD display, and thermal sensor. It outputs 12 VDC to the heating element and fan. In order for the heating element to function properly, it needs to be powered by 72 W. Therefore, the power supply must support a load of 6A of current.

2 Design

2.1 Design Alternatives

The first design consideration we had was what type of temperature sensor to use, thermal IR sensor or an internal meat probe. There were several problems with using an internal probe. Firstly, with a fully frozen piece of meat it is difficult to insert a probe due to the extremely solid form. This would also make it a more hands on defrosting experience which we wanted to avoid. Furthermore, for temperature probes that had a corkscrew tip made for frozen meats the cost was upwards of $200 as these are typically only used in industrial applications. Finally, it would be difficult to reverse engineer a full handheld probe in order to retrieve temperature data from it. The thermal IR sensor was better in nearly every aspect, it was non-contact and therefore did not need user interaction, inexpensive and was easily integrated with our PCB due to the I2C interface. The only shortcoming was the thermal IR sensor relied on the external temperature of the meat rather than internal. For this reason we needed to run trials to effectively calibrate an estimation of the internal temperature based on the external temperature of the meat.

The other main design alternative was our heating subsystem. Initially we had a PTC plate heater before adjusting to a PTC ripple heater. The PTC plate heater did not disperse heat away from itself very effectively so while the heater itself would become extremely hot it did not heat the air inside of the container efficiently. The ripple heater proved to be much more effective with the fan blowing air through it and heating the container. Unfortunately, due to our late adjustment of the heating element type we were only able to mount the heater with epoxy directly onto the plate. This caused much of the heat to be transferred directly to the defrosting plate and into the chicken, different from our initial proposal to use the heating element strictly to heat the air.

2.2 Control Subsystem

Design Procedure:

The main feature we were interested in from our processor was simplicity and the ability to connect our I2C thermal sensor and I2C display. Beyond that, our microcontroller only needs to perform trivial calculations and input or output analog signals. We also wanted a MCU popular enough to have a development board available, so that we could learn to program our MCU concurrently with designing and soldering our PCB.

Design Details:

At the heart of this subsystem is the ATMEGA328PB microcontroller unit (MCU). We will operate the MCU at 5V to allow for its fastest operating speed, and because 5V is needed elsewhere in the device. A power button will be connected to a port that is active low as it is initialized with an internal pull-up resistor, and the other input to the microcontroller will be I2C data from the thermal sensor. We will also output I2C data to the LCD display, a heat subsystem signal to activate the heating element and fan, and a 5 kHz square wave to the piezoelectric buzzer. The ATMEGA328PB has two separate I2C buses available, so we will not be concerned with any communication conflicts for either I2C device. Also, the MCU has an internal oscillator so that we are able to program it without additional circuitry.

Figure 4 below shows the functional flow of the MCU program for the device. The main idea is to activate the device with the button, and once activated, enable the fan and heating element. The next stage is to wait for the meat to fully defrost, once the estimated internal temperature reaches 0°C it will disable the heating element and fan, display on the LCD the device has reached completion, and sound the buzzer.

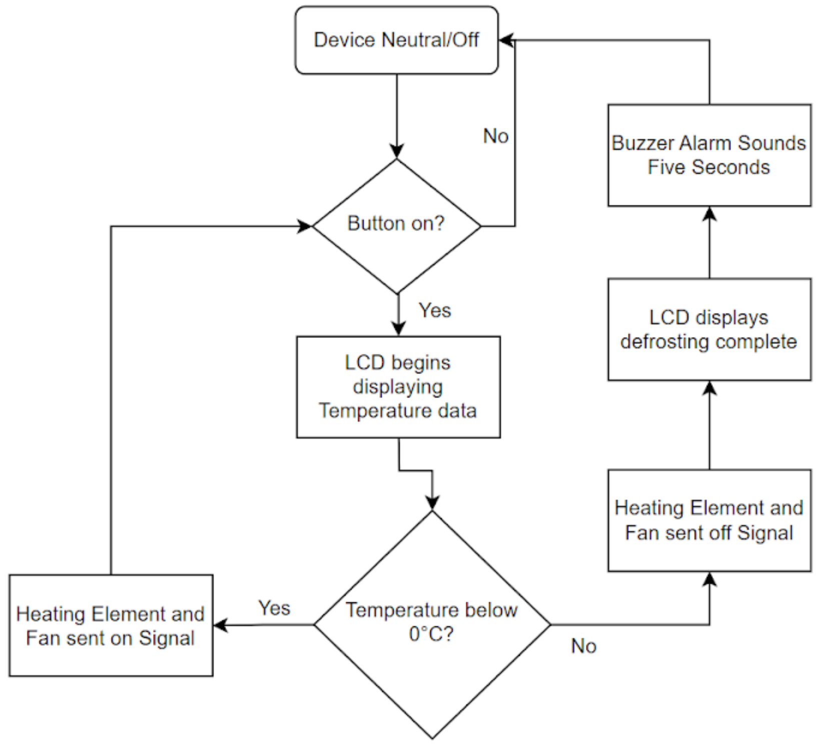

Figure 4: Functional Flow Chart of the Microcontroller Program

2.3 Sensor Subsystem

Design Procedure: Our main goal with the sensor subsystem was for it to be contactless. The infrared sensor operates without touching the meat and can record accurate readings far enough away to be attached to the top of the defroster. This non-contact feature prevents any potential damage to the sensor and ensures reliable long-term operation. It also makes it much more user-friendly since there is no need to try to puncture the meat at all. The sensor we chose is also very accurate for a wide range of temperatures, allowing us to predict the internal temperature of the frozen meat reliably and consistently.
Another design consideration we took with this subsystem is the interaction with the microcontroller. The infrared sensor seamlessly integrates with our microcontroller's I2C port. This made it easy for us to rapidly read the sensor and quickly output the data to the LCD display.

Design Details: When deciding where to place the sensor, we needed to take into consideration the impact of the heating element and the distance that the sensor was from the plate. In order to ensure that the thermal sensor readings were not impacted by the heat coming from the heating element, we elected to place the thermal sensor on top of the cooler. However, as shown in Figure 5, the detection radius is based on the distance from the thermal sensor. To account for this, we had to lift up the heating place to achieve a detection radius of 4.7 cm. This ensured that we were only measuring the meat and not also the plate.

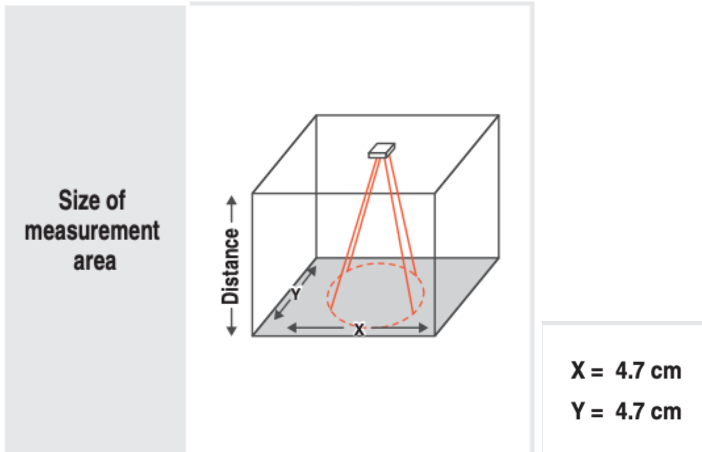

Figure 5: Diagram of the thermal sensor detection radius based on distance

Additionally, since we had to predict the internal temperature of the meat based on our thermal sensor reading of the external temperature, we had to conduct a series of calibration tests. During these tests we took the actual internal temperature of the meat and mapped it to the reading from the thermal sensor. We then created a regression fit between these two and used it to predict the internal temperature.

2.4 Heat Subsystem

Design Procedure: Our plan for the project was to have the majority of the heating be done through convection with the conduction from the plate being an additional heat mechanic. We chose convection due to it being an overall faster defrost procedure and ensuring even heat for the entire chicken breast. Since we used a cooler as our container, we decided to mount the fan and heater to the tray to avoid the interior walls burning. At first, we used a PTC heat plate that was 30V and 1A for 30W of power. However, the plate was poor at radiating heat, so we altered the design to use a ripple heater that we saw is commonly used with a fan for heating [13]. We increased the wattage of the ripple heater to 70W based on the equation below (2.4.1). Throughout the process, we had issues powering the heater. We discovered that when the heater is first turned on, the resistance is very low and increases as it heats up (Figure 6). This activated our original power supply to begin overcurrent protection causing no current to be sent to the system. To combat this we increased our power supply to provide 6A of current.

Design Details: Our goal for the heating subsystem was to reach 32.22°C (90°F) within 15 min and stay at that temperature throughout the rest of the defrosting process. Unfortunately, we were unable to reach this goal. Without anything in the cooler, the temperature would reach around 29.4°C (85°F), and when chicken was added only reached 26.1°C (79°F), shown in Figure 7. This occurred due to the heater being mounted to the conductive tray. The tray would absorb some of the heat causing it to be colder than expected. It also increased the conduction heat process making it the dominant heating method. Also, with the inclusion of the water introduced by the frozen chicken, it cooled the air.

2.1.4

Convection heating equation:

$$Q = V \times \rho \times c \times \Delta T$$

$$7579.86\,J = 0.2755 \times 1.204 \times 1870 \times 12.22$$

$$120\ sec\ to\ heat\ up: \frac{7579.86J}{120sec} \approx 70W$$

Heat energy required (Q), J

Volume space (V), m^3: $0.2755m^3$

$0.2286m \times 0.38735m \times 0.31115m$

Temperature difference (ΔT), °C: $12.22\,°C$

$32.22\,°C - 20.00\,°C$

Density of air (ρ), $\frac{kg}{m^3}$: $1.204\,\frac{kg}{m^3}$

Specific heat capacity of air (c), $\frac{J \times kg}{°C}$: $1870\,\frac{J \times kg}{°C}$

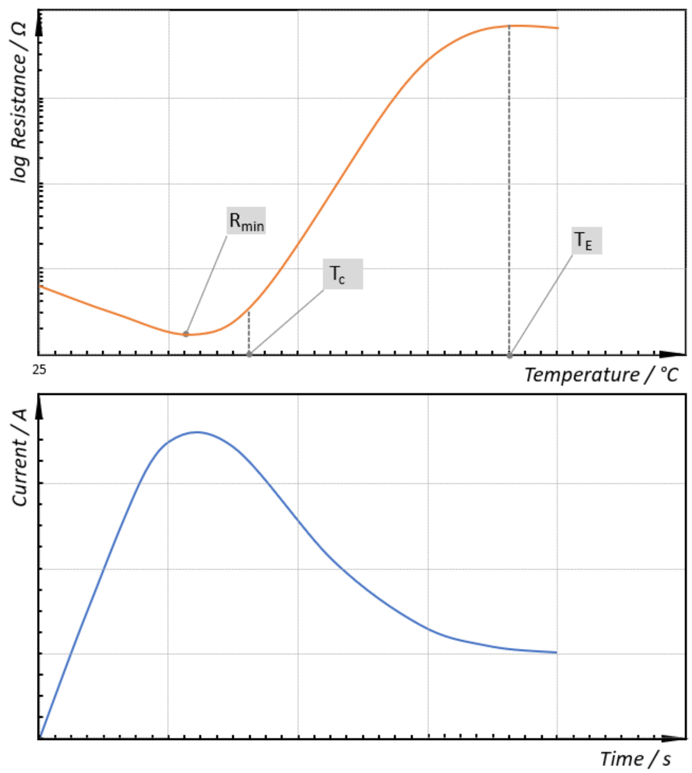

Figure 6: Diagram of how the resistance and current change over time within the heater

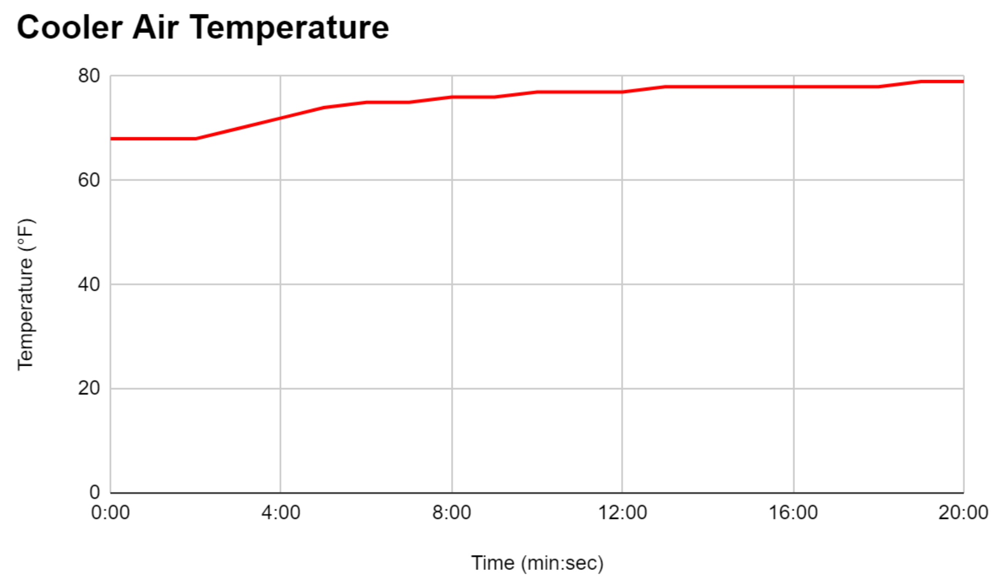

Figure 7: Graph of the temperature within the cooler throughout the chicken defrost process

2.5 UI Subsystem

Design Procedure:
For our User Interface subsystem we intended to provide the user with clear information regarding the status of the device, and easy operation without any potentially unnecessary controls. For these reasons we decided on an LCD display, audible buzzer, and basic push button.

Design Details:
To begin the defrosting process a simple push button that was available in the ECE445 lab is pressed. We connected the button from an input port on the MCU to ground. The port the button was input to was initialized using the ATMEGA328PB's internal pullup resistor functionality. This allowed for easy programming of the device where that port was active low, when the button was pressed it shorted the port to ground. Central to the user interface is the SunFounder I2C LCD1602 display. There are three main status stages which are displayed before, during and after defrosting. When the device is first plugged in the display starts in a neutral or off stage, shown in Figure 8.a signaling that the device has not yet been activated and the heating element and fan are disabled. Once the button has been pressed, the device is actively defrosting the meat and the display changes to constantly update the estimated internal temperature of the meat, seen in Figure 8.b. The thermal sensor reads a temperature value ten times per second. So, to give a clearer more legible display we average every ten temperature points and then update the display once per second. Finally, once the device has detected defrost completion the display reads "Defrosting Complete" as shown in Figure 8.c.

(a) Neutral Stage (b) Active Stage (c) Completion Stage

Figure 8: Each Status Stage of the LCD Display.

The final piece to our User Interface subsystem was a PS1420P02CT piezoelectric buzzer which sounded upon defrost completion. We programmed the buzzer by rapidly turning on and off the 5V output port connected to it. With a 200 microsecond period we effectively produced a 5 kHz Square Wave, which produced the highest decibel level available on this buzzer shown in Figure 9.

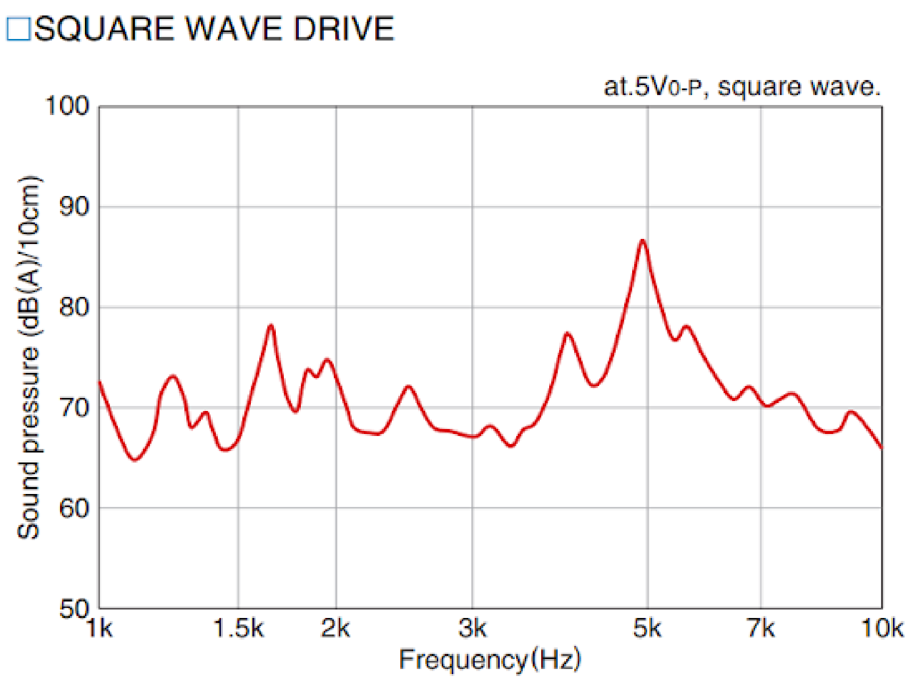

Figure 9: PS1420P02CT Buzzer Frequency Sound Pressure Characteristics.

2.6 Power Subsystem

Design Procedure: Initially in the power subsystem we were using a 30V AC/DC power supply with a current rating of 1A. The reason we chose this was that our first heating element was rated at 30V, 30W. We anticipated that this would draw 1A of current based on the P = IV power equation. However, we didn't fully understand beforehand exactly how the heater drew current. Since it's a PTC heater, its resistance increases significantly when its temperature increases.

When we first began testing, we connected our power supply to the heater and noticed that the power supply would shut off. After some debugging, we discovered this to be a result of the overcurrent protection feature in the power supply, since when cold the heater would draw about 3A. Our initial fix to this issue was to connect the heating element to the output of our 12V synchronous buck converter that we used to power the fan. To overcome this we had to pivot to using a different power supply 12V, 6A. Since our highest voltage level is now 12V, there was no need for us to use the synchronous buck converter. We were able to remove this component from the power system design.

Design Details:

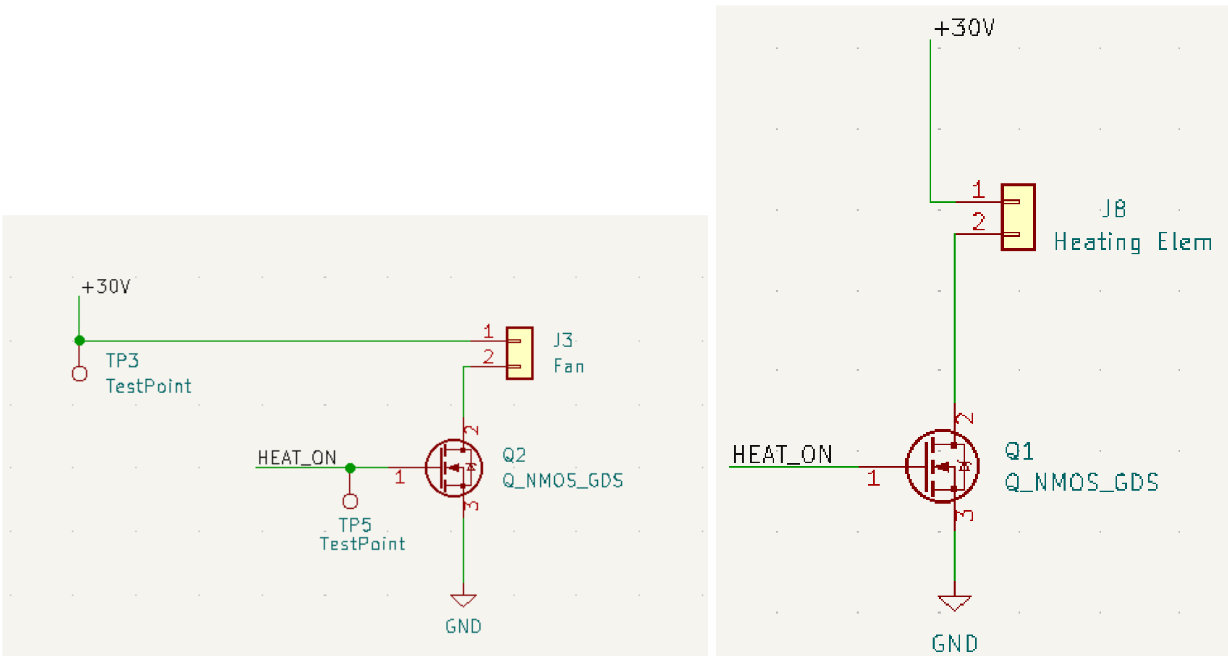

Figure 10: Dual MOSFET wiring from heat subsystem to ground

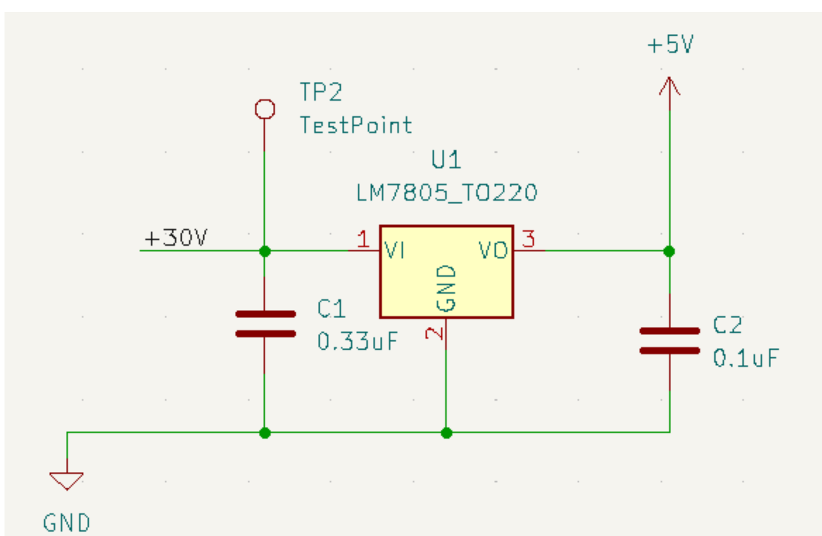

Figure 11: 5V Linear Voltage Regulator

The main aspects of our power subsystem design were the MOSFETs connected to each element of the heat subsystem and the 5V linear voltage regulator. As shown in Figure 10, we decided to use MOSFETs to act as switches because when powered with a threshold voltage, they turn on and allow current to flow through them. When this signal is off, they act as an open circuit. This made them an ideal candidate for our circuit because we could power them with the *HEAT_ON* signal to allow current to flow through the heater and fan.

Since our heating element requires a large amount of current, we chose to use the IRL40B215 MOSFET. This is rated at 40V, 120A, so it's easily large enough for our purposes. It also is able to conduct about 100A of current with an input voltage of 4.7V, which is what our microcontroller provides. This ensures that our heater is not negatively impacted by the quality of the MOSFET.

The LM7805 linear voltage regulator shown in Figure 11 ensures that our 5V power remains consistent, between 4.5V and 5.5V. The thermal sensor functions at a current of 3.5 mA, the microcontroller operates at 0.5 mA, and the LCD screen operates at 7mA. As a result, the LM7805 will dissipate power according to the formula:

$$Power\ Dissipation\ =\ (Vin\ -\ Vout)\ *\ Iout$$

$$(30\,VDC\ -\ 5VDC)\ *\ (3.5mA\ +\ 0.5mA\ +\ 7mA)\ =\ 0.275W \qquad (2.2)$$

Through our calculations in equation *2.2*, we can see that the linear regulator will not be dissipating enough power to overheat.

3. Design Verification

3.1 Control and User Interface Subsystem

Our first requirement was the ability to display temperature data on the LCD screen which had to be powered with 5V. Using the MCU we were able to successfully receive data from the thermal sensor, calculate an estimated internal temperature based off of our calibration and display it as seen in Figure 8.b. A multimeter was used to verify the display received the expected 5V. Our other requirement for the MCU was the ability to properly complete defrosting across the other subsystems. This was another successful requirement; as demonstrated during our final demo the MCU output that toggles the fan and heating element MOSFET's is disabled upon completion, along with the display showing completion seen in Figure 8.c, and output a 5 kHz square wave which produced an easily audible buzzing heard in the final demo.

3.2 Sensor Subsystem

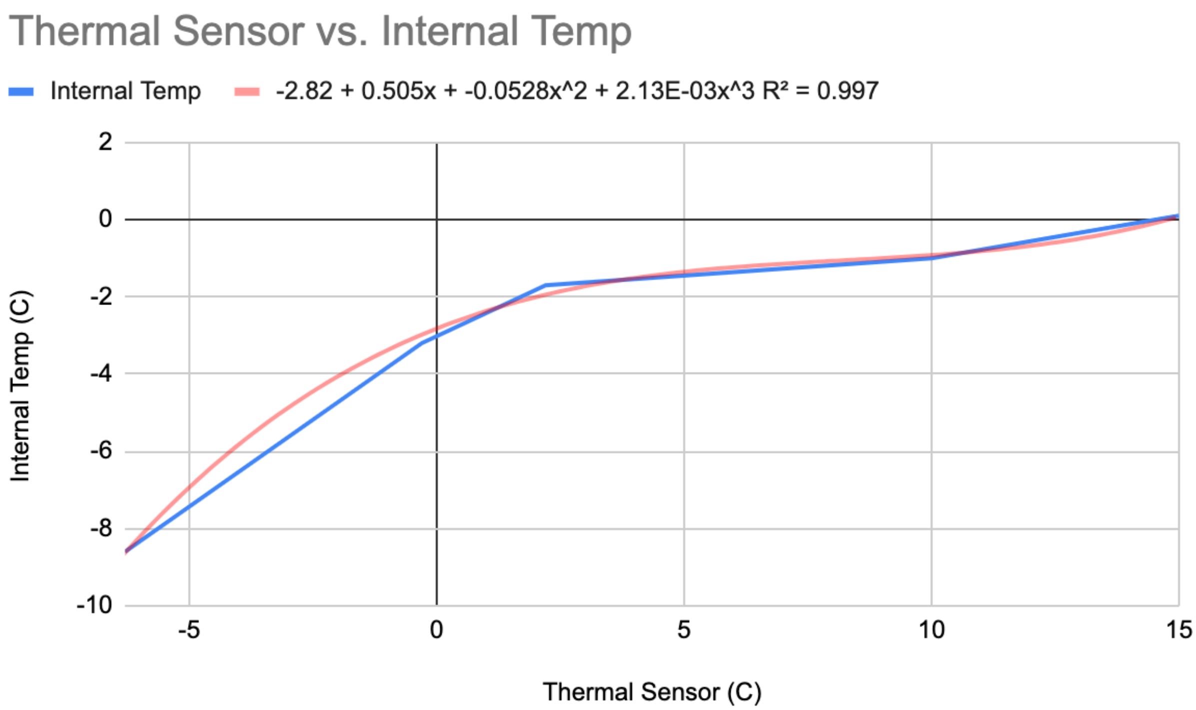

Figure 12: Thermal Sensor to Internal Temperature Poly Regression Line

In the sensor subsystem, we had two main requirements to meet. Our first requirement was to read the surface temperature of the meat within a range of ±2°C. Since our thermal sensor is rated to have a 1°C range of error we were able to achieve this simply by making sure our thermal sensor was placed at the correct distance.

Our second requirement was to predict the internal temperature of the meat with an accuracy of ±2°C. We achieved this by creating a polynomial regression line, as seen in Figure 12, mapping the thermal sensor reading to the internal temperature of the meat. It can be seen from Figure 12 that the regression line never deviates from the actual temperature of the meat more than 2°C.

3.3 Power Subsystem

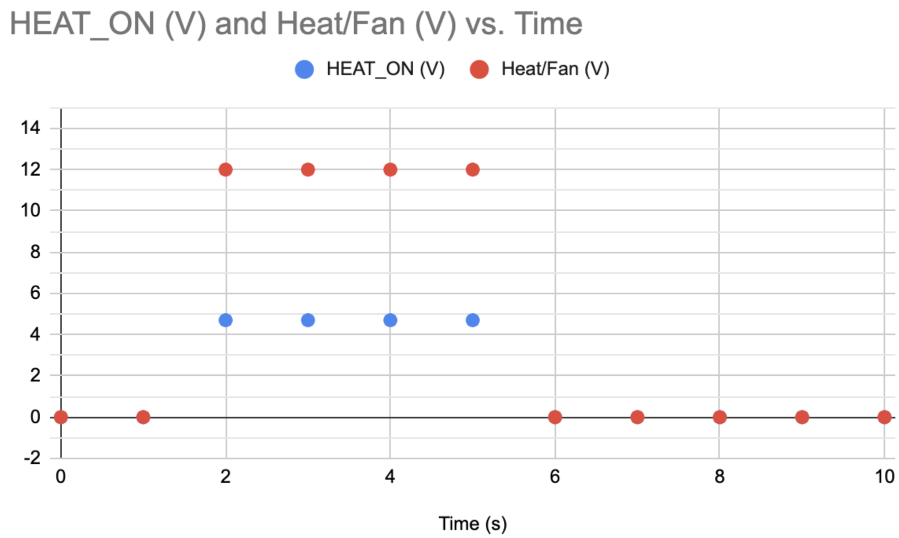

Figure 13: Data Plot of HEAT_ON Signal and Heat/Fan voltage over time

Our first two requirements for the power subsystem was to maintain an output voltage between 4.5 and 5.5V for the microcontroller, thermal sensor, and LCD display, and to provide an output voltage of 12V for the fan and heater. After changing our power supply, the 12V came directly from the AC/DC converter. We verified that the output voltage from the regulator was 5V and never fluctuated more than 0.5V by probing the circuit with a voltmeter and consulting the error on the LM7805 chip, which was 0.2V.

Our final requirement was to stop supplying power to the heating subsystem when the *HEAT_ON* signal was switched off. The results of our verification test for this are shown in Figure 13. When the *HEAT_ON* signal is 4.7V, the heater and fan both receive the expected 12V.

4. Costs

4.1 Parts

Table 1: Labor and Parts Cost

Labor			**Cost**	**Count**	**Total Cost**
Computer Engineer			$15,195	3	$45,585
Parts					
Type	**Manufacturer**	**Number**			
Cooler	Walmart	----------	$21.88	1	$21.88
Temperature Sensor	Omron	D6T-1A-02	$7.09	1	$7.09
Beeper	TDK	PS1440P02BT	$0.75	1	$0.75
Button	TE Connectivity	1825910-2	$0.15	1	$0.15

LED Display	Adafruit	878	$9.95	1	$9.95
PTC Heater Ripple	hotzzz	------------	$11.29	1	$11.29
Microcontroller	Microchip	ATMEGA328PB-ANR	$1.87	1	$1.87
5V Voltage Regulator	Jameco	LM7805	$1.19	1	$1.19
Fan	Orion	OD4010-12HSS	$7.71	1	$7.71
12V, 6A Power Supply	COOLM	------------	$14.59	1	$14.59
Plastic Enclosure	Polycase	LP11F	$3.20	1	$3.20
MOSFET 40V, 120A	Infineon Technology	IRL40B215	$2.35	2	$4.70
Defrosting Tray	YUNDOOG	------------	$19.89	1	$19.89
				Parts	$104.24
				Total	$45,689

4.2 Labor

According to data from the 2020-2021 graduating class, the average starting salary of a UIUC graduate is $105,352 for computer engineers. One may work around 2080 hours per year (40-hour week * 52 weeks). We will be working on our project for 12 weeks and expect around 10 hours of work per week since it is a 4-credit hour course. Also, we apply a 2.5 overhead multiplier to account for unexpected costs. Labor cost per partner would total $15,195 as shown below and inputted into Table 1.

$$\frac{\$105{,}352}{1\ year} * \frac{1\ year}{2080\ hours} * \frac{10\ hours}{1\ week} * 12\ weeks * 2.5 = \$15,195\ (per\ Computer\ Engineer) \quad (4.1)$$

4.3 Schedule

Week	Ben Civjan	Brad Palagi	Payton Thompson
2/27/23	Discuss voltage and power management of subsystems with Jason or Matthew (step downs/AND gates)	Research microcontroller, understand capabilities of ATmega, how we will program it	Discuss container dimensions, heat placement with Machine Shop
3/6/23	Order power system components including AC/DC converter and regulators	Begin ordering initial parts for control, UI and sensor subsystems	Order heating element, submit plans to machine shop with selected cooler and polycase
3/13/23	HAVE	SAFE	FUN!

3/20/23	- Breadboard Power Subsystem - Update PCB with new parts	- Get supplies to Machine Shop - Breadboard Control Subsystem	- Order final parts - Reevaluate capacitor/resistor values for PCB
3/27/23	- Breadboard Power Subsystem - Order new PCB	- Breadboard Control Subsystem - Breadboard UI Subsystem	- Get Final supplies to Machine Shop - Breadboard Heating Subsystem - Solder PCB 1
4/3/23	- Breadboard entire System	- Code microcontroller	- Breadboard Sensor Subsystem
4/10/23	- Solder PCB 2	- Code microcontroller - Connect PCB to cooler	- Breadboard entire System
4/17/23	- Final Implementation - Test Power Subsystem - Test UI Subsystem	- Final Implementation - Test Control Subsystem	- Final Implementation - Test Heating Subsystem - Test Sensor Subsystem
4/24/23	- Write code for new LCD screen	- Write code for new LCD screen	- Run calibration tests on sensor subsystem
5/1/23	Practice Presentation and Finish Final Paper	Practice Presentation and Finish Final Paper	Practice Presentation and Finish Final Paper

5. Conclusion

5.1 Accomplishments

We have found our project to be an overall success. We were able to accomplish all of our high-level requirements and even exceeded our expectations of the speed of the defrost process being cut in half. The chicken was evenly defrosted with a small variance in temperature when comparing the thin versus thick parts. The cooler was easily washable with the defrosting tray being removable from the container to make cleaning the bottom of the cooler simple. Also, our entire circuitry was on the PCB.

5.2 Uncertainties

In our testing environment, all our systems functioned as expected. However, our design is calibrated to be used with chicken that is around 220 g. In the real world frozen meats will come in all different shapes and sizes. If the chicken placed in our defroster was significantly larger than what we calibrated it with, then we likely would have inaccurate readings and not fully defrost the meat.

5.3 Ethical Considerations

Throughout the design of our product we considered the ethical guidelines provided by IEEE. One crucial aspect we prioritize is user safety. According to section 1 of the IEEE code of conduct, it is essential “to hold paramount the safety, health, and welfare of the public” [6]. Sections 1.2 and 1.3 of the ACM code of conduct state that a computing professional should “avoid harm” and “be honest and trustworthy” [5]. In the case of a device that will generate heat and be in contact with food, it is crucial for us to maintain full honesty in our design process. We understand the potential risks associated with electrical appliances, especially when handling frozen meat, and thus we ensure that our defroster incorporates robust safety features. We meticulously design the product with insulation, grounding, and protection mechanisms to prevent electrical shocks or fire hazards, ensuring the well-being of our users.

Furthermore, we are cognizant of the potential health risks associated with improper defrosting practices. It is our ethical duty to educate users about safe handling and defrosting methods. We ensure that the defroster is designed in a way that facilitates easy cleaning and maintenance, preventing the accumulation of harmful bacteria or contaminants that could jeopardize food safety.

By incorporating robust safety features, optimizing energy consumption, and promoting proper food handling practices, we aim to deliver a reliable, eco-friendly, and user-friendly product that meets the highest ethical standards.

5.4 Future work

We have identified two areas of concern: cosmetic and performance. For cosmetics, we would like to use a more compact container. This would allow for easy storage and keep the chicken close to the thermal sensor for accurate readings. Also, separate the heater and fan from the defrost tray. This would make it more efficient to clean with the defrost tray completely removable and protect the devices from being damaged by water.
The performance of the heater already exceeded the goal of a 20% decrease in time. However, we would want to further push that limit by adding a more advanced heating subsystem that focuses on the air convection heating method. This could be done by adding air ducts that would increase the circulation of air causing the hot air to move more rapidly. We would also like to expand our device to be able to defrost all shapes and sizes of a variety of meat. This could be accomplished by finding surface/internal temperature ratios for multiple types of meat and testing for how the size of meat affects this ratio as well. This way a user could select a meat they would like to defrost and include its weight for fast and hands-free defrosting.

References

[1] “Amazon.com: Evelots meat thawing tray for frozen meat.” [Online]. Available: https://www.amazon.com/Evelots-Rapid-Defrosting-Frozen-Naturally/dp/B01K3A5X26.

[2] “What Is a Defrosting Tray and Does It Really Work?” [Online]. Available: https://www.thespruceeats.com/what-is-a-defrosting-tray-4782382.

[3] “ATMEGA328PB.” [Online]. Available: https://www.microchip.com/en-us/product/atmega328pb

[4] “The Big Thaw — Safe Defrosting Methods | Food Safety Inspection Service.” [Online]. Available: https://www.fsis.usda.gov/food-safety/safe-food-handling-and-preparation/food-safety-basics/big-thaw-safe-defrosting-methods.

[5] “ACM Code of Ethics and Professional Conduct.” [Online]. Available: https://www.acm.org/code-of-ethics [Accessed: Feb.9, 2023].

[6] “IEEE Code of Ethics”, IEEE. [Online]. Available:https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed: Feb.9, 2023].

[7] S. Pelonis, “What you need to know about PTC heaters,” *Pelonis Technologies, INC.* [Online]. Available: https://www.pelonistechnologies.com/blog/what-you-need-to-know-about-ptc-heaters.

[8] “D6T Thermal Sensors Datasheet.” [Online]. Available: https://www.mouser.com/datasheet/2/307/en_D6T_catalog-1633759.pdf

[9] “How to Install PTC Heater in an Incubator.” [Online]. Available: https://www.youtube.com/watch?v=13modII5Zew.

[10] “Adafruit 0.56" 4-Digit 7-Segment Display with I2C Backpack.” [Online]. Available: https://www.adafruit.com/product/878.

[11] “PERFEIDY 30V 1A Universal AC Adapter.” [Online]. Available: https://www.amazon.com/PERFEIDY-Switching-Regulated-Transformer-Interchangeable/dp/B0B2RLKLMH/ref=sr_1_6?crid=SEQ750XDWWP&keywords=30v+power+supply&qid=1677196116&sprefix=30v+power%2Caps%2C158&sr=8-6.

[12] “LM7805CT 5V 1A Positive Voltage Regulator.” [Online]. Available: https://www.digipart.com/part/LM7805.

[13] “DIY 12v Ceramic Air Heater (w/fan!).” [Online]. Available: https://www.youtube.com/watch?v=GoLGtR9ElpA.

Appendix A

Requirement and Verification Table

Table 2: System Requirements and Verifications

Requirement	Verification
Container air temperature consistently at 37.78 ±2°C	Periodically check the temperature of a thermometer inside the container
Container air temperature takes less than 15min to reach 37.78°C	Measure the length of time it takes for a thermometer inside the container to reach 37.78°C
Pass temperature data from the microcontroller to the LED Display	- Press the on button for the device and ensure the LED display receives power - Ensure the user can view the meats current temperature on the seven segment display
Microcontroller disables the heating element once the meat temperature reaches 1.66°C	- Enable the heating element and wait for the thermal sensor to detect a temperature of 1.66°C Ensure the heating element is no longer receiving a voltage
Microcontroller sounds a defrost completion alarm	- Run the device until temperature reaches 1.66°C - Ensure a sound tone is heard for three seconds - As the meat continues to raise in temperature the sound should play again for every degree increase - If the meat reaches 4.44°C then ensure the sound remains constant until the button is pressed off
Calculate the internal temperature with an accuracy of ±2 C.	Run calibration trials with various meat. At intermittent intervals, take out the meat to perform a heat check with a thermometer. Compare this to what we expect the internal temperature to be based on our thermal reading and calculation.

Maintain an internal air temperature between -40 C and +80 C.	During internal temperature tests (as described above), take readings of the air temperature to ensure that it has not gone outside of the acceptable range.
Ensure voltage does not fluctuate more than the allowed 0.5 V.	Hook up an oscilloscope to the output of the Linear Voltage Regulator (input of the thermal sensor) during trial runs to track input voltage.
LED Display Shows Current Temperature	- Link the I2C output of the microcontroller to the quad seven segment display - Ensure the display shows the current meat temperature throughout the thawing process
Buzzer Plays Audible Tone at defrosting completion	- Verify that the buzzer can be controlled by the microcontroller - Ensure the buzzer sounds once the temperature has reached 35 degrees, and resounds for every degree increment until 40 degrees where the buzz stays constant
Maintain an output voltage between 4.5 and 5.5 V (for microcontroller and thermal sensor)	Connect the oscilloscope to the output of this element during trial runs and verify that the voltage stays consistent
Provide an output voltage of 12 V to the fan	Connect the oscilloscope to the output of this element during trial runs and verify that the voltage stays consistent
Stop supplying power to the heating plate once the *HEAT_ON* signal has been set to 0	Record the point in time that this signal is sent to the voltage switch circuit, and verify via oscilloscopes that the voltage of reaching the heater
All power controllable with On/Off button	Measure the power at all components when the device is plugged in but switched off. Verify that this is 0V. When the device is switched on, probe all components and check against the expected results.