MULTIPURPOSE TEMPERATURE CONTROLLED CHAMBER (FOR CONSUMER APPLICATIONS)

ECE 445 DESIGN DOCUMENT - Spring 2024

Project # 42

Stefan Sokolowski, Isaac Brorson, Mitchell Stermer

TA: Selva Subramanium

Table of contents

1. Introduction

1.1 Problem

Oftentimes, people store food in various locations such as the kitchen, refrigerator, or freezer with only a general idea of how the temperature of that item will change over time. In some cases, they'll come back hours later only to find that their food is not properly thawed or frozen as they were hoping. A few specific examples of this could include: putting a beverage in the freezer to quickly cooler it, only to forget about it and later find it exploded and frozen; planning to cook a steak, but forgetting to move it from the freezer to the refrigerator the previous day; or setting food out overnight in order to prepare it for the next day only to find that it didn't thaw as expected. After observing these problems and many others similar to them, we believe that there needs to be an intelligent device that could quickly cool or warm food without freezing or cooking it.

1.2 Solution

Our solution is a programmable temperature controlled chamber which allows a user to set the temperature curve of a food item they are planning on consuming in the near future. By setting a temperature curve, the user will be able to precisely control what temperature changes the food undergoes within a certain interval of time. In general, this device would be able to quickly heat or cool food to a desired temperature, then hold it at that temperature until the user is ready to use the food. Someone would use this device by placing their food item in the device's insulative chamber and closing the door. Next, the user interface would present a variety of options: standard heating or cooling presets for common food items, temperature set and hold, or the ability to set a detailed temperature curve. The user may then leave while the chamber performs its function and return at the desired time at which they set it to be ready. Temperature controlled chambers on the market are exorbitantly expensive and large for a household kitchen, and this design is better suited for consumer application.

1.3 Visual Aid

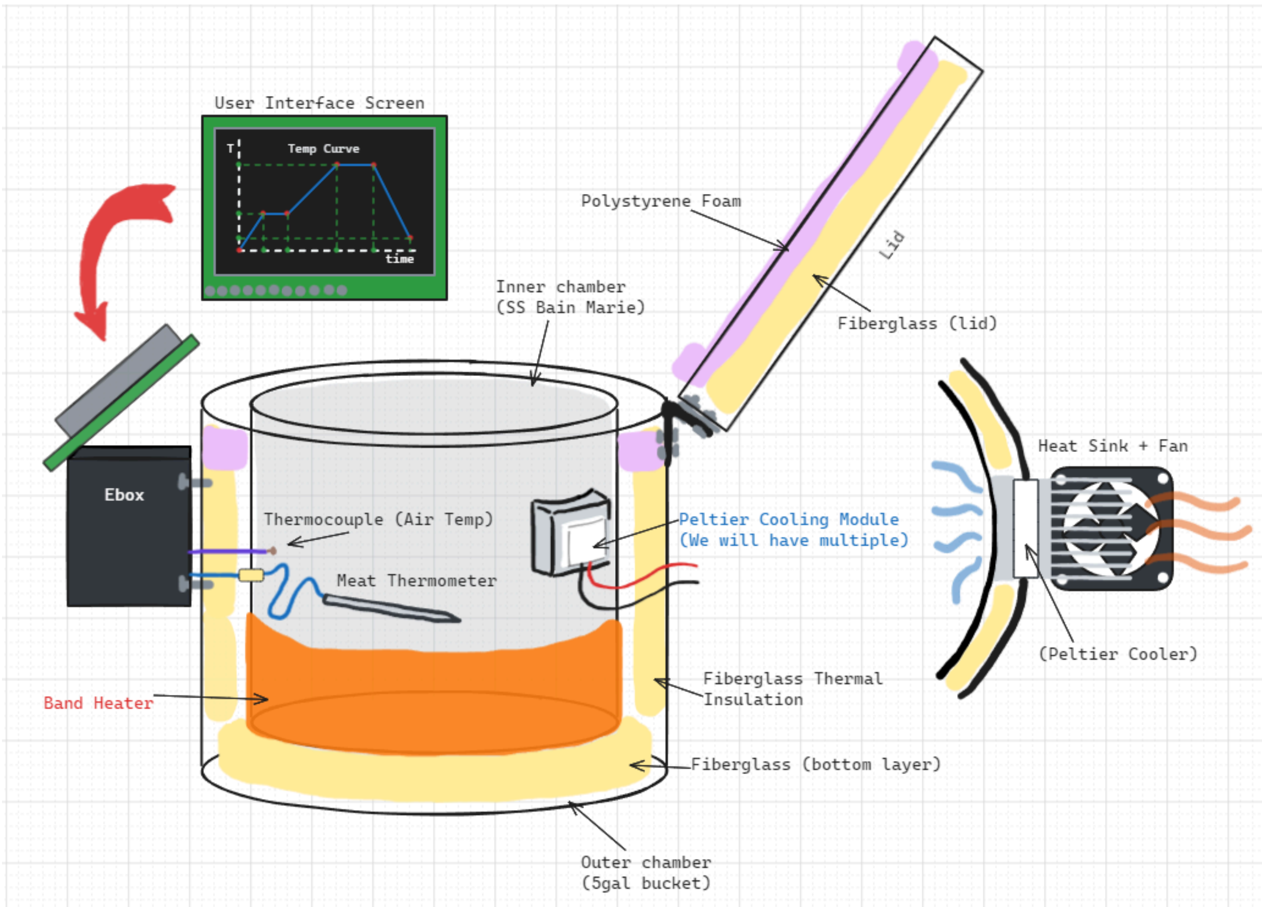


Figure 1: Temperature Controlled Chamber Visual Aid

1.4 High Level Requirements

To consider ourselves successful, we must be able to fulfill the following fundamental requirements:

1. The user will have the ability to set a target final temperature, heating/cooling curve and max/min temperature allowances through GUI on an LCD display.
2. The device will have a temperature floor of at most 0 degrees Celsius, and a temperature ceiling of at least 40 degrees Celsius. The temperature floor requirement includes the ability to freeze pure water.

3. The device will be able to hold temperature to within ±5 degrees Celsius of target temperature at any given time.

*This device will be powered by 120VAC while in use, so it should be able to maintain these temperatures listed above indefinitely i.e. this device should work for consecutive days, let alone multiple hours while in use.

2. Design

2.1 Physical Design

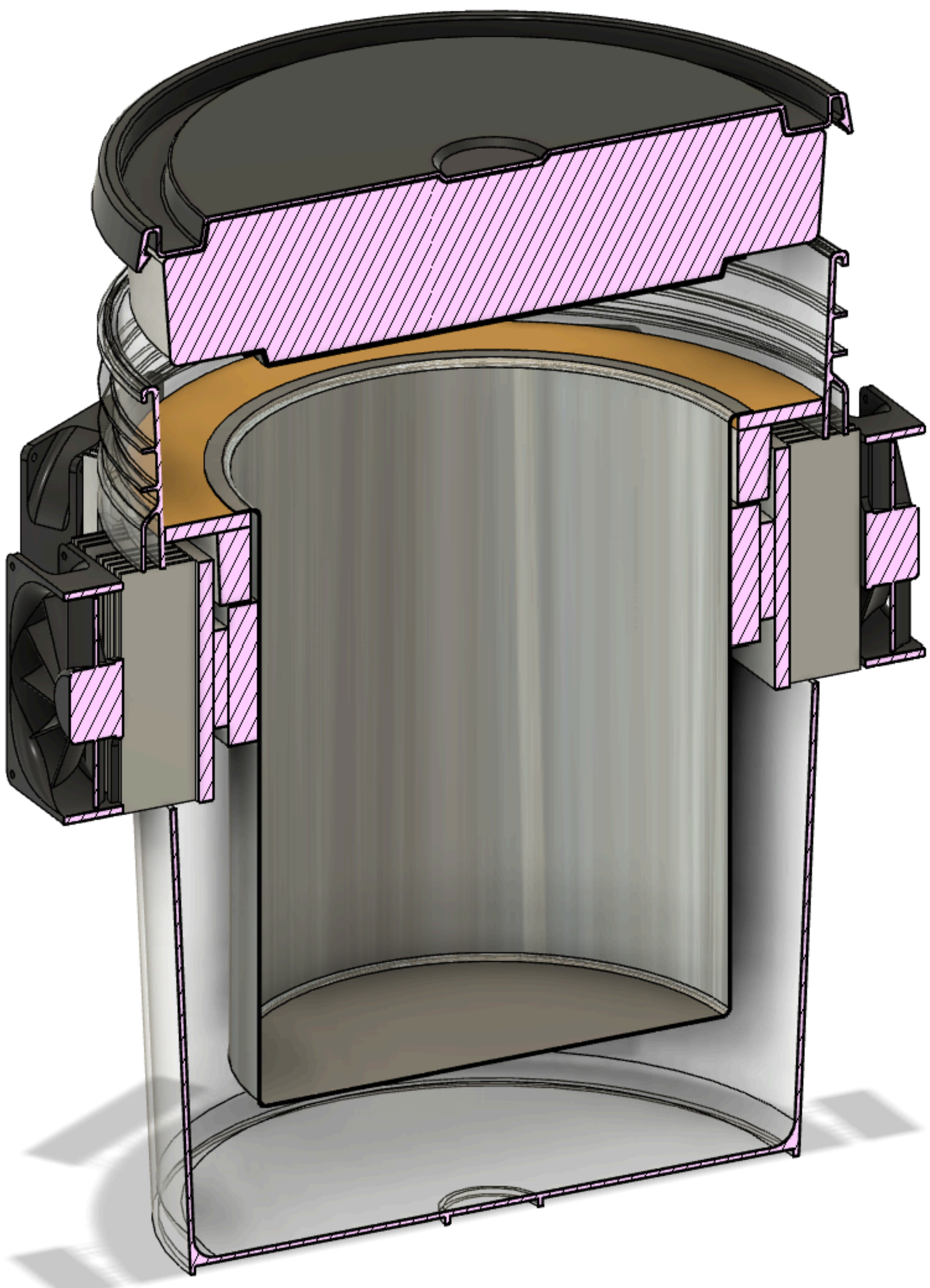


Figure 2: Physical Design of Chamber

We plan to use a 5 gallon bucket as our outer housing, and a metal pot as the inner chamber that nests inside of the 5 gallon bucket. There needs to be enough spacing between the outer chamber and the inner chamber to allow space for our band heater, electrical connections, temperature sensors, and insulation. We plan to use a combination of fiberglass and polystyrene foam to insulate the chamber. Ideally we would only use polystyrene, but since the temperature of the band heater will likely exceed the 40℃ ceiling set for the chamber, there is concern that it could melt polystyrene. Thus, we'll use fiberglass as insulation close to the band heater. However, since fiberglass isn't food safe, we'll use the polystyrene to separate it from any part of the device that could be accessible to the user. The insulation for the lid will be composed entirely of polystyrene foam since it sits directly above the chamber that holds the food. A ring of polystyrene will sit above the fiberglass in the main body of the device. There will also be an acrylic ring above this polystyrene which serves the dual purpose of supporting the inner chamber and ensuring that none of the fiberglass can escape.

As with any culinary appliance, microbial growth could occur in our device. This is one reason why we made sure to use waterproof, foodsafe materials for anything that will come in contact with food. The inner chamber we chose is a stainless steel bain marie intended for professional kitchens. To prevent microbial growth, the user should clean the inner chamber with a disinfectant as they would for any other culinary appliance.

2.2 Block Diagram

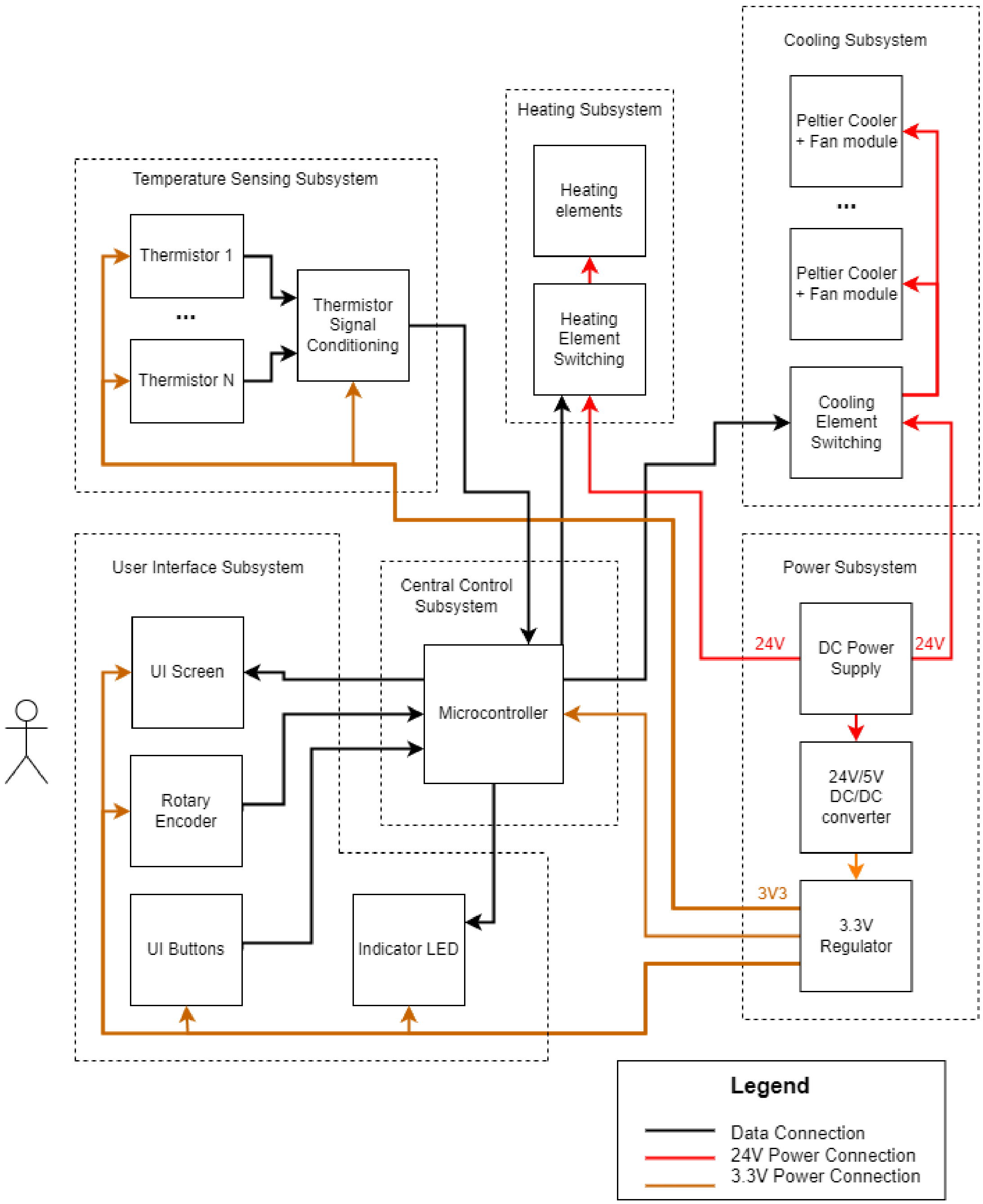


Figure 3: Block Design Diagram

2.3 Functional Overview and Requirements

2.3.1 Power Supply Subsystem

Description: The power supply subsystem has the express purpose of providing the correct voltage and current to power all of the other systems in our project. It consists of various converters, starting off with an off-the-shelf 300W power supply, we will provide 24V to the rest of the systems on the device. We then feed 24V into our Peltiers and to a 24-5V Buck Converter which will then provide at least 10W to our logical level systems. Then, in order to cut down on switching noise, we use a LDO to power all of our 3.3V systems. Based on the LCD power requirements and voltage rating, we may need to increase our LDO power requirement. The power requirements for each module of this subsystem are listed below. These power ratings should be more than enough to power all of our systems in the project. As far as size constraints, we don't really have major limits on size and all these components should be more than enough for our requirements. Price is the biggest optimization in this list, as we use already available parts or inexpensive controllers.

Requirements	Verification
24V Output at minimum 100W	● Wire power supply to wall outlet ● Set up test setup with power supply connected to programmable load with high gauge wire ● Once everything is set up, plug device into wall and verify correct power output on load
5V Output at minimum 10W	● Once 24V power verified, assemble 5V buck on the board ● Use the 5V test points on the board to connect programmable load and program load to correct power output ● Switch the buck on and verify correct power draw is reached with minimal (less than 250mV) voltage sag
12V Output at minimum 5W	● Once 24V power verified, assemble 12V buck on the board ● Use the 12V test points on the board to connect programmable load and program load to correct power output ● Switch the buck on and verify correct power draw is reached with minimal (less than 500mV) voltage sag

3.3V Output at minimum 1W	• Assemble LDO component on the board • Use the 3.3V test point to connect programmable load • Ensure proper power can be drawn from LDO and minimal voltage sag (less than 150mV) is achieved on the output of the LDO

Parts:

- **Off the shelf 24V Power supply**
  - **Minimum 100W requirement**
    - EMH350PS24 **Power Supply**
- **24-5V Buck**
  - **Minimum 10W**
    - RT6285 **Buck Controller**
- **5V-3.3V LDO**
  - **Minimum 1W**
    - AP2111H **LDO**

2.3.2 Central Control Subsystem

Description: This subsystem is meant to control the whole device. At its center is a STM32 microcontroller which should have enough IO in order to read our temperature sensors, control our LCD, and run the control of our heating/cooling systems. The microcontroller will be programmed in C and the clocking is more than enough to perform all necessary functions.The main requirement is to have the required IO we need. So the microcontroller will need to have the correct 5 wire SPI interface required for our display, the required I2c ports for our possible temperature sensor alternatives, the 2-3 analog inputs for our thermistors, and 5-6 GPIO in order to account for the binary control of our heater relay, cooler H-Bridge, and all necessary fans. The microcontroller would also need a RTC subsystem in order to account for the long time heating/cooling curve capability.

Requirements	Verification
Control PWM and polarity of output to Peltier modules	• We will assemble the half bridge component of the PCB • Then, we will apply a 50% PWM signal to each half bridge while applying a 0% PWM signal to the other side • We then read an oscilloscope measurement on the output and ensure that we are getting the proper outputs with the proper polarities
Control user interface and user input from on board touch screen and buttons/encoders	• Assemble the user interface components on the board • Connect screen, buttons, and encoder • Write and flash echo sketch, where user input is piped into feedback LEDs on the board and to the serial monitor
Be able to switch power going to heating modules	• Assemble the heater subsystem and relays • Use slow PWM to switch the system on and off to ensure that relay control is working properly • Verify by hooking up multimeter to output and making sure it works
Have the ability to read temperature from various points on the product and process values to control temperature outputs	• Assembled temperature sensing subsystem on board • Connect external thermistors and TMP sensor sub-boards • Use a similar echo sketch to pipe sensor data to the serial monitor and ensure proper readings by using and external temperature sensor to confirm readings
Capability to set heating curves and execute them over extended periods of time.	• First, control of power going to the heating and cooling subsystems needs to be programmed • Once that is done, program the ability to use the RTC on the STM to follow setting heating and cooling points • Verify temperature stability by programming heating curves and using external thermocouples and temperature sensors to ensure the system is following the proper temperature profiles

Design: Our temperature system will be controlled using a PID loop. This loop will use proportional, integrative, and derivative control to ensure that the chamber temperature stays within our ±5℃ tolerance of the target temperature. We will establish a feedback loop using our temperature control system as well as our temperature sensors and use PWM outputs to have fine grain control over our outputs to our temperature control subsystems. We plan on starting out with some precalculated PID values which will be determined by a dry run of our thermal subsystems. We will put them at 100% duty cycle (or as close as we can get to 100%) and then find out the rate at which the temperature changes. Once we determine the max rate of temperature change, we can easily plug in PID values that control the rate of change of temperature to be below that threshold.

We will then tune our PID loops to update in a timely manner to allow the proper change in output temperature based on our inputs. We can do this by looking at our max temperature vector and then tune the loops so that they do not overshoot our target temp within the values which we determined in our requirements section (+- 5C). We are also going to need to keep PWM timing in mind. Our Peltier coolers can be modulated at a higher frequency because of their solid state nature but the mechanical nature of the relays that control our heaters mean we have to modulate them at a very low frequency in order to keep mechanical wear in mind.

Parts:

- **STM32F103c8 here**
- **Clocks (to run RTC) and decoupling capacitors**

2.3.3 Sensor Array

Description: The purpose of the sensor array is to detect the current temperatures and monitor the food/drink inside of the cooler body. We will achieve this using a number of various sensors, having a parallel layout to ensure that if one does not work during development, we can use the other. Our first sensor alternative is the TMP1075, which is an integrated temperature sensor from TI. We'll make a daughter board with an RJ-12 connector for the I2C communication and mount this daughter board wherever we want to measure the temperature. The other alternative is using the MF52D NTC Thermistor, whose analog signal we have to process using an Opamp and then feed into an analog pin on the microcontroller. These sensors should fall within our planned temperature range (-5°C-45°C) and operate at the logic level voltage of our microcontroller (3.3V). Both of these parts selected fall within these requirements.

Requirements	**Verification**
Accurately relay temperature information (within 5 degrees C) back to STM microcontroller	• Assemble the Sensor Array part of the PCB after verifying both heating and cooling components • Connect sensors and breakout boards • Turn on heating components, measure temperature using both sensor technologies on the board, then measure the same location using an external thermocouple • Compare the temperatures from all three values and attempt to tune the temperature readings from both sensors to fall into the same value as the external thermocouple
Multiple temperature sensing options to verify measurements and ensure proper operation	• Design a PCB with various temperature options • Verify each option independently using the same method as described above • Use both sensing options in order to get redundant measurements in case of sensor malfunction or misuse

Parts:

- **I2C Temp sensor sub board with RJ-12 Connector**
  - **TMP1075**
  - **RJ-12 connector here**
- **External Thermistor with Op Amp Conditioning Circuit**
  - **NTC Thermistor here**
    - **Use this TI Circuit as an example**
    - **Into Analog input of microcontroller**
    - **TLV9002 Opamp**

2.3.4 Cooling Subsystem

Description: Thermoelectric (Peltier) coolers will provide the cooling. These work as heat pumps, so we'll need heat sinks and cooling fans to dissipate the heat they produce. The thermoelectric coolers and fans will be run off the 24V rail, arranged in a 2s2p combination so that the maximum voltage across each thermoelectric cooler is 12V. We want to have the option to run the thermoelectric coolers in reverse while the chamber is heating to prevent their heat sinks from cooling down the chamber. To do this we'll need to power the thermoelectric coolers through an H-bridge so that we can reverse their polarities. The H-bridge will be composed of four N-channel MOSFETs and two bootstrap dual gate drivers. These gate drivers will be capable of driving the high-side N channel MOSFETs, and will be controlled by the microcontroller, allowing us to use PWM to vary the power supplied to the thermoelectric coolers. The cooling subsystem should be able to reach a temperature of 0°C, so all of our Peltier modules should be able to withstand this temperature. Our MOSFETs should be able to handle a drain to source voltage rating of 24V and should have a constant current rating of at least 10A to handle peak currents and the possibility of adding more coolers. The active cooling fans should be able to be powered by our 24v system and the heatsinks have to be the correct size to fit our Peltier module.

Requirements	**Verification**
Cool the product to freezing temperatures (0℃)	• Construct the mechanical components of the build • Attach necessary cooling components and their fans/heatsinks • Operate all the fans using the assembled and tested full bridge setup at the maximum allowable duty cycle of our bootstrapped gate drivers • Ensure we can reach freezing or below at these maximal conditions
Have the ability to switch polarity to help in cases of both heating and cooling	• Assemble full bridge assembly • Operate one side of the half bridge at 0% duty cycle and the other at the max duty cycle • Operate fans in one mode and turn them off in the other • Use external temperature sensors to verify that we are getting the correct temperature coefficient as expected

Parts:

- **Peltier Modules**
- **H-Bridge**
  - **MOSFET and MOSFET Driver**
- **Active Cooling – Fans**
- **Passive Cooling – Heatsinks**

2.3.5 Heating Subsystem

Description: The purpose of the heating subsystem is to provide a positive temperature velocity when we want to quickly heat up the environment in the chamber. We plan to use a Band Heater wrapped around the bottom of the chamber to provide the necessary heating. This is a fairly simple system and it will be controlled through a relay capable of handling 120VAC. The Band Heater should have the capability to reach at least 40°C and hold that temperature. It should also be large enough to wrap around our 8 inch diameter inner chamber. The control systems should be able to be controlled through a single GPIO and handle the necessary voltage/current of our heating/cooling coil.

Requirements	**Verification**
Should be able to heat chamber to above 40 C	• First assemble the relay section of the board • Plug in the heating coils into the relay, making sure we are connected to the NC contacts • Drive the relays using the STM32 and ensure our coil heats up • Use an external temperature sensor the ensure that chamber is reaching desired temperatures after giving chamber ample time to heat up
Allow control of slow PWM using relays	• Connect heaters to the assembled relay section of the board • Use slow PWM (PWM period around 5s) to pulse the heater and attempt to control heating to hover around desired temperature • Will need to design feedback system using temperature sensors to ensure this can properly work

Parts:

- **Nichrome Heating Wire or Band Heater**
  - **24V or 120VAC across the wire.**

- **High Power Relay and Driver**

2.3.6 User Interface Subsystem

Description: The interface system will allow the user to use buttons and encoders as well as a full color LCD touchscreen. Doing so allows them to set heating curves, the time of the device, and certain pre-set heating programs such as thawing of steak or cooling of a drink. While in operation, the LCD will display a visual of the given temperature curve in progress, as well as the current point on the curve. The LCD will also display useful properties such as the current food temperature and internal chamber temperature. Additionally, the user can cancel an operation while it is in progress.

Requirements: The technical requirements for this section are not as important as the UI/UX considerations. We would like to give the user a fine grain adjustment option, in this case the encoder, and a select/cancel option, in the form of the buttons. We also want an alternative input method in case our button/encoder scheme is lacking, in which case the LCD module should have a touch screen. The LCD module itself should be able to be powered off of 3.3V or 5V and communicate with 3.3V logic level GPIO over SPI so that our microcontroller can properly interface with it. It would also be preferable to have a way to offset graphics processing on the LCD in order to prevent having too much MCU compute power dedicated to screen processing.

Requirements	Verification
The device should allow the user to set a preset heating curve from a menu to execute at a predetermined time. Additionally, the device will also allow the option for the user to customize their own temperatures vs times curve.	● First ensure that STM32 is working and can be properly flashed by verifying using a simple blink sketch ● Once that is done, assemble the user interface portion, verifying each component as it build by using echo sketches flashed to the STM32 ● Program the ability to execute heating curves and test using manual heating curves as explained in the RV table for the Central Control System ● Program the screen that depicts the set up of a heating or cooling curve, allow a user to input the curve they want to heat ● Input and test a curve constantly monitoring using an external temperature system outside

	of the products internal sensors
The device should display the current temperature of the food and chamber on the LCD.	• Verify the display can properly work • Print test text to the display with a simple test sketch • Complete verification of temperature sensors • Pass temperature sensor values in real time to the screen to allow user to read them
Device should have the option to set time in order to ensure proper timing of device	• Verify working of RTC system as described in Central Control System • Verify screen, buttons, and encoders are properly working • Program the ability for the user to set the time registers on the device through the GUI on the screen

Design: The first screen will ask if the user wants to operate in Preset Mode, or Entry Mode. If "Preset Mode" is selected, the user is taken to a menu of pre-programmed heating curves for possible options, such as thawing a steak, cooling a beverage, etc. Once the user selects a preset, they will then estimate roughly how much their food weighs by choosing one of three predefined weight classes. Next, they will enter an amount of time, representing how long it will take until their food is finished. Depending on which weight class they entered and the preset selected, there will be a minimum allowed amount of time they can set, as we cannot expect drastic temperature changes for food that weighs too much. Also, for a fixed preset, there will be a general heating curve whose shape will be stretched or shrunk in various ways depending on the amount of time and weight class the user enters. After the preset, weight, and time are entered, the user input is complete and the operation begins.

If "Entry Mode" is selected, the user will also select the weight class and time from present at which their food should be prepared. After this, however, they will be asked how many intervals they wish to control. Essentially, the number of intervals will be the amount of data points (temperature vs time) points on the heating curve that they will set in the next menu. (The device will linearly vary temperature between each data point through the PID loop as described in the central control subsystem section above) After this, the user will manually enter in all the times and the corresponding temperatures that the food should be at said times. Given the weight class the user entered, our program will make sure that there will be no unreasonable temperature changes within a given interval of time. After all, food that is much heavier will have a lower temperature rate of change. After entering in acceptable data points, the operation begins.

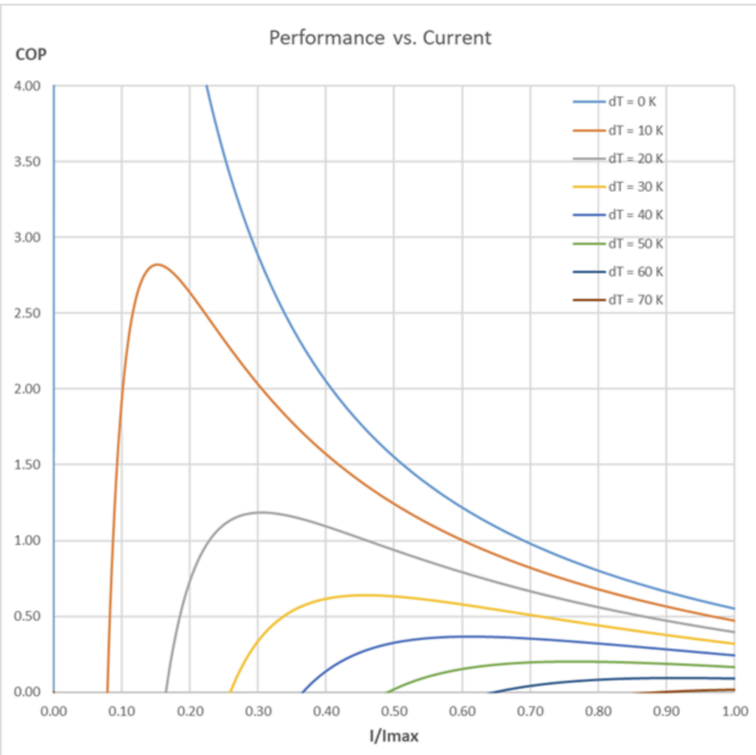
Once the device begins operation, whether Preset Mode or Entry Mode was chosen, the LCD will display a graph of the temperature vs time curve with the key points that the food should reach. The user may re-edit the data as many times as they wish while in operation if they want to change something.

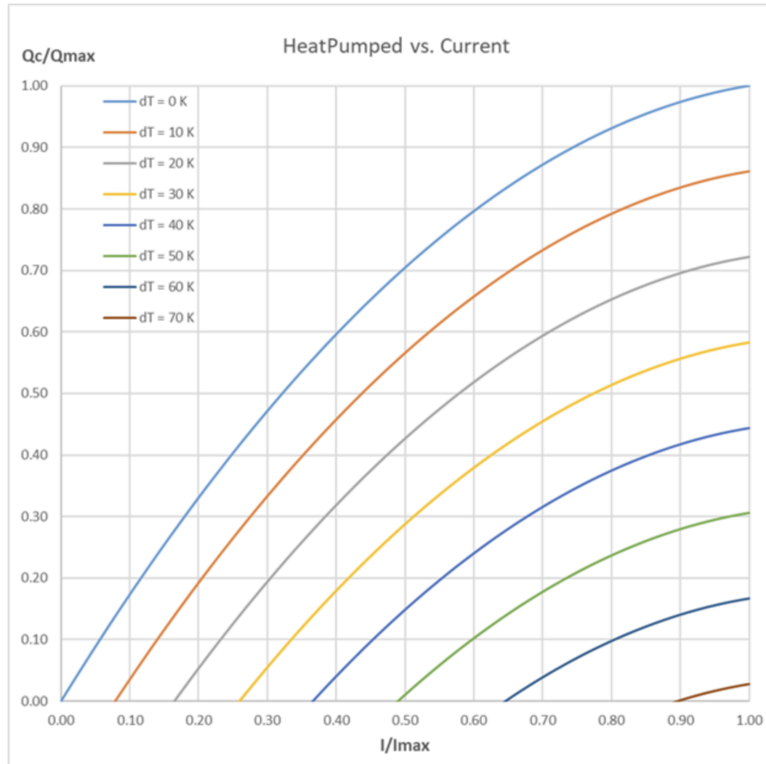
Parts:

- **Color LCD Module**
    - **SPI communication**
    - **Minimum 3" Full Color**
    - **Touchscreen capability for alternate input**
        - **4" ST7796S Hosyond Display here**
- **Buttons**
- **Encoder**
    - **Rotary Encoder for User input**
        - **PEC12R with Switch**

2.4 Tolerance Analysis

2.4.1 Coefficient of Performance





Figures 4 and 5: COP efficiency and Heating Power vs Current trends

The Coefficient of Performance is the efficiency at which heat is removed from the inside chamber with respect to the power drawn from the Peltier cooler, and is strongly related to the current drawn by the cooler and the temperature difference dT between the hot and cold sides. The highest COP is the optimal point to run the cooler. However, the COP changes as the temperature of the chamber, and thus, temperature difference changes. Based on the characteristics, such as the maximum current and heating power, provided from the TEC1-12706 datasheet, we can estimate crucial properties of the chamber.

2.4.2 Analysis

0.1 Error Analysis

The variable we'll be running our error analysis on will be $T_{c,min}$, the minimum temperature achievable by our chamber.

We obtained specifications for the TEC1-12706 thermoelectric (Peltier) cooler module from the Conrad Electronics datasheet [5], and data about the thermal conductivity of fiberglass insulation from The Engineering ToolBox's website [6].

Assumptions

For this error analysis, we'll be making the following assumptions:

1. Peltier cooler hot side temperature $T_h = 30°\text{C}$. This is 10°C above standard room temperature. (20°C)
2. We will use four TEC1-12706 thermoelectric cooling modules.
3. We will operate each Thermoelectric cooling module at a current of 3 [A]. From the Meerstetter Engineering articles [4,5], this seems to be the optimal current to balance temperature gradient with coefficient of performance.

Calculations

First, we need to calculate the thermal energy flow needed to cool our chamber to some temperature T_c. To do this, we'll need the surface area of our chamber, the thermal conductivity of our insulation, and its thickness.

$$(1) \qquad Q_c(chamber) = \frac{k_{ins}SA}{t_{ins}}\Delta T \quad [W]$$

Where:

- $Q_c(chamber)$ is the energy flow from our cold chamber to the environment
- k_{ins} is the thermal conductivity of our insulation, with units $[\frac{W}{mK}]$
- SA is the surface are of our cold chamber, with units $[m^2]$
- t_{ins} is the thickness of our insulation, with units $[m]$
- ΔT is the temperature difference between the inside of the chamber and the external air, in units $[°K]$

Since our inner chamber will be a cylinder with a 10" diameter and a height of 12", and our outer chamber will be 1 inch larger on all sides, we can approximate the surface are of our chamber as:

$$\phi \approx 11\text{"} = 0.28\ [m]$$

$$h \approx 13\text{"} = 0.33\ [m]$$

$$SA \approx 2\pi(\frac{\phi}{2})^2 + \pi\phi h = 0.413\ [m^2]$$

We also know that t_{ins} will be 1" = .0254 $[m]$.

$$t_{ins} = 0.0254\ [m]$$

From The Engineering ToolBox [6], the thermal conductivity k_{ins} for fiberglass insulation has a roughly linear relationship with temperature between 250°K and 400°K which can be approximated as:

$$k_{ins} \approx \frac{1}{7500} T_{ins}(^\circ K) \quad [\frac{W}{mK}]$$

Where $T_{ins}(^\circ K)$ is the temperature of the insulation in Kelvin. Since our equation for $Q_c(chamber)$ is in terms of ΔT, we'll rewrite k_{ins} in terms of ΔT:

$$T_{ins} \approx T_{external} - \frac{1}{2}\Delta T$$

(Assume $T_{external}$ is $300^\circ K$)

$$k_{ins} \approx \frac{300 - 0.5\Delta T}{7500} \quad [\frac{W}{mK}]$$

Inserting these values into equation (1), we can obtain an expression for $Q_c(chamber)$ in terms of ΔT:

$$(2) \qquad Q_c(chamber) = \frac{16.26(300 - 0.5\Delta T)}{7500}\Delta T \quad [W]$$

Next, using the Conrad Electronics datasheet for the TEC1-12706 module [5], we fitted a line to the $Q_c/\Delta T$ curve corresponding to 3 [A]. The tolerance given by the datasheet for thermal parameters was ±10%, so we applied this tolerance to the vertical axis intersect:

$$(3) \qquad Q_c(Peltier) \approx 40 \pm 10\% - \left(\frac{8}{11}\right)\Delta T \quad [W]$$

By setting $Q_c(chamber)$ equal to $Q_c(Peltier)$, we can find the point at which our Peltier coolers aren't able to cool the chamber any further. (ΔT_{max}) Since we have four Peltier coolers, we'll multiply equation (3) by four:

$$Q_c(chamber) = 4Q_c(Peltier)$$

$$\frac{16.26(300 - 0.5\Delta T_{max})}{7500}\Delta T_{max} = 4\left(40 \pm 10\% - \left(\frac{8}{11}\right)\Delta T_{max}\right)$$

The result is a quadratic equation that can be solved using the quadratic formula:

$$.00108\Delta T_{max}^2 - 3.559\Delta T_{max} + 160 \pm 16 = 0$$

$$\Delta T_{max} = \frac{3.559 - \sqrt{3.559^2 - 4(.00108)(160 \pm 16)}}{2(.00108)}$$

The two resulting values for ΔT are:

$$\Delta T_{max} = \{40.966, 50.213\} \quad [^\circ K]$$

We can obtain $T_{c,min}$ by subtracting these values from the hot side temperature T_h:

$$T_{c,min} = T_h - \Delta T_{max}$$

$$T_{c,min} = 30^\circ C - \Delta T_{max}$$

$$T_{c,min} = \{-10.966, -20.213\} \quad [^\circ C]$$

Putting this result in terms of a tolerance:

$$T_{c,min} = -15.590 \pm 4.623 \quad [^\circ C]$$

3. Cost and Schedule

3.1 Cost Analysis

3.1.1 Electrical Cost Analysis:

The electrical components will be retrieved from a variety of sources. First, we plan to order PCBs ourselves. We also plan on asking the ECEB Supply Center for passive components and a few small ICs. Finally, we will order the rest of the parts from Mouser and Amazon.

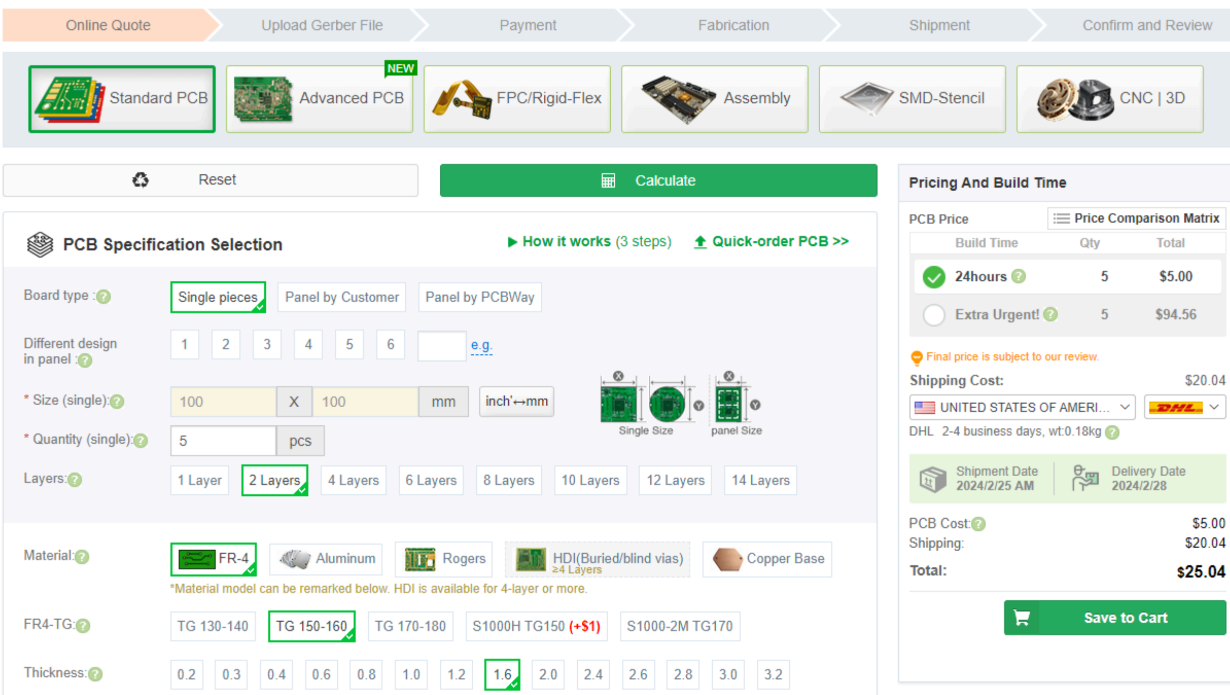


Figure 6: Total PCB cost including shipping

Many of our parts have been left over from other projects, thus meaning many of our passive components and most of our user interface parts are already accounted for. Throughout this project, we are assuming a build volume of 5 boards, where we refrain from buying extras of expensive components such as screens and encoders, in order to cut down on development costs. We chose 5 boards because that is the minimum order quantity from our PCB manufacturer and from past experience, having 5 redundant backups is good during testing of various systems and possible mistakes in assembly. The parts that we still need to purchase for this prototype can be found both online and in the ECEB Service Center.

The parts from the service store required are as follows:

Part	Designators	Service Shop Part #	Quantity per Board	Number of Boards	Total Quantity Needed
1k 0805 Resistor	R1, R5, R7, R38, R39, R43, R44	RMCF0805JT 1K00	7	5	35
10k 0805 Resistors	R6, R8, R18, R20, R19, R21, R25, R27, R26, R28	RMCF0805JG 10K0	10	5	50
Screw Terminals	J2, J3, J4, J10, J11, J12, J13, TH1, TH2	1715721	9	5	45
10uF Capacitors	C17,C18,C38,C39,C40,C41,C42,C43, C19, C20, C21, C22	GRM21BR61 H106ME43L	12	5	60
1.5k 0805 Resistors	R40, R45	RMCF0805FT 1K50	2	5	10
330 Ohm 0805 Resistors	R17, R24, R14, R31, R34, R37	RK73H2ATT D3300F	6	5	30
0.01uF 0805 Capacitors	C29, C30, C32, C33, C14, C15	C0603C103K4 RAC7867	6	5	30
0.1uF 0805 Capacitors	C28, C31, C27, C34, C35, C36, C1, C4, C5, C6, C8, C3	CL21F104ZA ANNNC	12	5	60
3.3V LDO	U1	AZ1117CD-3. 3TRG1	1	5	5
100k 0805 Resistors	R2, R4, R9, R11	RMCF0805JT 100K	4	5	20
1uF 0805 Capacitors	C10, C11	CL21B105KB FNNNG	2	5	10
STM32F401	IC1	STM32F401R BT6	1	5	5
10pF Capacitor	C2, C7	0603N100J50 0CT	2	5	10

Figure 7: Components used from the service shop. (The cost of these is negligible as most of them are free)

The components we still need to purchase at mouser to produce around these 5 boards is as follows:

	Mouser #	Mfr. #	Manufacturer	Order Qty.	Price (USD)	Ext.: (USD)	
1	755-RB550ASA30FHT2RB	RB550ASA-30FHT2RB	ROHM Semiconductor	20	$0.319	$6.38	
2	835-RT6285GSP	RT6285GSP	Richtek	10	$0.761	$7.61	
3	71-CRCW0805220KFKEAC	CRCW0805220KFKEAC	Vishay	100	$0.014	$1.40	
4	603-AC0805FR-07100RL	AC0805FR-07100RL	YAGEO	100	$0.011	$1.10	
5	652-PEC12R4222FS0024	PEC12R-4222F-S0024	Bourns	4	$1.61	$6.44	
6	530-SS-90000-003	SS-90000-003	Bel	20	$0.383	$7.66	
7	595-TLV9001IDCKR	TLV9001IDCKR	Texas Instruments	10	$0.348	$3.48	
8	637-1N4002	1N4002	Diotec Semiconductor	20	$0.071	$1.42	
						Merchandise:	$35.49

Figure 8: Total Cost of electrical components ordered online from Mouser.

Finally, we need to buy an LCD monitor off of Amazon. We found a 4.0" 480x320 TFT Touch Screen here which costs 20.73 after shipping. In total, all of the electrical components will incur a total cost of $81.23 in order to produce 5 boards.

3.2.2 Mechanical Cost Analysis

Below is the cost of all the mechanical parts. We've purchased as many items in person as possible, but some of them need to be online purchases.

Item	Qty.	Cost per unit	Total Cost	Total Cost after Tax & Shipping	Vendor
5 Gallon Bucket	1	$6.98	$6.98	$7.61	Home Depot
5 Gallon Bucket Lid	1	$2.98	$2.98	$3.25	Home Depot
8.25 Qt SS Bain Marie, 8" dia, 9.75" ht.	1	$40.65	$40.65	$59.27	Katom Restaurant Supply
Lid for 8.25 Qt Bain Marie	1	$13.30	$13.30	$19.39	Katom Restaurant Supply
Fiberglass insulation	1	$9.97	$9.97	$10.87	Home Depot
Peltier Cooler + HS + Fan module	4	$23.54	$94.16	$100.60	Amazon
			Total Sum:	$200.99	

Figure 9: Total Cost of mechanical components. The total cost of all these is $200.99.

3.2.3 Total Cost

We could reasonably expect salaries at around $50/hr. Over the course of about 2 months, we'd likely spend an average of 3 hours per day working. So the total labor cost will be $50/hr x 2.5 hr/day x 60 days x 3 people = $22,500. With the electrical cost at $81.23, the mechanical cost at $200.99, the total cost of the project will be $22,782.22.

3.2 Schedule

Week of 2/19	• Design Review with Instructor and TAs (Everyone) • Review order with TA and submit order request (Stefan) • Finish designing PCB (Stefan) • Begin learning STMCUBEIDE (Mitchell) • Begin 3D CAD for mechanical design (Isaac) • Visit machine shop to submit design (Isaac) • Order parts (Isaac and Stefan)

Week of 2/26	Software: (Mitchell) ● Begin designing UI and LCD with rotary encoder and buttons Electrical: (Stefan) ● Finish Schematics ● Get BOM for electrical components that will be on board ● Begin to order components Mechanical: (Isaac) ● Complete mechanical CAD ● Make sure all mechanical parts are ordered ● Submit design to the machine shop for the parts we need machined
Week of 3/4	Software: (Mitchell) ● Continue learning STMCUBEIDE ● Continue designing UI and LCD with rotary encoder and buttons ● Electrical: (Stefan) ● Design review of boards ● Order boards Mechanical: (Isaac) Begin mechanical construction: ● First make sure Peltier coolers can be mounted to the inner chamber ● Figure out how to attach the heating band to the inner chamber ● Decide on a mechanism for attaching the lid
Week of 3/11	Software: (Mitchell) ● Continue developing UI and LCD system ● Begin developing Peltier element logic and start integrating software with PCBs Electrical: (Stefan) ● Waiting for boards ● Help out with software and GPIO bringup for STM32 Mechanical: (Isaac) Continue mechanical construction: ● Finish all drill holes / cutouts in the clamber walls that are needed for Peltier coolers, wires, standoffs, etc… ● Put the chamber together ● Build the lid
Week of 3/18	● Integrate PCB, chamber, and LCD together (Everyone) ● Boards should arrive, begin assembly of boards and spot testing of subsystems (Stefan and Mitchell)

	• Attach Ebox to the side of the device (Isaac) • Connect all wires that need to run through the walls of the chamber (Isaac) • Mount LCD & user interface (Stefan and Mitchell)
Week of 3/25	• Same tasks as last week and additionally: • Begin running thorough tests to ensure system runs properly (Everyone) • Verify modification of subsystems and mechanical components (Everyone) • If time, make changes to the board and order a new one (Stefan)
Week of 4/1	• Same as last week and additionally: • Run through practice demos for next week (Everyone)
Week of 4/8	• Run through practice demos for next week
Week of 4/15	• Mock demo with TA
Week of 4/22	• Final demo with TA
Week of 4/29	• Final presentation

4. Ethics and Safety

When the device is under construction, proper safety procedures are extremely crucial. We will make sure to use necessary PPE when working with tools and components. We will also ensure a clutter-free workspace, check our tools regularly, and clean up the workspace afterwards. High voltage tests will be performed in a controlled environment. Power tools will be turned on only when testing device characteristics and will be powered off directly after use to prevent accidental injury or damage.

High Temperature and Fire Safety Concerns: The drastic temperature changes this device exhibits requires heating elements located toward the bottom of the container. Without proper precaution, this could pose fire hazards, or at the very least, extreme heats unsafe for human proximity. The integral component of the device itself is the temperature sensing, so the internal temperature will never exceed its maximum. Layers of fiberglass will be used around and underneath the container, as well as between the container and the lid which will act as heat insulation. Also, heating wires at high voltages poses significant safety risks, especially if short circuits or other failures occur. Our original plan was to use resistive coils to act as the active heating, but after further consideration, we decided to substitute this heating method with a band heater. The band heater wraps around the food/drink to be heated, reducing the potential electrical fire hazard or even direct burn risk for this subsystem. Also, we will incorporate as many enclosures to isolate subsystems when possible. The electric box will be located outside the outer chamber, ensuring it will not be affected and cause any electric hazards by the extreme temperature changes inside.

As stated in the IEEE Code of Ethics Section I, it is our obligation to “disclose promptly factors that might endanger the public or the environment.” The highest safety priority is that the user operates the device with absolutely no risk to themselves or others and we will take all precautions to ensure this. Keeping this in mind, our user interface system will be efficient yet intuitive to minimize all difficulties inexperienced users could have with the device.

5. References

[1] Conrad Electronic SE. “Specification of Thermoelectric Module.” (), [Online]. Available: https://asset.re-in.de/add/160267/c1/-/en/000189115DS02/DA_TRU-Components-TEC1-12706-Peltier-Element-15V-6.4A-65W-L-x-B-x-H-40-x-40-x-3.8mm.pdf (visited on 02/08/2024).

[2] IEEE. ”IEEE Code of Ethics.” (2024), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 02/08/2024).

[3] Meerstetter Engineering. “Peltier Elements.” (2024), [Online]. Available: https://www.meerstetter.ch/customer-center/compendium/70-peltier-elements#D_Heatpumped%20vs%20Current (visited on 02/08/2024).

[4] Meerstetter Engineering. “Peltier Element Efficiency.” (2024), [Online]. Available: https://www.meerstetter.ch/customer-center/compendium/71-peltier-element-efficiency (visited on 02/08/2024).

[5] Conrad Electronics SE. “Specification of Thermoelectric Module TEC1-12706” [Online] Available: https://asset.re-in.de/add/160267/c1/-/en/000189115DS02/DA_TRU-Components-TEC1-12706-Peltier-Element-15V-6.4A-65W-L-x-B-x-H-40-x-40-x-3.8mm.pdf (visited on 02/08/2024).

[6] The Engineering ToolBox "Thermal conductivity of fiberglass insulation - temperature and k-values." [Online] Available: https://www.engineeringtoolbox.com/fiberglas-insulation-k-values-d_1172.html (visited on 02/08/2024).