Bluetooth Burner

ECE 445 Design Document - Spring 2024

Shaunak Fadnis, Varun Kowdle, Navin Ranganathan

Project #63

Professor Viktor Gruev

Teaching Assistant Zicheng Ma

Contents

1 Introduction

1.1 Problem

Each day, millions of people drink warm coffee, tea, or soup. However, one common challenge faced is maintaining the ideal temperature over time, especially in large sit-down environments or during extended periods of consumption. Moreover, traditional methods like reheating in microwaves can degrade the quality of the drink or food, while passive insulating containers often fail to maintain the desired temperature for long. The repeated process of reheating can be time-consuming and energy-inefficient, making it a less than ideal solution for both home and office settings. This results in a compromised experience, as the taste derived from hot beverages and soups is significantly tied to their warmth. Currently, the beverage warmer market only focuses on two main targets, high-end and low-end, leaving no options for a quality, yet affordable warmer.

1.2 Solution

To address this issue, we propose to make a heating pad with Bluetooth capabilities so that users can adjust temperature to three settings. This allows users to change the heating pad to their ideal temperature to the requirements of the beverage or soup. Integration of Bluetooth allows for a convenient and remote control, enabling users to adjust settings directly from their smartphones. The pad will be energy-efficient and durable, suitable for both home and office use. It will also feature a smart timer function for automatic temperature adjustments or shutdown, promoting convenience and energy savings. Our design prioritizes eco-friendliness by using sustainable materials, aligning with consumer demand for environmentally responsible products. The surface is designed to accommodate a variety of cup and bowl sizes, making it versatile for different beverages and soups. A companion app will offer control over the pad and provide hydration reminders and optimal consumption prompts. Our solution aims to bridge the market gap with a quality, affordable product, enhancing the hot beverage and soup experience.

1.3 Visual Aid

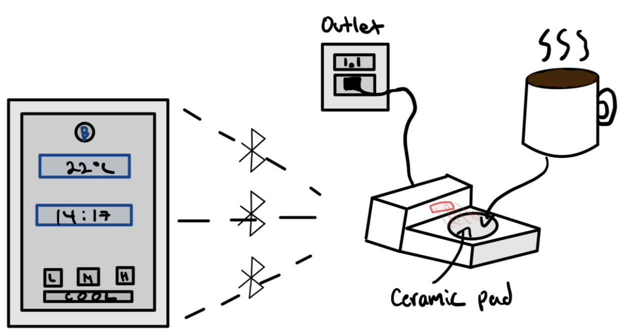

Figure 1: Visual Aid

1.4 High Level Requirements

1. The heating pad should have temperature capabilities of $30 - 60^o$ C for heating and reach at least -10^oC for cooling.

2. The infrared sensor observing the heating pad should be able to identify pad temperature within 1^oC

3. The device should communicate and receive information such as change in temperature via an iPhone app in a range of at least 10 - 20 meters.

2 Design

2.1 Physical Design

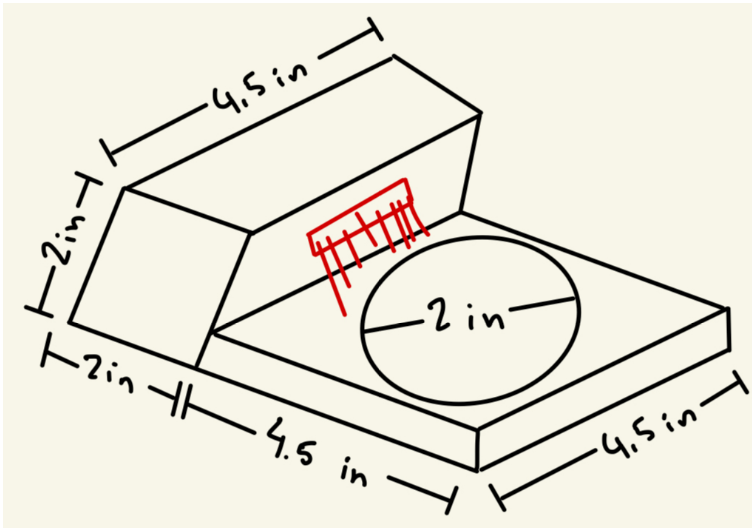

Figure 2: Physical Design - Side View

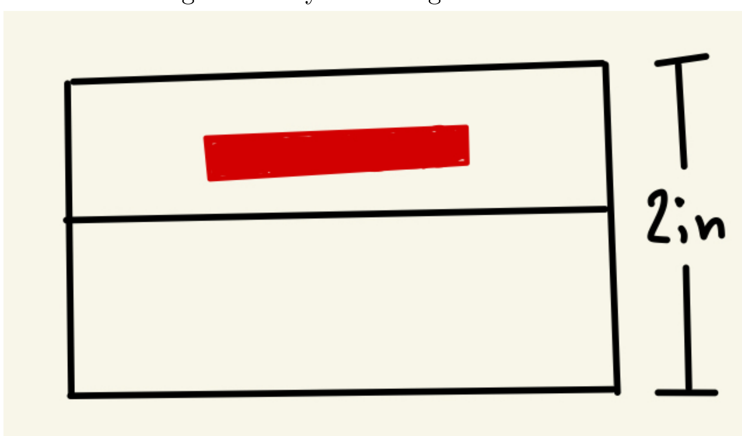

Figure 3: Physical Design - Front View

We have a 4.5 in x 4.5 in unit, that has the heating/cooling ceramic pad on top at roughly a 2 inch diameter. We have our infrared sensor mounted above the ceramic pad at about 2 inches over the flat end of the heating unit, so that it can safely monitor the pad's temperature with little noise. The ceramic pad also has a GPIO capacitve touch wire to for human contact detection. This wire will be inlaid in our ceramic pad in a spiral shape and act as the conductive bridge upon which human contact can be detected. Our enclosure for the circuitry and heating subsystem's will be 3D printed, as polymer or nylon are robust enough to handle the heat outputted by our circuit. The raised up section will contain the majority of electrical components while the pad enclosure will have the heating elements connected in an isolated manner to the main circuitry.

2.2 Block Diagram

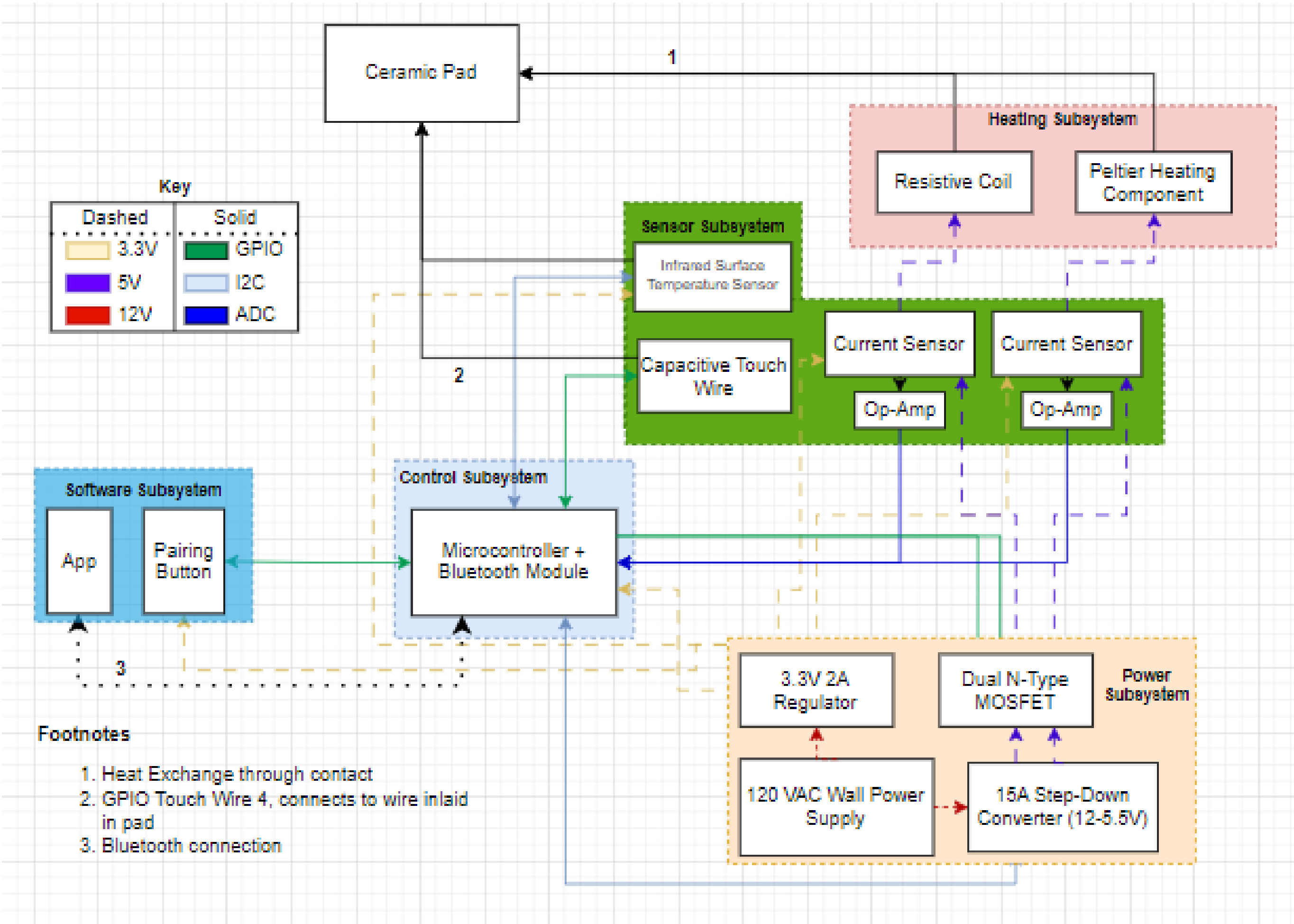

Figure 4: Block Diagram

*Schematics below don't have resistor or capacitor values, these can be found in the datasheets for the ACS711EX Current Sensor [1] , MCP6001U Op-Amp [2], TPS540A50 Buck Converter [3], AP62200WU-7 3.3V Converter [4], and ESP32-WROOM-32E [5]

2.3 Subsystem Overview & Requirements

2.3.1 Control Subsystem

Overview

The control subsystem serves as the central processing unit of the Bluetooth Burner, powered by an ESP32-WROOM-32E-H4 equipped with a Bluetooth module. Its primary functions include controlling the heating/cooling devices, monitoring the burner's status, and facilitating communication with the user's device via Bluetooth. It receives input from the sensor subsystem, which includes an infrared temperature sensor and a current sensor, the former using I2C connection while the latter uses ADC pins. Additionally, the control subsystem links with the software subsystem, which allows users to interact with the burner through a dedicated application. The ESP32 also connects to a button that initiates the Bluetooth pairing process. This subsystem is also responsible for operating the power relay, which switches the MOSFETS responsible for the Peltier module and the Resistive Coil.

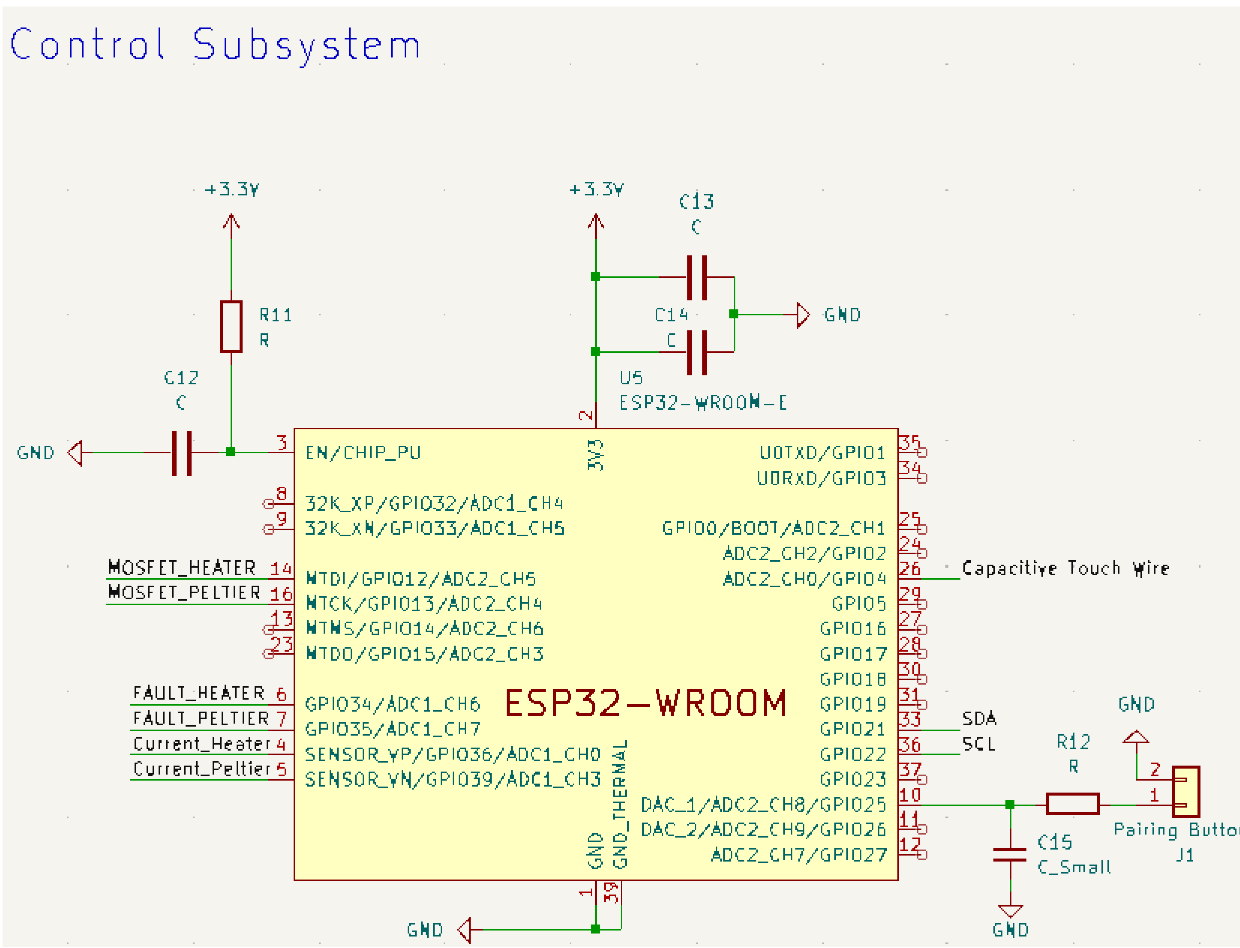

Figure 5: Control Schematic

Interfaces

- Software Subsystem
  - ESP32 microcontroller under the control subsystem includes bluetooth capabilities, which allows it to communicate with a wireless device. Users are able to control heater and receive real time statuses.
- Sensor Subsystem
  - The Control Subsystem uses an Op-Amp to amplify its output signal for the ADC pins on the ESP32 to read the voltage change e.g. change in ^oC.
  - The ESP32 provides pins to detect the Fault pin on the current sensors (active low, with pullup resistor).
  - The IR Sensor uses I2C communication for data transfer.
- Power Subsystem
  - The control subsystem operates the Dual N-type MOSFET that controls the 5V to the heating subsystem.
  - Uses I2C connection to operate the 5V Buck Converter that feeds into the MOSFETS.

Subsystem Requirements

Requirements	Verifications
The ESP32 must initiate the Bluetooth pairing process within 30 seconds upon user command through the pairing button, ensuring a quick and seamless connection with the user's device.	Test the response time of the ESP32 after the pairing button is pressed to ensure it initiates the Bluetooth pairing process within one minute. Complete over 20 tests and record the average connection time.
The ESP32 must consistently read data at a rate of 2.2 MHz from the infrared and current sensors via the I2C communication protocal and ADC pins, ensuring accurate and timely monitoring of the burner's temperature and electrical current.	Conduct a throughput test to confirm that the ESP32 can read data from the sensors at the required rate with minimal errors.
The control subsystem should control power relay, enabling the device to toggle the heating in response to the app's commands and the sensor's feedback.	Test a series of sequences over time where the ESP32 is commanded to toggle the power relay and verify the operation of the peltier heating component

Table 1: Control Subsystem Requirements & Verifications

2.3.2 Heating Subsystem

Overview

The heating subsystem is responsible for heating and cooling the pad upon which our user can place items. The heating device we are using is a Resistive Coil using Nichrome wire. Our cooling device is the CP40236 Peltier Module. The two heating elements will be powered using 5V from our 12V-5V Step-Down Buck Converter. The ESP32 is in control of turning the heating elements on/off. The pad we are using will be made of ceramic and go on top of the heating subsystem. The IR Temperature sensor will relay the surface temperature of the pad back to the ESP32 so it can better control the heating elements.

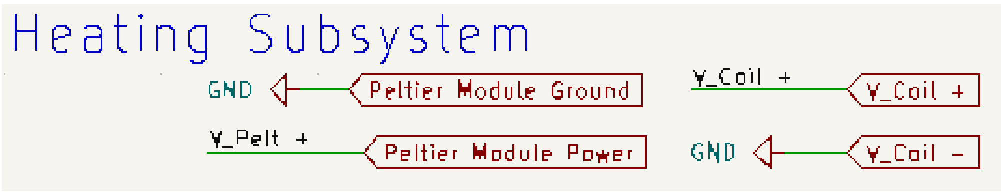

Figure 6: Heating Schematic

Interfaces

- Power Subsystem
  - Both heating elements can receive 5V and upto 15A from the Power subsystem.
- Sensor Subsystem
  - Current sensors monitor the current provided for each of the elements.
  - IR Sensor detects surface temperature of the pad and monitors the output of the heating subsystem.
  - The Capacitive Touch Wire will sense human contact on the pad and report it back to the ESP32, in case the device needs to be turned off.
- Control Subsystem
  - ESP32 controls the MOSFET gates allowing for 5V to go to the heating subsystem.
  - ESP32 can only turn one element on at a time, with a buffer of 5 clock cycles in between turning one device off and another on.

Subsystem Requirements

Requirements	Verifications
Resistive Coil must be able to go up to 65^oC.	Test Coil temperature using power supply to monitor power consumption.
Resistive Coil must be able to impart (heat energy transfer) at least 11J ± 2J per minute.	Measure the temperature rise in 100 grams of water using the peltier module at its highest setting for one minute. Calculate the heat energy imparted $Q = m \times c \times \Delta T$ to verify that it falls within the required range of 11J ± 2J
Peltier module must have a lower-bound temperature for T_C (cold-side temperature) of -10^oC.	Measure temperature using configured IR Sensor or conduct test similar to above using equations for ceramic heat transfer [6]

Table 2: Heating Subsystem Requirements & Verifications

2.3.3 Sensor Subsystem

Overview

The Sensor Subsystem serves to provide real-time monitoring and data collection regarding the operational status of the burner. It includes two current sensors, a capacitive touch sensor, and an infrared surface temperature sensor. These sensors that comprise the subsystem are essential in ensuring the safety, efficiency, and effectiveness of the heating process, each serving a unique purpose. The two current sensors are connected to the power lines for the heating subsystem to monitor the power output, while the GPIO capacitive touch pin helps to detect human contact through the inlaid wire on the ceramic pad, and the infrared surface temperature sensor provides temperature control to prevent overheating. The sensor subsystem communicates relevant information to the controller and power subsystems using I2C for the IR sensors, and GPIO-ADC pins for the MOSFETS and the current sensors respectively. In the case of the ADC pins connected to the current sensors, the output of our current sensors goes through an amplifier to take it from a range of 0-110mV to a readable range on the ADC pin.

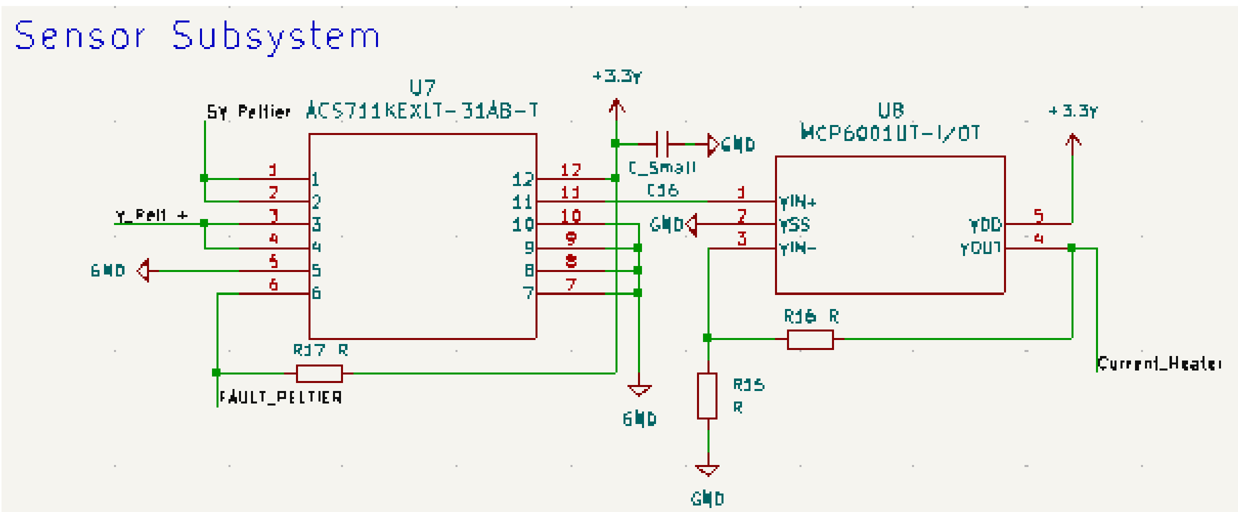

Figure 7: Peltier Current Sensor Schematic

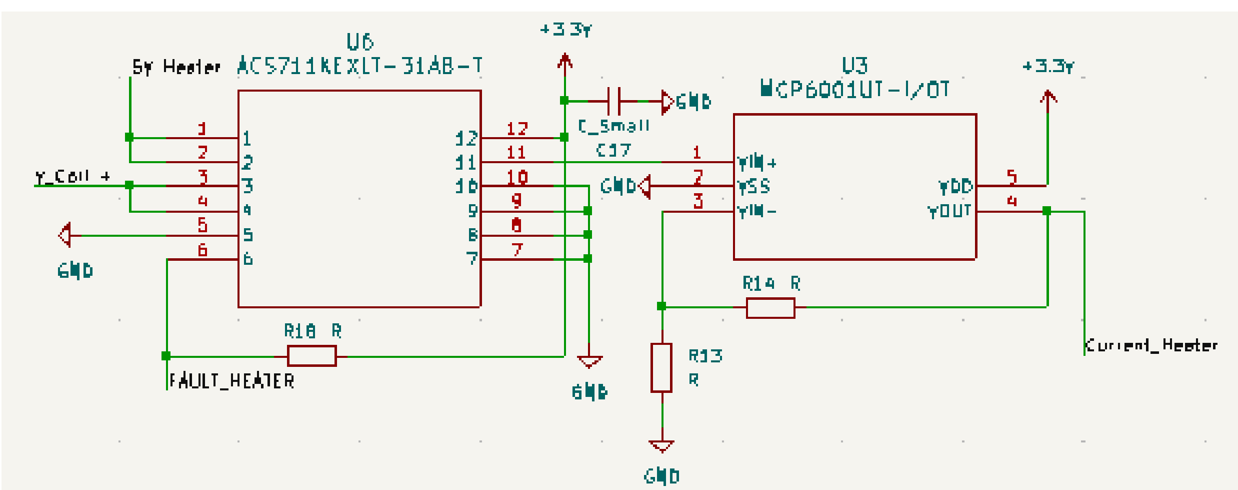

Figure 8: Coil Current Sensor Schematic

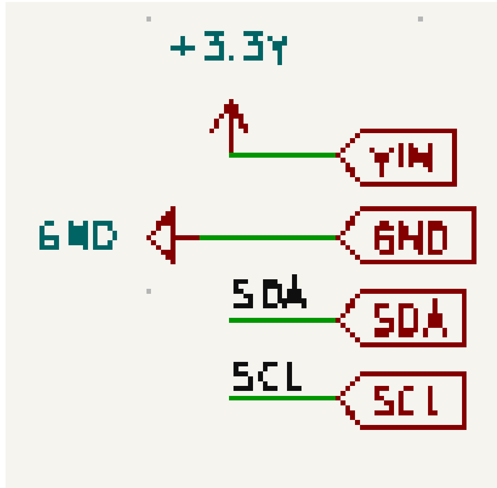

Figure 9: IR Temp. Sensor Schematic

Interfaces

- Power Subsystem
  - Current sensors monitor the 5V going to the Heating subsystem.
  - 3.3V is provided for Current sensors and IR sensors.
- Control Subsystem
  - GPIO touch pins send or receive data to/from the microcontroller, and will trigger the microcontroller to turn off the heating subsystem in the event of human contact on heating pad.
  - Current sensors connect to ADC pins for output data and GPIO pins for Fault detection.
  - IR sensor uses I2C connection to communicate with ESP32.
- Heating Subsystem
  - IR sensor monitors the surface temperature of the ceramic pad.

Subsystem Requirements

Requirements	Verifications
Infrared sensor must accurately measure the surface temperature of the heating path with a precision of ± 1°C, providing feedback to the control unit to ensure the user set level is maintained	Conduct a series of heat tests by providing the sensor with temperature data in various ranges, as well as changing them quickly to measure adaptability of the heating pad temperature
The current sensors should have an accuracy of $\pm 0.5\%$ of the measured value to ensure precise monitoring of electrical consumption and to detect any unusual current draw.	Test each sensor individually with various amounts of current to measure if it is within the desired threshold with respect to its designated heat setting (over multiple trials)
The GPIO capacitive touch wire should make sure to respond within 0.1-0.5 seconds of a human touch to protect users from any heat damage	Test the GPIO wire before implementation, then, with slight heat, check over the reaction time of another object in contact with the heating pad over several trials.
Both the current and infrared sensors must be able to read the data of their respective purposes despite negligible obstacles or noise	Run through the various testing and validation processes for each sensor while providing minimal noise that should be voided from the sensors' readings

Table 3: Sensor Subsystem Requirements & Verifications

2.3.4 Power Subsystem

Overview

This subsystem is responsible for powering the control subsystem, the heating subsystem and the control subsystem. The Power subsystem takes in 12V from an AC/DC wall-adapter. This 12V is stepped down by the TPS540A50 Buck Converter to 5V to be used by the heating subsystem. The 12V is also Vin for the 3.3V AP62200WU Converter that powers the Control and Sensor subsystems. The 5V for the heating subsystem is also controlled by a Dual N-type MOSFET with the gate's connected to GPIO pins on our ESP32. These MOSFETS are able to provide our desired power to the heating subsystem and has a P_{max} of $48W$.

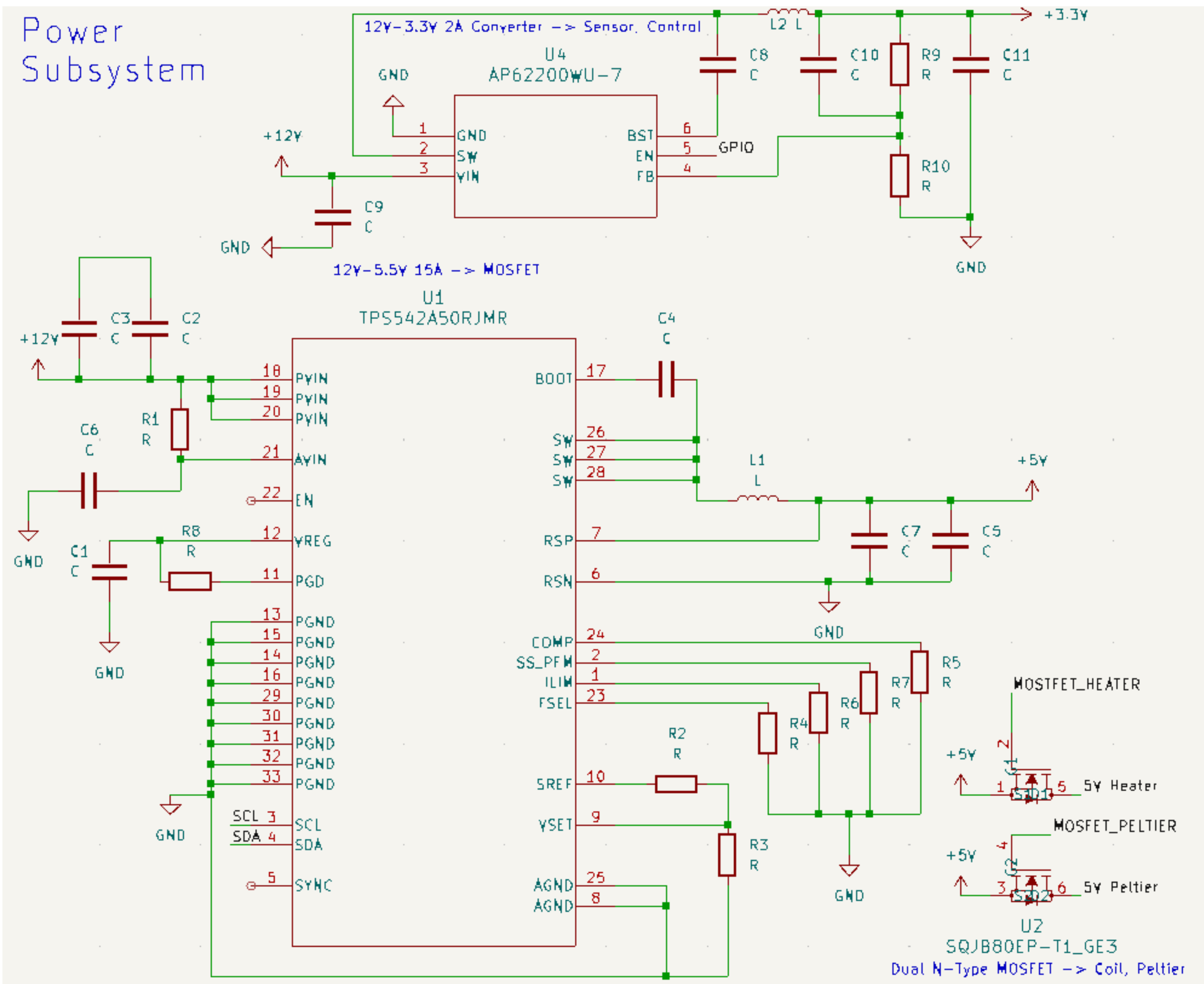

Figure 10: Power Schematic

Interfaces

- Control Subsystem
  - Power subsystem provides 3.3V ± 0.3V to power the ESP32 microcontroller.
  - The Step-Down Buck Converter is controlled by the ESP32 using I2C master-slave connection.
  - The ESP32 controls the gates of the MOSFETS to let 5V through.
- Sensor Subsystem
  - IR Sensor and Current sensors are powered by 3.3V regulator of power subsystem.
  - Current sensors connects to the 5V output of the MOSFETS to monitor power to heating subsystem.
- Heating Subsystem
  - 5V and up to 15A of power is provided to the heating elements of the heating subsystem.

Subsystem Requirements

Requirements	**Verifications**
The power subsystem must provide 3.3V ± 0.1V to the control subsystem and sensor subsytem.	Conduct a precision voltage measurement using a calibrated digital multimeter. Verify the output is within 3.3V ± 0.1V under no-load and full-load conditions to test the regulator's stability and load regulation.
The power subsystem must provide 5V as output from the Step-Down Buck Converter.	Conduct a precision voltage measurement using a multimeter.
The power subsystem must provide up to 15A as output current from Buck Converter.	Check both circuit using multimeter and output from I2C data to determine output.
MOSFETS must be capable of switching to allow 5V to heating subsystem.	Test MOSFETS at low voltage, but above V_{gs} threshold voltage [3] to test ESP32 control over switch. Use multimeter to test connection from Buck Converter and then output after switching.

Table 4: Power Subsystem Requirements & Verifications

2.3.5 Software Subsystem

Overview

The software subsystem of the Bluetooth Burner serves as the graphical user interface for interacting with the device, providing a seamless and intuitive experience. Moreover, the primary function of the software subsystem is to enable users to initiate the Bluetooth pairing process, adjust temperature settings, and monitor the status of their Bluetooth Burner in real time. At the heart of the subsystem lies the bluetooth communication facilitated by the ESP32's onboard module, which bridges the gap between the user's commands and the device's operational logic in the control subsystem. By using ESP32 communication protocols, the software subsystem is able to send and receive data, which will utilized for visual indicators of the device's current state and providing feedback on system statuses as detected by various subsystem.

App Design

The software subsystem is broken down into two components: the client and the server. In this case, the ESP32 will serve as a server and a mobile device will serve as the client. Each component will have specific responsibilities. For the server side, we will be using Arduino BLE modules to communicate with our client side code code. Data will be sent through the bluetooth channel configured on both server and client. On the client side, users will have to connect to bluetooth via the iOS interface; however, once a device is connected, the app will determine if its the ESP32 by a set UUID and then be able to transmit data. The client application is developed on Swift, using the CoreBluetooth module's CBCentralManager to communicate with the ESP32. Additionally, the app's UI will consist of various event listeners to be able to remotely control the ESP32.

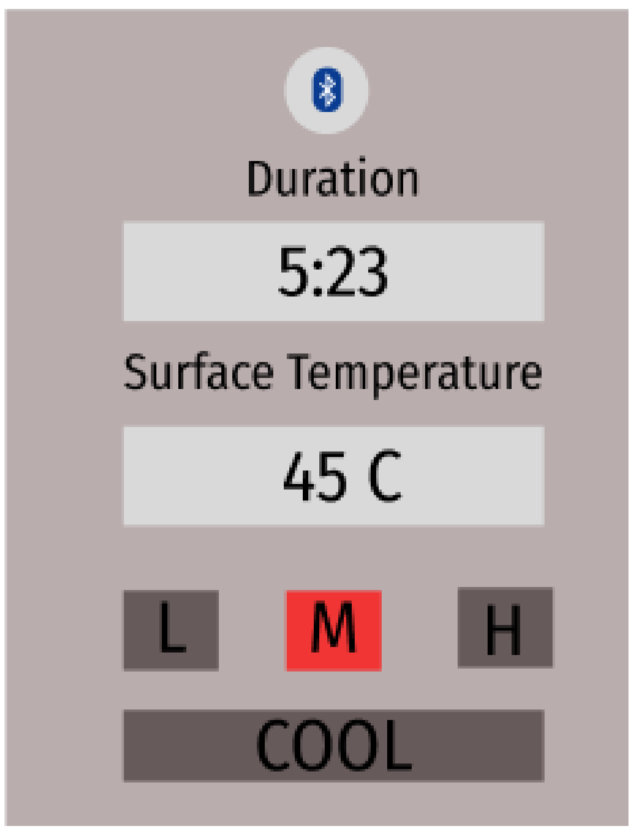

Figure 11: Software Visual

Interfaces

- Control Subsystem
  - Interacts with the Control Subsystem by handling wireless communication between the user's iPhone app and the Control Subsystem's ESP32 microcontroller.
  - Receives data from the software subsystem including temperature chance and power management
  - Ensures that user commands are conveyed and system status updates are received for real-time monitoring and control.
- Sensor Subsystem
  - Receives and processes data from the sensor subsystem via ESP32 from the control Subsystem. This information is interpreted for user comprehension and system reports
  - The processed sensor data informs the Control Subsystem's decisions, enabling it to adjust the Power Subsystem's output to maintain desired operational conditions.

Subsystem Requirements

Requirement	**Verifications**
Software subsystem must be able to communicate with the control subsystem to change the temperature to three pre-set settings, and a single cooling, with a precision of 1^oC for each setting.	Place a mug with room temperature water on the pad. Test each setting by adjusting the setting on the app to the various temperature and observing the water's temperature over time.
Phone app must be able to connect to device within 10 - 20 meters of it	Connect phone app to bluetooth warmer and gradually walk away from it. At each increase in meter, ensure that the connection is stable and the temperature of the warmer can be increased.
Pairing button must initiate pairing search for Bluetooth devices and successfully connect within at most one minute	Pair bluetooth device with phone over twenty times during the course of a week. At each pairing, ensure that the connection was successful, and record the pairing time duration to ensure that average connection time is below one minute.

Table 5: Software Subsystem Requirements & Verifications

2.4 Tolerance Analysis

When looking at our device, the first section that we wished to analyze was the heating of the ceramic pad. Using small estimates of 10cm x 10cm x 2mm for the pad itself we can calculate how much energy we would need to transfer from the resistive coil using the specific heat capacity formula (with an estimate 100 J/kg^{o}C specific heat, and $2500kg/m^3$ ceramic density).

$$\text{Area} = L \times W = 0.1\,\text{m} \times 0.1\,\text{m} = 0.01\,\text{m}^2$$
$$\text{Volume} = \text{Area} \times \text{Thickness} = 0.01\,\text{m}^2 \times 0.002\,\text{m} = 0.00002\,\text{m}^3$$
$$m = \rho \times \text{Volume} = 2500\,\text{kg/m}^3 \times 0.00002\,\text{m}^3 = 0.05\,\text{kg}$$
$$Q = mc\Delta T = 0.05\,\text{kg} \times 1000\,\text{J/(kg°C)} \times 30\,\text{°C} = 1500\,\text{J}$$

To get the 1500J we have to check the power requirements from the resistive coil. We get

$$P = \frac{1500\,\text{J}}{30\,\text{s}} = 50\,\text{W}$$

$$I = \frac{50\,\text{W}}{5\,\text{V}} = 10\,\text{A}$$

, using the equations [7]:

$$P = \frac{E}{t},\, t = 30sec$$
$$I = \frac{P}{V},\, V = 5V$$

Looking at the equations above we would need 5V and 10A, which our TPS540A50 is capable of reaching, as well as the MOSFETS we are using to control them.

Another aspect of our project that poses a potential risk is making sure that our heating pad works accurately and reliably. With various different bowl/cup sizes and materials, it may prove challenging to maintain the desired temperatures precisely. Below are some examples of different heat levels we may use and different materials of the containers. For each of these temperature ranges, using the Heat transfer calculation, we have represented an ideal heat coefficient of the material that would be suitable for temperature maintenance at the desired levels, with a ± 0.2 to account for using the average surface area of our container being 0.03 m. When using the calculations from above we can see there is than enough power and current coming from the Nichrome coil that it will be able to transfer the necessary heat.

Table 6: Tolerance Range

Temperature Range (^{o}C)	**Heat Transfer**	$\sim U$
Low (40-50°C)	U x 0.03 x 25	Approx 1.33 ± 0.2 W/m^2°C
Medium (50-60°C)	U x 0.03 x 30	Approx 1.11 ± 0.2 W/m^2°C
High (60-70°C)	U x 0.03 x 35	Approx 0.95 ± 0.2 W/m^2°C

3 Cost & Schedule

3.1 Cost Analysis

3.1.1 Estimated Cost of Development

After doing some research into what the average UIUC graduate from Computer Engineering would make annually, we established that the wage of each team member would be about $52 per hour. We estimated that each team member will do about 15 hours of work per week for the next 8 weeks, coming out to a cost of $6,240 per teammate, or $18,720 overall for labor costs.

3.1.2 Components Breakdown

Component (Link)	Vendor	Quantity	Total Cost
ESP32-WROOM-32E-H4	DigiKey	1	$2.68
Peltier Thermoelectric Module	DigiKey	1	$14.57
Digital IR Temperature Sensor	Amazon	1	$15.99
5.5V I2C Buck Converter	DigiKey	1	$5.46
Hall Effect Linear Current Sensor	DigiKey	2	$3.02
12V AC/DC Wall Converter	Amazon	1	$11.99
Dual P-Type MOSFET	DigiKey	1	$1.55
12V-3.3V Buck Converter	DigiKey	1	$0.41
Op-Amp	DigiKey	2	$0.60
MCP4018 Digital Potentiometer	DigiKey	1	$0.68

Table 7: Bill of Materials

3.1.3 Total Cost

When adding together our total labor cost ($18,720) with the total cost of our components broken down in the table (total componenets cost of $59.60), we compute a total cost of **$18779.60** for our Bluetooth Burner project.

3.2 Schedule

Week	Major Deadlines	Shaunak	Varun	Navin
2/26	Design Review	Finalize PCB	Finalize PCB	Finalize PCB
3/4	First Round PCB Orders	Order Parts	Start front-end of application	start front-end of application
3/11	Spring Break	N/A	N/A	N/A
3/18	Second Round PCB Orders	Order PCB/start backend	Start backend for bluetooth interface	Start backend for bluetooth interface
3/25		Soldering	Soldering	Soldering
4/1		Test Heating System	Test Control System	Test Sensor System
4/8		Design Enclosure/Test Feedback loop with heating and control systems	Test bluetooth interface with control system	Test capacitive touch sensor
4/15	Mock Demo	Finish Unit testing/Begin full device testing	Finish Unit testing/Begin full device testing	Finish Unit testing/Begin full device testing
4/22	Final Demo	Last minute changes	Last minute changes	Last minute changes
4/29	Final Paper	Finalize Paper	Finalize Paper	Finalize Paper

Table 8: Weekly Schedule

4 Ethics & Safety

As our product is something that will be heated, we will make sure to have ethical and safety precautions that can help avoid burning the users or other objects within a close vicinity. As the team developing the project, we are also aware of the precautionary measures required as we work in the lab, and we will make sure to be as careful and responsible as possible to avoid any misuses of any potentially harmful materials. Our product also contains a bluetooth aspect, so we will make sure to respect the user's privacy when connecting their device.

As per the context of IEEE Code of Ethics I-2 [8], we will make sure that users of our product understand the correct methods of usage of our product, specifically that it is not meant in any way, shape, or form to harm others or burn other items. Our team will emphasize that the heating pad may still be warm after it is turned off for a bit of time, and there will be a temperature limitation to mitigate risk of overheating.

References

[1] Current sensor datasheet. hhttps://www.digikey.com/en/products/detail/allegro-microsystems/ACS711KEXLT-31AB-T/3868195.

[2] Op-amp datasheet. https://ww1.microchip.com/downloads/en/DeviceDoc/MCP6001-1R-1U-2-4-1-MHz-Low-Power-Op-Amp-DS20001733L.pdf.

[3] Tps540a50 datasheet. https://www.ti.com/lit/ds/symlink/tps542a50.pdf.

[4] 3.3v converter datasheet. https://www.diodes.com/assets/Datasheets/AP62200_AP62201_AP62200T.pdf.

[5] Esp32-wroom-32e datasheet. https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf.

[6] Thermal conductivity of ceramics. https://global.kyocera.com/fcworld/charact/heat/thermalcond.html.

[7] Nichrome wire heating. https://wiretron.com/nichrome-wire-tech-tips/.

[8] Ieee code of ethics. https://www.ieee.org/about/corporate/governance/p7-8.html.