Induction Bottle (I - Bottle)

ECE 445 Design Document

Evan Dawson (evanfd2), Michael Tzeng (mhtzeng2), Homero Vazquez (hvazqu6)

Team X

TA:

Spring 2022
1 Introduction

1.1 Objective

The Induction-Bottle utilizes wireless magnetic induction to both power a heating coil and charge a battery in order to monitor temperature and control electronic modules, and the result is a portable liquid heating device.

The structure of the project revolves around an induction heating coil and the electrical systems necessary to power and control the coil. This design document is the organization of these electrical components necessary to properly implement the temperature monitoring, control, and adjustment. First, a series of Microcontrollers, assigned as the Central Computing Unit (CPU) and the Temperature Control Unit (TCU), will control central communications throughout the I-Bottle system. The TCU will receive input data from the user interface and various sensors, and will find the current temperature of the system alongside the desired temperature of the user. Once these values are found, the CPU will receive this data from the TCU, and data will be sent to an LCD display for the user, and a temperature control signal will be sent to the bottle, which instructs the power supply control bridge what setting the induction heating coil should be set to.

1.2 Background

Our project aims to implement existing technology into a portable, easy to use, and efficient device. Wireless induction charging has been in smartphone technology for years now, and induction heating kettles have been increasing in popularity over gas powered stoves. Some houses have even begun implementing wireless induction stoves into kitchens, and electric stovetops have proven a reliable cooking method. The I-Bottle will combine the induction heating and wireless charging technologies into a bottle-pad system, where wireless induction can take place to induce current into a rechargeable battery and into the heating coils.

The wireless pad will be a circular structure housing the LCD display, the user input keypad, the main PCB with the CPU, TCU, and rechargeable battery. This pad will have a signal physical connection to the bottle for a temperature control bit from the CPU, allowing for temperature control and monitoring only while the wireless power supply is connected. The bottle will lose the temperature control function while in portable mode, but will utilize any excess power remaining in the heating coil to maintain temperature control for a short time on
the go. There will also be a cap module to seal the system and make temperature changes easier to reliably implement. The cap will have a pressure and temperature detection system in order to ensure the bottle does not explode from high pressure within the cap.

Finally, the main PCB will have the LCD display and keypad directly connected (Surface mounted) and the shell of the pad system will have holes cut for easy mounting implementation. The physical connection to the bottle will be made via the pad shell as well, with design focus around connections to the PCB. The physical connection will be made as a bridge between the wireless induction generator and the heating coils, and a series of transformers and transistors will take the temperature control signal and convert it into a power output level for the heating coil.

1.3 High level requirements

1.31 Induction Heating Coils: Operation and Control

1.32 Sensor: Accurate Temperature Monitoring

1.33 Wireless Induction Charging and Power Supply

1.34 User Interface and Interactivity

1.35 Temperature Computations and Control Signals Distribution
2 Design

2.1 Block Diagram
2.2 Physical Design

Physical Power Control Implementation and State Machine
2.3 Visual Aid
3 Requirements & Verification Tables

3.1 Sensor Unit

3.11 Waterproof Temperature Sensor

One temperature sensor needs to be waterproof to measure the temperature of liquids. The part number for this component is DS18B20 according to Sparkfun. It takes 3.0-5.5 volts as input. It has a temperature range of -55°C to 125°C. Its accuracy is within ±0.5°C from -10°C to +85°C. This sensor will satisfy all of our requirements for measuring the temperature of the liquid in our container, and will connect to our temperature control unit microcontroller.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The accuracy of the Temperature sensor should be within ±0.5 degrees Celsius</td>
<td>1. Utilize an external thermometer used for checking oil or liquid temperature to check accuracy of the waterproof temperature sensor.</td>
</tr>
<tr>
<td>2. Should work concurrently with other sensors to communicate with the TCU microcontroller. Should also function between 3.0 V and 5.5 V.</td>
<td>2. Various input voltages will be tested to ensure proper functionality of the sensor.</td>
</tr>
</tbody>
</table>

3.12 Base Heating Plate Temperature Sensor

One temperature sensor needs to be used to measure the temperature of the heating element sending heat to the bottle. We will utilize the TC-SA Type K/T Surface Thermocouple with Adhesive Backing, found on iothrifty.com, and insert the probe of this device into our base heating plate. This sensor will connect to our temperature control unit microcontroller.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The sensor should accurately measure the temperature of the base plate.</td>
<td>1. Utilize an external thermometer like an infrared sensor to measure surface temperature of the base plate. Will need to determine correlation between varying thermocouple voltage and corresponding temperature.</td>
</tr>
<tr>
<td>2. Should work concurrently with other sensors to communicate with the TCU microcontroller.</td>
<td>2. Various input voltages will be tested to ensure proper functionality of the sensor.</td>
</tr>
</tbody>
</table>

3.13 Pressure Sensor

We will use the MPL115A1 barometric pressure sensor to measure the pressure of our closed system, so when the bottle has a cap on it. We will need the pressure sensor to determine safe
pressures for our bottle to be at and build in a failsafe if our bottle reaches a dangerous pressure.

3.14 pH Sensor

3.15 Conductivity Sensor

### 3.2 User Interface Unit

#### 3.21 LCD Display

We will utilize the LCD1602 Module, which will display the current temperature, pressure or humidity for the user to view. This component will connect to our system control unit microcontroller.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The LCD Display should be able to display whatever our microcontroller tells it to.</td>
<td>1. Test LCD Display separately to make sure the display works as advertised.</td>
</tr>
<tr>
<td>2. Should work concurrently with other temperature sensors that communicate with our temperature control unit to obtain the calculated temperature of our liquid.</td>
<td>2. Obtain and display calculated temperature data on the LCD Display, and verify that these temperature readings are the same as what we’ve measured from our individual sensors.</td>
</tr>
</tbody>
</table>

#### 3.22 Temperature control button pad

We will use a 4x4 Matrix membrane keypad to allow the user to input the desired temperature of their liquid. It is rated for a maximum of 24 VDC and 30 mA. It’s operating temperature is between 0 and 50 degrees celsius (32 to 122 degrees Fahrenheit). The button pad will integrate seamlessly with our LCD display through our TCU and SCU.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The button pad should allow the user to input the desired temperature of their liquid. It should display this desired temperature on the LCD Display,</td>
<td>1. Test the button pad input and output this on the LCD Display. Next would be adding some text signifying the desired temperature is being input.</td>
</tr>
<tr>
<td>2. Communicate with the TCU to store the desired temperature.</td>
<td>2. Test TCU microcontroller to ensure proper storage of data from peripherals such as button</td>
</tr>
</tbody>
</table>
3.3 Power Supply Unit

3.31 Wireless Charging Pad (AC)

3.32 Rechargeable Battery

3.33 Power Regulator and Impedance Matching System (voltage regulation)

3.34 RLC Tank Circuit (Heat Distribution through Induction Coils)

3.35 Induction Generator

3.4 Control Unit

3.41 Temperature Control Unit

We will use the ATtiny85 20s??? microcontroller to communicate with our sensor modules, including our two temperature sensors measuring liquid temperature and base plate temperature, as well as a pressure sensor and other sensors (pH, conductivity) to determine the purity of the water.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The Temperature Control Unit should be able to communicate with the</td>
<td>1. Display button pad input on LCD display with SCU microcontroller.</td>
</tr>
<tr>
<td>temperature control button pad.</td>
<td></td>
</tr>
<tr>
<td>2. Communicate with the temperature sensors measuring temperature of the</td>
<td>2. Test each temperature sensor by routing data to LCD display through TCU.</td>
</tr>
<tr>
<td>liquid and base heating pad.</td>
<td></td>
</tr>
<tr>
<td>3. Send data to the SCU</td>
<td>3. Test communication between TCU and SCU by sending test signals from the</td>
</tr>
<tr>
<td></td>
<td>TCU to the SCU and printing on LCD display.</td>
</tr>
</tbody>
</table>
4. (extra) Communicate with the pressure sensor to measure pressure of system when closed (capped)

5. (extra) Communicate with the pH sensor to measure the pH of the liquid

4. Test pressure sensor by routing data to LCD display through TCU. Compare pressures of open and closed systems to confirm that the sensor is sending correct data

5. Test pH sensor by using a pH strip to determine the accuracy of our sensor.

3.42 System Control Unit

We will use the ATtiny85 20s??? microcontroller to communicate with our system modules including the LCD Display, temperature control button pad, power bridge, as well as our temperature control unit.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The System Control Unit should be able to communicate with the LCD display.</td>
<td>1. Display test input from SCU on LCD display.</td>
</tr>
<tr>
<td>2. Communicate power routing data to the power bridge.</td>
<td>2. Test each power level (low, med, high) and confirm that the correct path for power is taken for each respective selection.</td>
</tr>
<tr>
<td>3. Receive data from the TCU</td>
<td>3. Test communication between TCU and SCU by sending test signals from the TCU to the SCU and printing on LCD display.</td>
</tr>
</tbody>
</table>

4 Tolerance Analysis

The most important tolerance in our project is the temperature measurement from our temperature sensors, along with the desired temperature set by the user.

First, to calculate the temperature of the liquid, we will use both the liquid temperature sensor and the temperature of the base plate to calculate a more accurate temperature of the overall liquid.
5 Cost and Schedule

5.1 Cost Analysis

Labor per person:
$30/hour * 10 weeks * 10 hours/week = $3000 per person to work the next 10 weeks on project
3 people → 3 * $3000 = $9000 labor costs

Parts

5.2 Schedule

6 Ethics and Safety

7 Citations