Dryer Temperature Probe

ECE 445 Design Document
Team 34 - Joshua Rodriguez, Michael Pauls, and Yeongyoon Park
2/27/2020
TA: Charles Ross

1. Introduction

1.1 Problem and Solution Overview

Home appliances like dryer units are used several times a week by American consumers, and they serve a major function in one's everyday life. Yet, they are not always perfectly made, and improper use or lack of maintenance can cause malfunction. From 2010 to 2014, 15,970 reported home fires resulting from dryer fires accrued 13 deaths, 444 injuries, and $238 million in property damage. [1] With 27% of these fires originating from dust and lint igniting, this is a cause of concern. Excess lint from clothing and fabric softener from dryer sheets can accumulate in various areas of the dryer. This can then lead to air vents becoming clogged, and eventually, this excess lint may even cause overheating. When air temperatures exceed 250 degrees Fahrenheit, a fire hazard may occur since cotton and wool burn at those temperatures. Our goal is to accurately detect the current dryer air temperature and alert a homeowner or resident when the air temperature has exceeded the 250-degree threshold. By doing this, fires and property damage may be avoided.

A thermocouple could be used to measure the internal temperature of a dryer during its operation and a temperature above 250°F indicates a large accumulation of lint that must be cleaned out. However, such a solution is rather expensive with units costing about $100.A more cost-effective user-friendly dryer temperature probe can be created by utilizing three components. The temperature probe would be a k-type thermocouple that can withstand and measure temperatures of up to 350°F inside the dryer while it is running. The hardware unit would physically be placed on top of the dryer while it is running to read the data, convert it from analog to digital, and transmit the data via wireless communication. A smartphone would then be used to view the temperature data in real-time and indicate whether or not the temperature is potentially dangerous. This interface would be more user-friendly compared to the LCD display commonly found on handheld thermocouple temperature sensors.

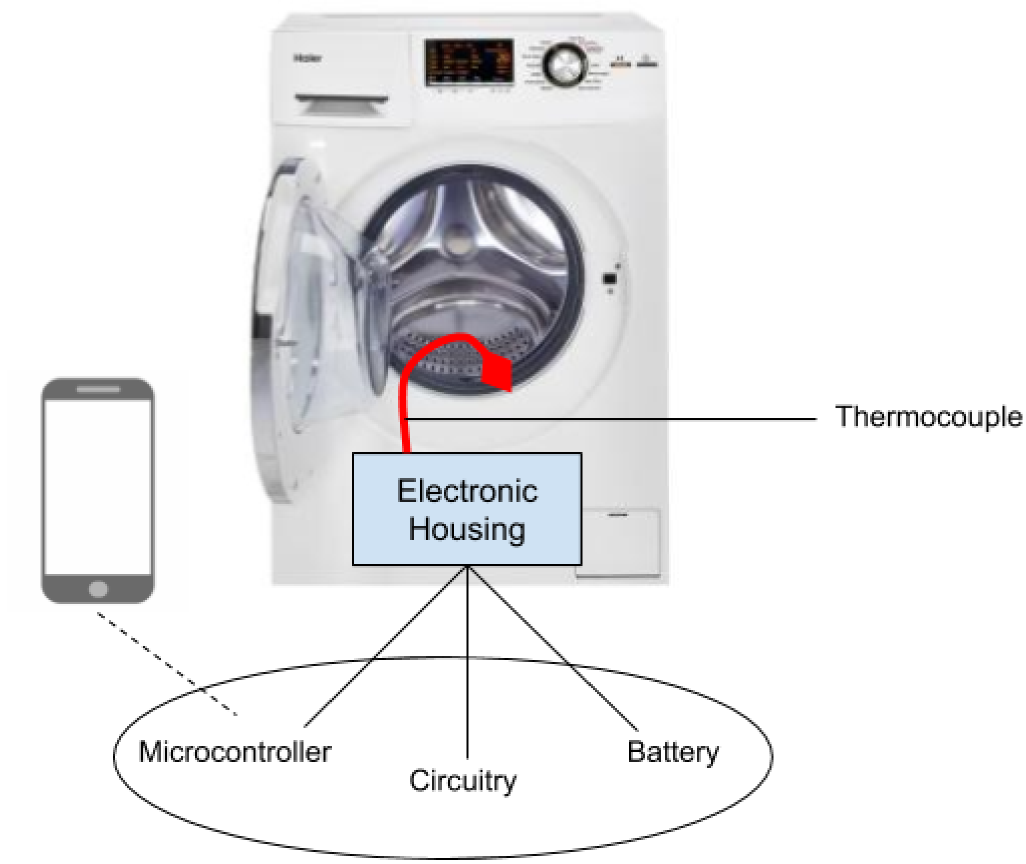
1.2 Background

Previous attempts have been made to come up with consumer-oriented solutions to dryer overheating. One existing cheaper alternative is thermal cut-off kit that is placed inside the dryer. The price range of the product is about $30 to $60, however it is difficult to have the product replaced whenever the kit malfunctions. It also does not notify the user of the current operating temperature of the dryer. Our solution will allow for easy battery and component replacement, and it will notify users of close to real-time temperatures through a smartphone application.

Other products exist that focus on the air pressure of a dryer's ventilation system to detect clogging. Yet they too can be rather expensive and don't detect the main cause of fire, rising temperatures [2]. There have also been previous solutions that are no longer on the market due to failed product launches like the “Dryer Detective” [3]. Although it is no longer available to consumers, it shows that there is market validation for a device that can alert homeowners of a potential fire hazard.

1.3 Visual Design

Our solution will be mounted on the front of a dryer unit in a protective box. It will have an opening for the thermocouple sensor to extend outside of the box and into the dryer’s lint trap through the dryer’s main door. The computational components will also connect wirelessly to a designated smartphone.



1.4 High-Level Requirements

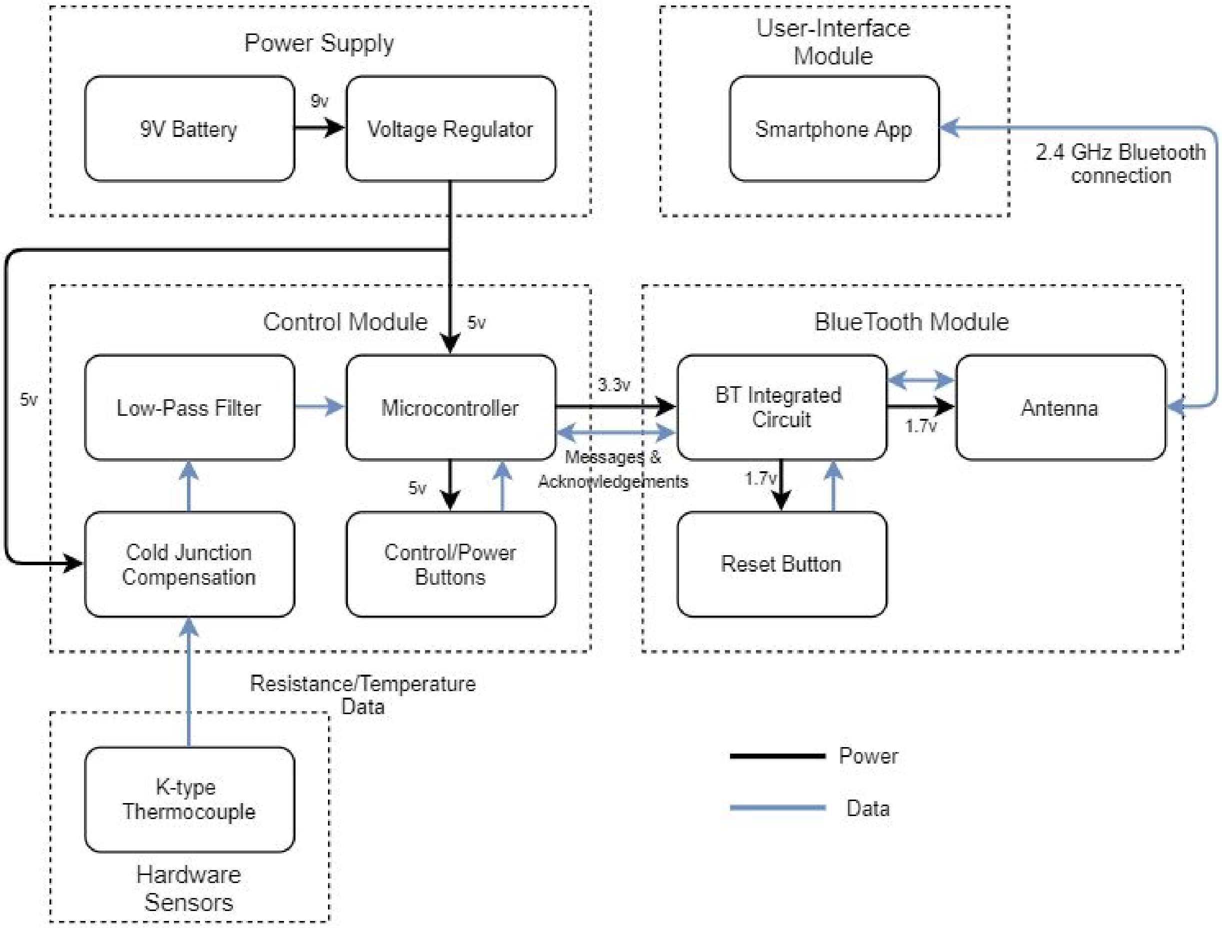
- The thermocouple must be able to accurately detect the current temperature in the lint trap within a tolerance of 2 degrees Fahrenheit. It should work within the temperature range of 32 degrees Fahrenheit to 350 degrees Fahrenheit.
- Our prototype must easily connect with a user’s smartphone via Bluetooth. It should provide a close to real-time (no more than a 2-minute delay) temperature reading.

- The smartphone app should display results and recommendations to the user in an easy-to-understand manner using temperature zones. It should indicate whether the dryer temperature is too low, just right, at a cautiously high temperature, or is at risk of starting a dryer fire. This will be accomplished by utilizing color-coded temperature zones. A blue color code will indicate temperatures less than 185 degrees fahrenheit which is nothing to be concerned about. A green color code indicates that operating temperatures are between 185 degrees and 210 degrees fahrenheit and are considered normal. A yellow color code indicates temperatures between 210 and 250 degrees fahrenheit, and a recommendation will be displayed telling the user to have maintenance on the dryer unit. Finally, a red color code will indicate temperatures above 250 degrees fahrenheit, and this will notify the user of unsafe operation.

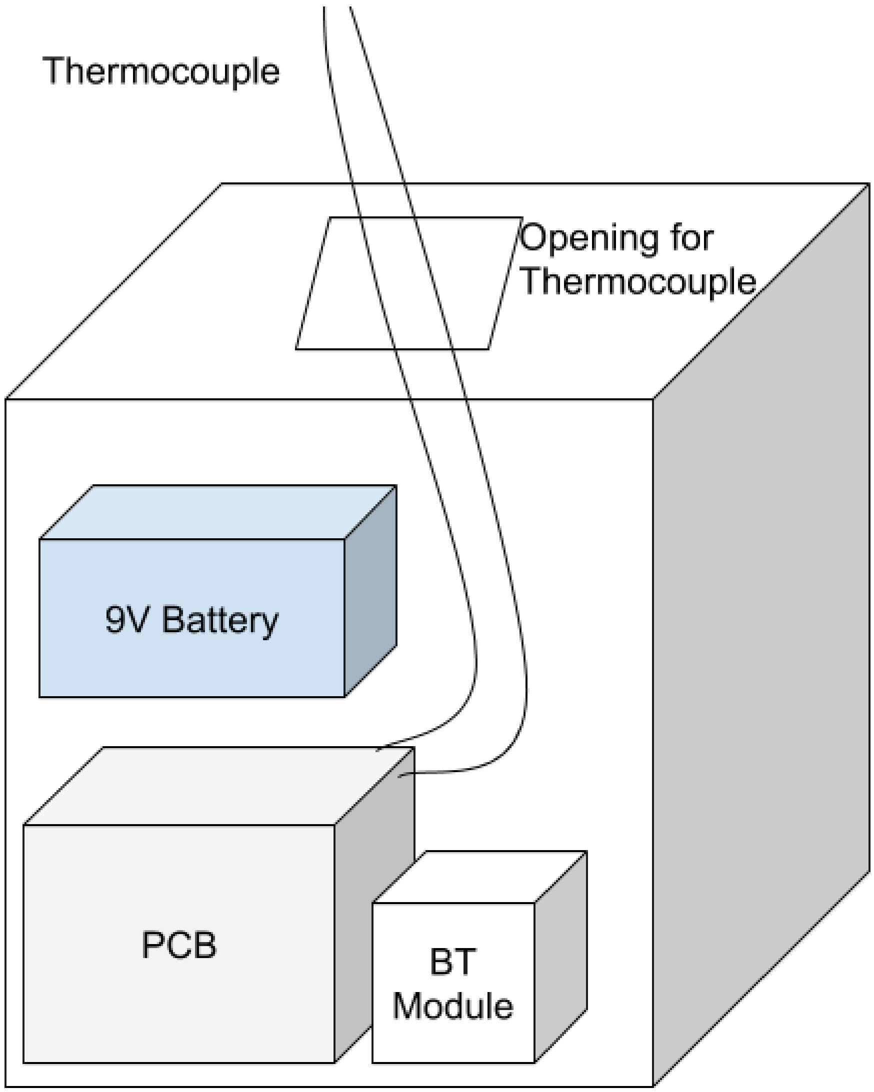
2. Design

2.1 Block Diagram

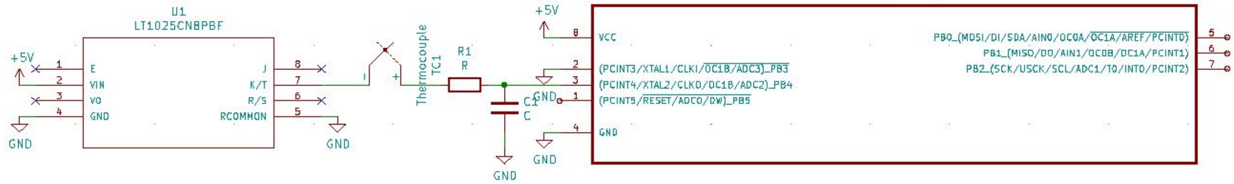
Our design requires several distinct modules: a power supply, control module, Bluetooth module, hardware sensor unit, and a user-interface module. The power supply allows for the entire unit to function whenever a dryer unit is on. The control module allocates power for Bluetooth data transmission and converts the thermocouple voltage data into human-readable temperature data. The Bluetooth module encodes data for transmission to a connected smartphone where data can be displayed. The hardware sensor unit ensures that the real-time temperature of the dryer lint trap is being read. Finally, the user-interface module allows data to be presented in an easy-to-understand format.


Power Supply
9V Battery
9v
Voltage Regulator
User-Interface Module
Smartphone App
2.4 GHz Bluetooth connection
Control Module
5v
Low-Pass Filter
Microcontroller
3.3v
BlueTooth Module
BT Integrated Circuit
Antenna
1.7v
Messages & Acknowledgements
5v
1.7v
Cold Junction Compensation
Control/Power Buttons
Reset Button
Resistance/Temperature Data
K-type Thermocouple
Hardware Sensors
Power
Data

2.2 Physical Design



2.3 Circuit Schematic



The above circuit diagram depicts the control module of the unit. The thermocouple (center, TC1) takes temperature data and produces a voltage that is biased by the cold-junction compensation IC (left, LT1025). The thermocouple signal is passed through a low-pass RC filter (center, R1 and C1) and the output is passed into the ADC of the microcontroller (right, ATtiny85, PB3, PB4). The microcontroller's output (PB1, PB2) is then passed to the Bluetooth module for data transmission.

2.4 Power Supply

The power supply will power the entire circuit and its components. It will also convert a 9V input to a 5V voltage in order to power the hardware components.

2.4.1 9V Battery

A disposable 9V alkaline battery will be used as a power source for the unit. Since consumers are already familiar with 9V batteries, this will allow for easy replacement and disposal of discharged batteries. A reverse polarity protection circuit made out of a diode is included to protect the circuit from incidental backwards insertion by the user.

Requirement	**Verification**
Reverse polarity insertion of 9V battery results in less than 1 mA of current entering device.	1. Connect a dummy load of 5kΩ to the end of the protection circuit. 2. Connect an ammeter between the battery and the start of the protection circuit. 3. Verify that the resulting measured current does not exceed the expected maximum designated current.

2.4.2 Voltage Regulator

The voltage regulator takes in the 9V input and outputs a constant 5V output. The 5V output voltage will also directly connect to the printed circuit board to power any hardware components like the microcontroller and cold-junction compensation IC.

Requirement	**Verification**
Voltage regulator outputs a 5V +/- 0.5V output with at least 12.08 mA.	1. Connect a DC power supply into the input of the voltage regulator.

	2. Connect a 400 Ω resistor to the output of the 5V output of the voltage regulator. 3. Connect an ammeter between the resistor and the voltage regulator output. Connect an oscilloscope probe to the output of the voltage regulator 4. Turn the DC power supply on. 5. Verify that the voltage stays within an acceptable range and that the output current meets the expected value.

2.5 Control Module

2.5.1 Microcontroller

The ATtiny85 microcontroller converts the conditioned analog reading from the thermocouple and controls the output that is sent to the Bluetooth module.

Requirement	**Verification**
The code is sent to the microcontroller successfully and runs.	1. Create a simple LED toggling code that constantly toggles output value on pin PB3 of the microcontroller every second 2. Compile the code with Arduino IDE 3. Sent the code using usb connection of the development board 4. Initiate the code 5. Check if LED toggles every second
There is no overflow in the microcontroller memory.	1. Carefully design the code to not exceed 8KB 2. Add constraints in the code to not exceed 512 bytes of internal memory 3. Create overflow test cases 4. Connect the development board and test the overflow by displaying the output value on the Arduino IDE
The ADC does not have bias.	1. Connect the thermocouple to the pin

	PB3 and PB4 2. Connect the output of the thermocouple to the voltage meter 3. Compare the voltage displayed on the ADC and the voltage meter 4. Recalibrate the microcontroller to reduce the bias
Gain of ADC shows correct value and does not have bias.	1. Connect the thermocouple to the pin PB3 and PB4 2. Connect the output of the thermocouple to the voltage meter 3. Set gain as x5 and compare the voltage displayed on the ADC and the voltage meter by manually multiplying the output of the voltage meter by 5 4. Recalibrate the gain to reduce the bias

2.5.2 Cold-Junction Compensation IC

The LT1025 cold-junction compensation IC biases the voltage of the cold junction of the thermocouple by detecting the surrounding temperature and adjusting the voltage appropriately. The thermocouple then produces a voltage output as if the cold junction were at 0 °C, resulting in accurate temperature detection. The IC has multiple outputs for different thermocouple types but in this case the negative terminal of the thermocouple is connected to the type T, K pin.

2.5.3 Low-Pass Filter

The RC low-pass filter aims to remove any noise coming from the thermocouple, mainly noise from the 60Hz power lines that are prevalent in

Requirements	**Verification**
RC low-pass filter decreases as a function of frequency by -20 db/decade starting at 60 Hz and ending at 960 Hz.	1. Connect function generator to input of filter and oscilloscope probe to output of filter 2. Generate a 2V peak-to-peak sinusoid on the output of the function generator at 30 Hz and record the resulting output voltage 3. Generate a 2V peak-to-peak sinusoid

	on the output of the function generator at 60 Hz and record the resulting output voltage 4. Generate a 2V peak-to-peak sinusoid on the output of the function generator at 120 Hz and record the resulting output voltage. 5. Generate a 2V peak-to-peak sinusoid on the output of the function generator at 240 Hz and record the resulting output voltage. 6. Generate a 2V peak-to-peak sinusoid on the output of the function generator at 480 Hz and record the resulting output voltage. 7. Generate a 2V peak-to-peak sinusoid on the output of the function generator at 960 Hz and record the resulting output voltage. 8. Output signal magnitude should roughly halve (1/1.8 to 1/2.2) for doubling of the input frequency.

2.5.3 Control & Power Buttons

This allows the user to control the power state of the unit, turning it off when not in use, and it allows us to reset the unit for testing and in the case of erroneous functionality.

2.6 Bluetooth Module

2.6.1 Bluetooth Integrated Circuit

The Bluetooth module takes the digital signal from the microcontroller and outputs it to the user's smartphone. It will have to process and manage pairing of devices such as nearby smartphones. It will be the primary network host for packaging information to send and to receive acknowledgments from the paired smartphone. An R&V table is provided below:

Requirement	**Verification**
The bluetooth Integrated Circuit (and module	Test the connection between microcontroller

as a whole) should successfully connect with the microcontroller, and pair with unique devices	and the bluetooth module by using the working nordic bluetooth app. After the functionality of microcontroller's capability of generating bluetooth protocol is verified, test pairing a smartphone with the bluetooth module With a smartphone, test searching for the bluetooth module. If successful, test whether the bluetooth module gets paired with the smartphone. Test if the connection lasts until bluetooth is disconnected manually. Connect and disconnect the bluetooth module constantly for 20 seconds.
The bluetooth module should successfully deliver messages to its paired device	Test sending a short text (ie. "Hello World") to its paired device. This should be correctly displayed via a smartphone application or other configuration on the paired smartphone. Test sending dummy temperature data in low frequency (1 message per 5 seconds). Test sending dummy temperature data in various frequencies, up to 1 message per second. Test if any message is lost after sending constant message for 30 minutes

2.6.2 Antenna

The antenna allows the Bluetooth module to communicate from a longer range which is helpful since most dryer units are placed either in closets or in the basement of users' residences, requiring better signal strength. This will serve as the generator of network waves to communicate with the smartphone. It follows the 2.4 GHz bandwidth standard for bluetooth communication.

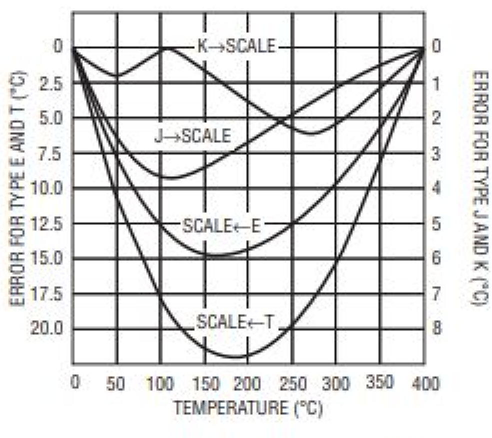
2.6.3 Reset Button

The reset button allows the user to reset the Bluetooth module in case of incorrect or unwanted operation. This will be primarily used in testing the individual module.

2.7 Hardware Sensor Unit

2.7.1 K-Type Thermocouple

This is a k-type thermocouple composed of chromel and alumel. It is rated at the dryer operating temperature range and produces a linear voltage that varies with temperature at about 40.6 μV/°C via the Seebeck effect . [4] The associated temperature is the temperature difference between the hot junction and the cold junction. This analog signal must be conditioned before being digitally converted. The expected error over the dryer operating range does not exceed 1 °C, making it pass the threshold for allowable temperature tolerance.



Non-linearity of thermocouple voltage and expected error curves [5]

Requirement	Verification
The thermocouple should generate a voltage within a tolerance of 1.3 degrees Celsiius (52.78 μV) with the actual measured temperature from the range of 32 degrees Fahrenheit to 350 degrees Fahrenheit.	1. Connect the thermocouple to the K, T output pin of the LT1025 cold junction compensation IC that is powered to a DC power supply of 5V V_{cc}. 2. Note the temperature of the room and measure the voltage generated by the thermocouple with a multimeter at room temperature. 3. Apply a controlled temperature by a heat gun to the hot junction of the thermocouple and measure the voltage generated with a multimeter. 4. Apply a controlled temperature by a soldering iron to the hot junction of the thermocouple and measure the voltage generated with a multimeter.

	5. The maximum deviation from the expected voltage in any test should be no greater than 52.78 μV.

2.8 User-Interface Module

2.8.1 Smartphone Application

The smartphone application alerts the user when the dryer temperature reaches above 250 °F. It also detects the maximum temperature detected and certain ranges indicate different operating problems.

Requirement	**Verification**
Bluetooth manager of the android app works correctly	1. Connect a smartphone to android studio and start debug simulation 2. Log all data produced from the bluetooth low energy manager object 3. Pair with the bluetooth module 4. Check the log to check if the bluetooth manager works properly
Application generates color code indicating each of four designated temperature levels	1. At the debugging phase of the app, write test cases with dummy temperature integer value ranging from 150 to 300 2. Check if the simulation outputs blue for value less than 185 3. Check if the simulation outputs green for value between 185 and 210 4. Check if the simulation outputs yellow for value between 210 and 250 5. Check if the simulation outputs red for value above 250 6. Connect a smartphone to android studio and start debug simulation 7. Log all data produced from the bluetooth low energy manager object of the app 8. Pair with the bluetooth module that sends integer value from 150 to 300 9. Check if the application outputs blue

	for value less than 185 10. Check if the application outputs green for value application 185 and 210 11. Check if the application outputs yellow for value between 210 and 250 12. Check if the application outputs red for value above 250

2.9 Physical Case

The physical casing surrounding all of our hardware components will have to be robust enough to withstand shaking from a dryer unit. This case will be 3D printed using on-campus resources. This will drastically reduce the cost of prototyping such a case, and it will allow for easy testing to be done. The case may have specially designed plastic inlets to prevent hardware components from moving around excessively. This will help mitigate the effects of dryer shaking or other movement which may be imposed on the case. It will also have 4 small magnets glued onto the case in order to be mounted on the front or side of the dryer. Since dryers are relatively well insulated, heat will not be too much of a concern, but any excess heat will be reflected or absorbed by the thick outer plastic layer.

2.10 Tolerance Analysis

The most important part of this project is the accuracy of the temperature reading. Possible sources of deviation from the expected value come mainly from two sources: the not-perfectly-linear temperature vs. voltage curve of the thermocouple and the resolution of the ADC of the microcontroller. The physical behavior of the thermocouple cannot be feasibly controlled, so the most controllable aspect of the design is the resolution of the ADC. The ATtiny85 microcontroller contains a 10-bit ADC which should result in a resolution of 1.0742 mV assuming a 1.1V internal reference voltage. [6]

$$2^{10} = 1024$$

$$\frac{1.1V}{1024} = 1.0742\ mV$$

This translates to a temperature resolution of 26.458 °C, a figure that is unacceptable given our stated tolerance of +/- 2 °C.

$$\frac{1074.2\ \mu V}{40.6\ \mu V/°C} = 26.458$$

However, the ADC on the ATtiny85 has an option of a gain of x20, resulting in an increased resolution, resulting in a temperature resolution of 1.3229 °C, a figure that is acceptable with our tolerance of +/-2 °C

$$\frac{26.458}{20} = 1.3229$$

In short, an accurate reading of the thermocouple voltage is possible given the ADC is set to x20 gain and the reference voltage is set to 1.1V. We will likely include a 8x gain amplifier to further amplify this temperature resolution to be within 2 degrees fahrenheit.

3. Cost and Schedule

3.1 Cost Analysis

Our development costs assume a developer salary close to $95,000 which is the ECE Illinois average for computer engineers right after graduation. This roughly translates to $50/hour. Our prototype will take roughly 10 weeks to complete with three people working for 10 hours each week. The total labor cost to develop this prototype comes out to:

$$\$50/hr \ * \ 2.5 * 10 \ hr/week \ * \ 3 \ people \ * \ 10 \ weeks \ = \$37,500$$

The component costs for the prototype are depicted below:

Component	**Cost (Prototype)**	**Cost (bulk manufacturing)**
9V Battery (Amazon, Energizer)	$8.98	$1
PCB (PCBWay)*	$23	$1.50
ATTINY85-20PU (microcontroller)**	$5.33	$1.20
UA78L05ACDR (voltage regulator)	$0.49	$0.10
1N5819-E3/54 (diode)	$0.46	$0.08
Assorted resistors and capacitors	$1.00	$0.20
Bluetooth Module (Mouser: nRF52833-DK)**	$46	$1.50
K-Type Thermocouple	$9.99	$1.25
Casing***	$1	$0.25

*PCB prototype cost including shipping cost 18$, minimum order quantity of 5
**Microcontroller and Bluetooth Module prototype cost include development kit
***Casing Prototype 3D printed

Our total prototype costs come out to $96.25. This is higher than expected particularly because of the purchasing platforms which we were required to use (Mouser, Digikey). The microcontroller module and bluetooth module could only be purchased individually with a development kit which exacerbated costs. If this product were to be made in bulk, say 1,000 units, the unit's cost would be $7.08. This could also be reduced further to $6.08 if batteries are not included in the packaging.

3.2 Schedule

Week	Mike	Josh	Yoon
2/17/2020	Research and order components, work on first draft of design document	Research and order components, work on first draft of design document	Research and order components, work on first draft of design document
2/24/2020	Finalize design criteria with Greg, work on design document	Design PCB, work on design document	Work on design document, Purchase BT module
3/2/2020	Re-adjust work plan if needed, confirm project scope with Greg, begin testing thermocouple	Finish PCB footprints and design. Begin testing thermocouple	Begin testing microcontroller, BT module separately
3/9/2020	Start testing thermocouple components with microcontroller	Submit PCB order, test thermocouple components with microcontroller	Begin integrating microcontroller with BT module
3/16/2020	Spring Break		
3/23/2020	Integrate and test BT module with the microcontroller	Integrate and test BT module with the microcontroller	Begin App design
3/30/2020	Begin integrating Thermocouple components with microcontroller/BT module	Begin battery/power supply testing	Finish app design and implementation
4/6/2020	Test sending data to smartphone, send first temperature data to smartphone	Implement Power supply into unit	Test sending data to smartphone, send first temperature data to smartphone
4/13/2020	Integrate all components, design physical casing	Perform system tests on completed unit	Help with system integration and testing
4/20/2020	Print physical casing,	Exhaustively test	Exhaustively test

	aid in testing	prototype, Extra work*	prototype, Extra work*
4/27/2020	Prepare report, demo, and complete extra work*	Prepare report, demo, and complete extra work*	Prepare report, demo, and complete extra work*
5/4/2020	Present	Present	Present

* Extra work denotes work to be done on researching and reporting on how to integrate rechargeable batteries and work to expand the design of the smartphone application.

4. Discussion of Ethics and Safety

There are several safety and ethical concerns regarding the operation of this device. We must ensure that the temperature reading is accurate since the IEEE code of ethics states “to avoid injuring others, their property, reputation, or employment by false or malicious action.” [7] Displaying false information could result in bodily or property damage. A false low temperature could drop a person’s guard, letting them touch a dangerously hot thermocouple without letting it cool down. A false low temperature could also persuade them not to have their dryer properly maintenanced, resulting in a higher probability of dryer fires, a situation that the aim of this device is attempting to reduce the likelihood of. The thermocouple could reach temperatures exceeding 120 °C. A partial skin thickness burn can result from one second of contact with a 70 °C surface [8]. Proper testing of the smartphone application and microcontroller logic will help to ensure that only accurate temperature readings are being displayed to the user.

Our circuitry and PCB may cause excess heating if not designed properly. This could result in a potential small fire within the device itself. In order to avoid this, we must rigorously design and test our circuit components before and after integrating them into the entire system. Furthermore, dryer units often shake which could cause some components to move out of place if not secured properly. To mitigate this, we should ensure that our casing tightly and securely keeps all components in place. Some dryers also exude heat which can potentially cause melting of the casing or components. Preventing this will require a higher quality plastic for the casing as well as rigorous testing.

There is a risk that the temperature unit may be used incorrectly, either placing the electronics housing in the dryer itself instead of keeping it outside of the dryer during operation or utilizing any part of the circuit with a washing machine. This will at the very least cause damage to the thermocouple sensor, the physical circuitry, and possibly the laundry machine itself. Warnings on the packaging will be essential to avoid this risk. For all problems that may arise due to user misuse, warnings would be apparent on the packaging to whoever uses the device.

5. Citations

[1] R. Campbell, “Home fires involving clothes dryers and washing machines.” NFPA.org. https://www.nfpa.org/News-and-Research/Data-research-and-tools/US-Fire-Problem/Home-fires-involving-clothes-dryers-and-washing-machines (accessed Feb. 23, 2020).

[2] “Protect Your Home And Family With LintAlert.” *LintAlert*, lintalert.com/?home=1. (accessed Feb. 26, 2020)

[3] *The Dryer Detective*, web.archive.org/web/20160112223247/dryerdetective.com/. (accessed Feb. 26, 2020)

[4] Linear Technology, “Micropower Thermocouple Cold Junction Compensator,” LT1025 datasheet.

[5] Linear Technology, Appl. Note 28, p. 3.

[6] Atmel, “Atmel 8-bit AVR Microcontroller with 2/4/8K Bytes In-System Programmable Flash,” ATtiny85 datasheet, [Revised Aug. 2013].

[7] “IEEE Code of Ethics.” IEEE.org. https://www.ieee.org/about/corporate/governance/p7-8.html (accessed Feb. 13, 2020).

[8] J. C. Lawrence, J.P. Bull, “Thermal conditions which cause skin burns,” *Eng. in Med,* vol. 5, no. 3, pp. 61-63, Jul. 1976.