Title: Design process of room temperature monitor

Jessica Qiu U98988251

1. Summary

The room temperature monitor design project aimed to create a portable, battery-powered device that accurately measures and displays the room temperature. The goal set for us was to make a room temperature beep and flash the red LED, which warns people when the temperature passes a certain range. This project utilizes electronic components such as Arduino board, temperature sensor, LCD screen, and LEDS. The design process involved designing the layout in TinkerCAD, programming the Arduino to read and display temperature values in Arduino IDE, and soldering the appropriate components together. With a specified range in the Arduino code, the final prototype provides an accurate temperature reading. The key outcomes of the project include the successful soldering and integration of components, the development of a functional prototype, and its usability in the real world. Future enhancements and improvements could also reduce the prototypes' size and weight.

2. Introduction

Temperature monitors are useful in various applications. In places like the Northeast of the United States where the temperature varies, many homes and workplaces have been renovated to have temperature monitors. Because the temperature monitors are versatile, it's interesting to study and breakdown its various components in this paper. Oftentimes, it has large applications such as houses and workplaces. However, sometimes, it is also implemented in smaller applications such as phones, tools, and other life-saving equipment. One of the reasons why the temperature monitors are so important in various applications is because of climate change, where humans have to adapt to. For example, if it's too cold or hot outside, we change the temperature indoors; by measuring our own body temperature, we can assess whether we are ill or not. This shows that knowing the temperature is crucial in our lives and it can benefit or even save our lives.

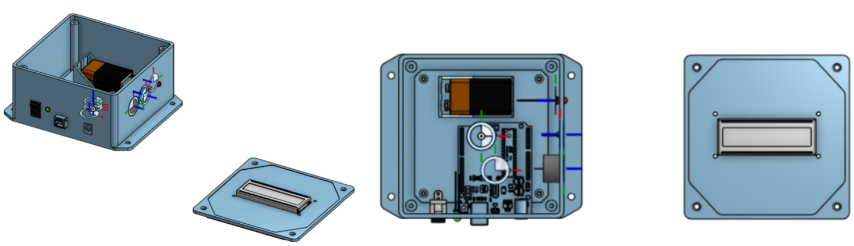
3. Design elements

a. Hardware: Black ABS enclosure box, box's lid, mounting plate, tether pins, rocket spades, twist nut caps, screws, bolts, 3D printed battery holder

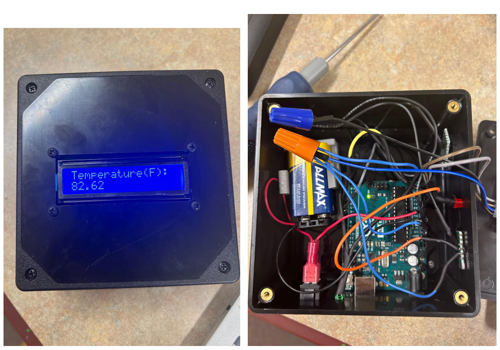
Electrical: switch, power jack, USB port, green LED, red LED, TMP36, Arduino uno board, LCD screen 2x16, 9V battery, buzzer, jumper wires, 22 AWG wires

b.

Item	Sketch Identification	W [mm]	L [mm]	H [mm]	Diameter [mm]
ABS enclosure	Part 5: (1)	118	143	70	N/A
Arduino Board	Part 5: (2)	69	54	13	N/A
Switch	Part 5: (3)	25	11	21	N/A
LCD 2x16	Part 5: (4)	35	80	20	N/A
Buzzer	Part 5: (5)	N/A	N/A	19	21
Temperature Sensor	Part 5: (6)	N/A	N/A	18.5	4.5
LED	Part 5: (7)	N/A	N/A	37	5



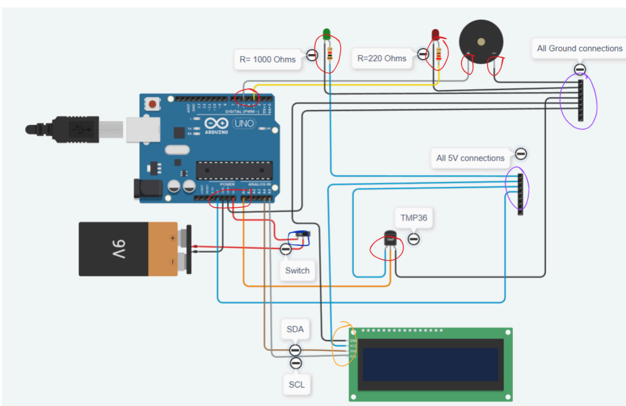
c.



d.

e. In general, Arduino boards read inputs and turn them into an output. In the case of our temperature sensor monitor, the Arduino board interfaces with the TMP36 sensor to read the temperature. The sensor gives an analog voltage output proportional to the temperature and the Arduino converts that to a digital value in order for it to process. Essentially, once the temperature is read, the Arduino processes the data to ensure accuracy. The temperature is displayed on the LCD screen which is controlled by the Arduino board. It sends the temperature data to the LCD screen and updates it from time to time. In short, the Arduino board acts like a brain of the room temperature monitor, giving data to other components where they can interact with one another to display the temperature.

f.



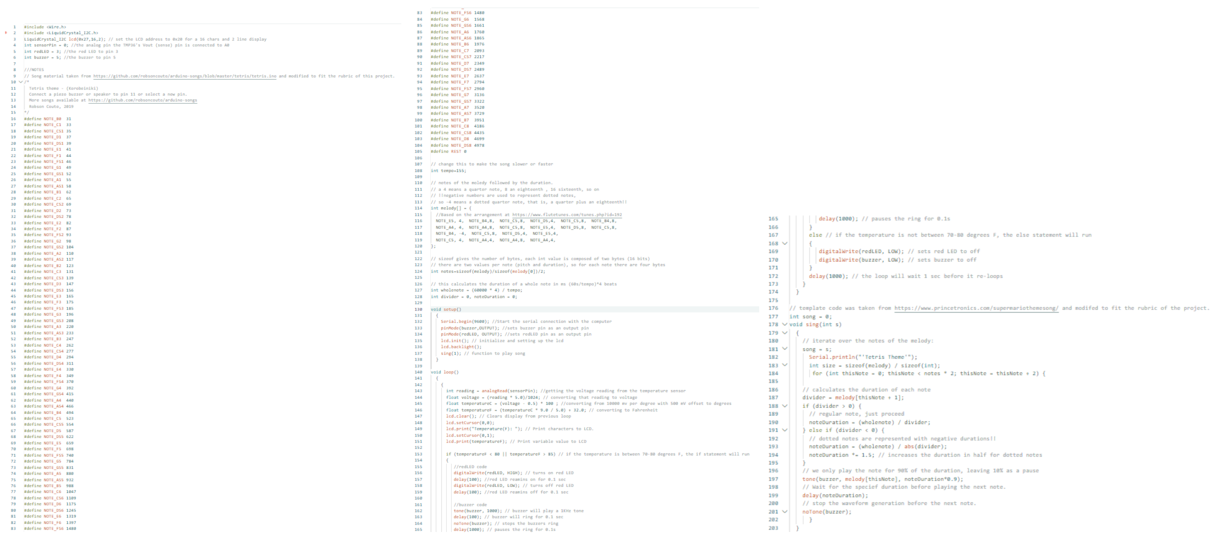
In the left diagram, the red circles represent that the components and solid wires should be soldered and shrink tubed. The red circles on the Arduino board represent the solid wires soldered to the tether pins which would connect to the Arduino board pins. The purple circles represent the solid wires that should be connected by twist nut caps as there are only a few ground connections to the Arduino board. The orange circle represents the jumper wires connected to the LCD screen as it is hard to solder wires to the small pins in the LCD screen as they're close together.

g. For low-power circuits like this prototype, 22 AWG wires are used as it provides a good balance between current carrying capacity and flexibility. Jumper wires are often used for temporary connections in a circuit, but 22 AWG solid wires are used for permanent connections as it won't break as often compared to the jumper wires and it provides better conductivity. A 1k ohm resistor is used for the green LED and a 220 ohm resistor is used for a red LED. These values are chosen to prevent damage to LEDs and control their brightness. To find the operating current, we can use KVL and V/R = I. The green LED's voltage is 2.2 V and so 2.2V/1000 ohm

= 2.2 mA, which is the operating current for the green LED. The red LED’s voltage is 2.0 V and so 2.0V/220 ohm = 9.09 mA, which is the operating current for the red LED.

h. We used a 9V battery because the Vin port of the Arduino board accepts voltage between 6V and 20V, where 9V is in that range. Because all the VCC pins are connected to 5V Arduino pins, the circuit’s operational voltage is 5V. Our 9V battery has a capacity of 600 mAh. It’s a good option as it fulfills voltage requirements and it’s small and lightweight to use. Assuming that 100 mA of current runs the circuit, the runtime will be 600/100 = 6 hours. Due to the way the Arduino board is built, I can power the device externally since I can connect an external power supply like a wall adapter to power the device instead of using an internal battery. This can be useful for applications where continuous power is needed.

i.

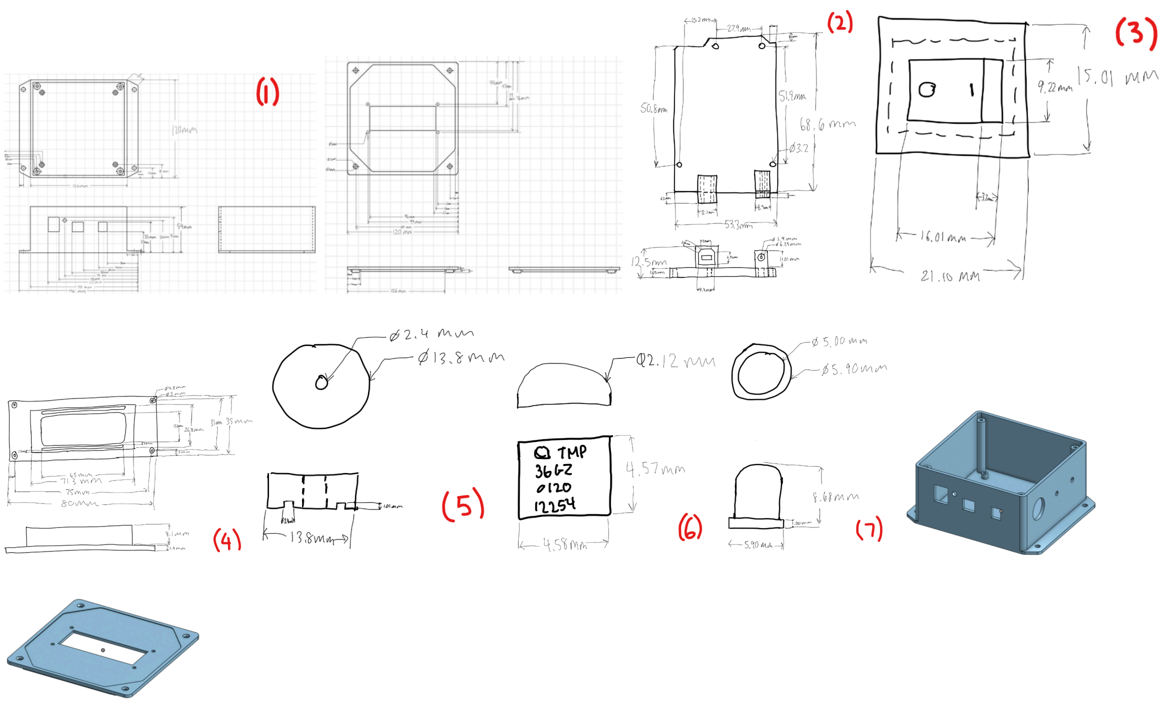


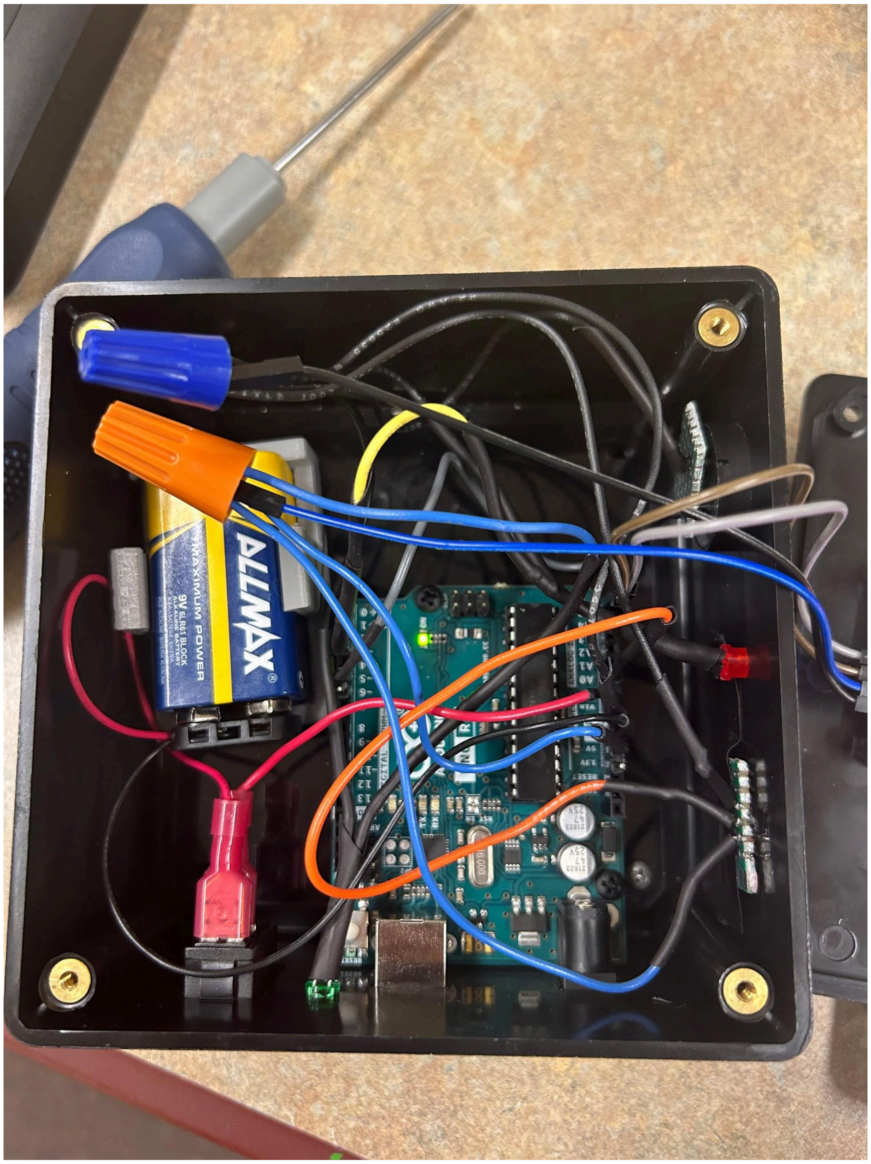
j. The voltage of the power supply is 9V from the battery. The operating voltage of the circuit is 5V which is regulated by the Arduino Board. The 1K Ohm resistor is used to limit the current through the green LED to a safe level, preventing it from burning out due to excess current. The total current drawn is 56 mA. The TMP36 temperature range is -40 degrees C to 125 degrees C. The range I am comfortable with is 20 degrees C to 25 degrees C. When the circuit is powered, voltage drop across the resistor (Vr) is calculated using Ohm's Law: Vr = I * R, where I is the current and R is the resistance. The 220 Ohm resistor is used to limit the current through the Red LED, ensuring that the current through the Red LED is at a level that produces a desirable brightness without exceeding the LED's maximum current. Based on KVL, changing the resistance affects the LED’s brightness, where voltage across the resistor (volts) equals to current through the resistors (amperes). Because a green LED has a higher voltage than a red LED, it needs to have higher resistance to drop the excess voltage from the power supply.

4. Evaluation of Results

After assembling the necessary components, the temperature monitor works well as it accurately measures and displays the temperature, given a specific temperature range. The circuit of the Arduino board reads the temperature using the TMP36 sensor, processes the data, and displays it on the LCD screen. Compared to standard thermometers, the room temperature has several advantages which include real-time temperature display and is easy to use. It also offers additional features such as digital display. The current design has a limited battery life, especially if this prototype is used continuously. The prototype also can be a bit heavy and large. Future work could focus on optimizing the battery life and have components that are lightweight. In short, the room temperature monitor provides a convenient and reliable way to read and display temperature data, which can be used in homes and workplaces optimally for users.

5. Supporting Materials




ALLMAX
MAXIMUM POWER