**Aim:**

The main aim of the project is to design and develop a fire alarm using temperature sensor.

**Requirement Analysis:**

Arduino UNO

Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins.

Temperature sensor

The Grove - Temperature Sensor uses a Thermistor to detect the ambient temperature. It's this characteristic that we use to calculate the ambient temperature. ... The detectable range of this sensor is -40 - 125ºC, and the accuracy is ±1.5ºC.

Jumper wires

A jump is an electrical wire, or group of them in a cable, with a connector or pin at each end, which is normally used to interconnect the components of a breadboard or other prototype or test.

Grove Display kit

The LCD (Liquid Crystal Display) is a type of display that uses the liquid crystals for its operation. Here, we will accept the serial input from the computer and upload the sketch to the Arduino. The characters will be displayed on the LCD.

Buzzer

An arduino buzzer is also called a piezo buzzer. It is basically a tiny speaker that you can connect directly to an Arduino. You can make it sound a tone at a frequency you set. The buzzer produces sound based on reverse of the piezoelectric effect.

Bread board

A breadboard, or protoboard, is a construction base for prototyping of electronics. Originally the word referred to a literal bread board, a polished piece of wood used when slicing bread. In the 1970s the solderless breadboard became available and nowadays the term "breadboard" is commonly used to refer to these

Arduino to USB cable

A standard USB 1.1/2.0 Type B printer cable can be used for the Arduino Uno.

Arduino IDE software

The Arduino Integrated Development Environment (IDE) is a cross-platform application written in C and C++ functions for Windows, macOS, and Linux. It's used to write and upload programmes to Arduino-compatible boards, as well as other vendor development boards with the support of third-party cores.

PC/Laptop

We have used laptop to code the logics where we have installed Arduino software.

**Objective:**

1. Designing of the digital circuit

The first objective is to design a digital circuit (simulation) using Tinker cad.

1. Programming the controller

C programming language is used to program. Then the code should be compiled and uploaded using Arduino IDE software to the Arduino UNO board.

1. designing the physical circuit:

The controller, sensor, and other components should be connected appropriately with proper power source.

Pseudocode:

START

SET value to 0

WHILE

IF value > 300 DO

LCD print

BUZZER beep

END IF

LCD print value

ELSE IF value < 300 DO

LCD print

BUZZER stop beep

END IF

LCD print value

END WHILE

END

Pseudocode Explanation:

The initial value of the temperature sensor is set to zero. If the value of the temperature sensor is higher than 300 then the buzzer beeps and the Liquid crystal display (LCD) displays the alert message and the value of the temperature sensor will be printed.

If in case the value is below 300 then the buzzer doesn’t beep and the LCD will print as Feels good along with the value of the temperature sensor.

State machine diagram:

temperature 300 or below

temperature 300 or above

BUZZER LOW

BUZZER HIGH

START

LOOP

END

*State Information Table:*

|  |  |
| --- | --- |
| STATE NAME | SPECIFICATIONS |
| LOW | The buzzer doesn’t beep. (Figure 1) |
| HIGH | The buzzer beeps. (Figure 2) |

A picture containing icon

Description automatically generated

Figure 1: Buzzer LOW

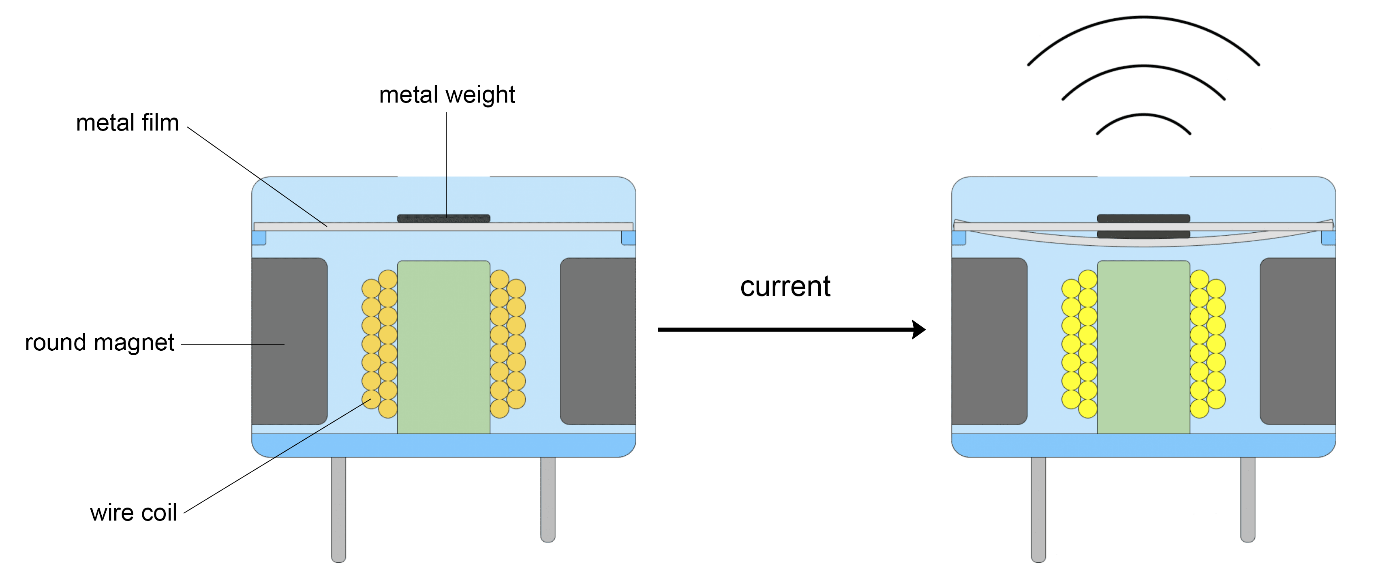


Figure 2: Buzzer HIGH

*Transition Information Table:*

|  |  |
| --- | --- |
| TRANSITION NAME | SPECIFICATIONS |
| buzzer LOW | Temperature sensor value 300 or below. |
| buzzer HIGH | Temperature sensor value 300 or above. |

Code:

#include <Wire.h>

#include "rgb\_lcd.h"

rgb\_lcd lcd;

const int colorR = 255;

const int colorG = 0;

const int colorB = 0;

int sensor = A0;

int buzzer = 2;

int value;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

pinMode(sensor, INPUT);

pinMode(touch, INPUT);

pinMode(2, OUTPUT);

lcd.begin(16, 2);

lcd.setRGB(colorR, colorG, colorB);

lcd.print("Current Temp...!");

delay(500);

}

void loop() {

// put your main code here, to run repeatedly:

value = analogRead(7);

if (value > 300){

lcd.setCursor(1, 1);

lcd.print("Fire Alert:");

digitalWrite(2, HIGH);

Serial.println(value);

}

lcd.setCursor(12, 1);

lcd.print(value);

delay(500);

if (value < 300){

lcd.setRGB(0, 0, 255);

lcd.setCursor(1, 1);

lcd.print("Feels Good:");

digitalWrite(2, LOW);

Serial.println(value);

}

lcd.setCursor(12, 1);

lcd.print(value);

delay(500);

}

Output:

A picture containing text, indoor

Description automatically generated

Figure 3: Temperature below 300

A picture containing text

Description automatically generated

Figure 4: Temperature above 300

Video clip link:

<https://drive.google.com/file/d/1PX8bwS_8rHCROmEf1HpzbcZxf55rIkDc/view?usp=sharing>

Test plan:

The temperature sensor is connected to the Analog pin 0 and buzzer is connected to the 2nd pin of Arduino UNO. the sensor will continuously reads the sensor data and if the sensor value is greater than 300 then the display module prints the message called “Fire alert” along with the temperature sensor value.

Reflective discussion: