

IBM ASSIGNMENT- I

TEAM ID : PNT2022TMID26661

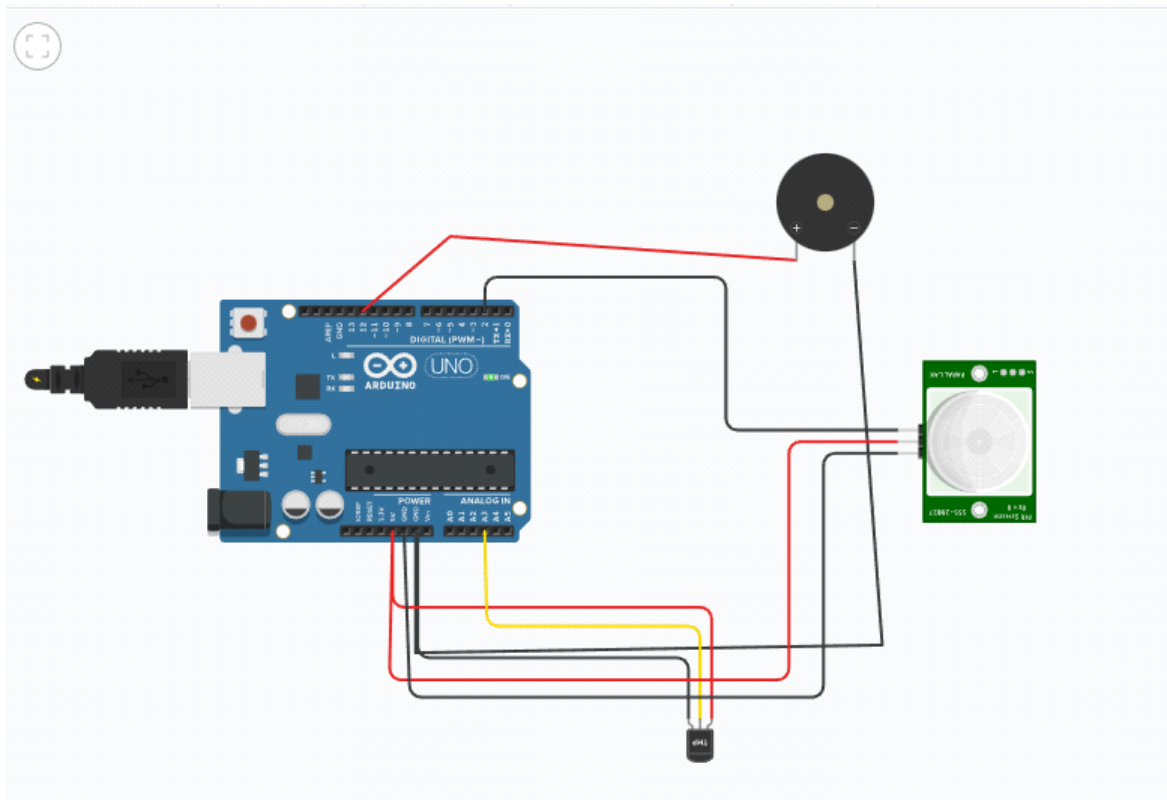
NAME : PAUL.V

ROLL NO: 212919106062

Create a circuit with Piezo alarm, PIR Sensor, TMP Sensor with below functionalities:

- 1.Alarm should sound in one manner if temp is above 60C
- 2.Alarm should sound with another frequency if motion is detected in PIR Sensor.

CIRCUIT LAYOUT:



CODE:

```
void setup()
{
  Serial.begin (9600);
  pinMode(2, INPUT);
  pinMode(12,OUTPUT);
  pinMode(A3,INPUT);
}

void loop()
{
  int motion = digitalRead(2);
  if (motion==1){
    Serial.println("Motion detected");
    tone (12,20000);
    delay(500);
  }
  else{
    Serial.println("No Motion");
    noTone(12);
  }
}

{
  double data=analogRead(A3);
  double n=data/1024;
  double volt=n*5;
  double off=volt-0.5;
  double temperature=off*100;
```

```

if (temperature>59.99){
  Serial.print("Temperature data:");
  Serial.println(temperature);
  tone(12,10000);
  delay(500);
}
else
  Serial.print("Temperature data: ");
  Serial.println(temperature);
  noTone(12);
}
}

```

OUTPUT:

PASSIVE INFRARED SENSOR:

The screenshot displays the Tinkercad web interface for an Arduino Uno R3 project. The workspace shows an Arduino Uno R3 connected to a Temperature Sensor (TMP36) and a Passive Infrared (PIR) sensor. The code editor contains the following program:

```

1 void setup()
2 {
3   Serial.begin (9600);
4   pinMode (2, INPUT);
5   pinMode (12, OUTPUT);
6   pinMode (A3, INPUT);
7 }
8
9 void loop ()
10 {
11   int motion = digitalRead(2);
12   if (motion==1){
13     Serial.println("Motion detected");
14     tone (12,20000);
15     delay(500);
16   }
17   else{
18     Serial.println("No Motion");
19   }
20   int temperature = analogRead(A0);
21   float voltage = voltage divider(temperature);
22   Serial.print("Temperature data: ");
23   Serial.println(voltage);
24   delay(1000);
25 }

```

The Serial Monitor shows the output of the program:

```

temperature data: 24.71
No Motion
Temperature data: 97.95
97.95
No Motion
Temperature data: 97.95

```

A "How the debugger works" tooltip is visible, providing instructions on how to use the debugger.

TEMPERTURE SENSOR (60 C):

Editing Components

Editing Components

4 / 4

Copy

Paste

Undo

Redo

Comment

Help

Run

Stop

Saved

Simulator time: 00:00:00

Code

Stop Simulation

Send To

Congratulations!

Your traffic lights should be blinking!

Continue Tinkering

- You can also change the color of your wires. Try selecting different components in this design and see what properties you can change.
- What happens to the brightness of the LEDs when you edit the resistance of the resistor its attached to? The resistor helps reduce current in the circuit so that the LED does not exceed its maximum rated current.
- In later tutorials, you'll learn how to program an Arduino to make your designs interactive. If you want to take a peek at

Reset

Done

PIR Sensor

Name: 1

Target X: -5.44

Target Y: -135.50

Target Y: -199.40

void setup()

{

Serial.begin (9600);

pinMode (2, INPUT);

pinMode (12, OUTPUT);

pinMode (A3, INPUT);

}

void loop()

{

int motion = digitalRead(2);

if (motion==1){

Serial.println("Motion detected");

tone (12,20000);

delay(500);

}

else{

}

}

Serial Monitor

temperature data: 24.71

Motion detected

Temperature data: 24.71

Motion detected

Temperature data: 24.71

Motion detected

Send

Clear

How the debugger works

1. Add breakpoints by clicking on the line numbers.

2. Hover over the variables while paused to see their value.

3. Use the buttons above to resume simulation or step one line at a time.