SmartBridge Externship

Internet of Things

Name: Binit Nayak

Reg. No.: 20BCE7420

Date: 20-05-2023

Assignment 1: In Wowki, if the distance is less than 100cms for an ultrasonic sensor, glow a LED

Code:

- **Sketch.io**

```
#define ECHO_PIN 2
#define TRIG_PIN 3

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
}

float readDistanceCM() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  int duration = pulseIn(ECHO_PIN, HIGH);
  return duration * 0.034 / 2;
}

void loop() {
  // put your main code here, to run repeatedly:
  float distance = readDistanceCM();

  bool isNearby = distance < 100;
  digitalWrite(LED_BUILTIN, isNearby);

  Serial.print("Measured distance: ");
  Serial.println(readDistanceCM());
```

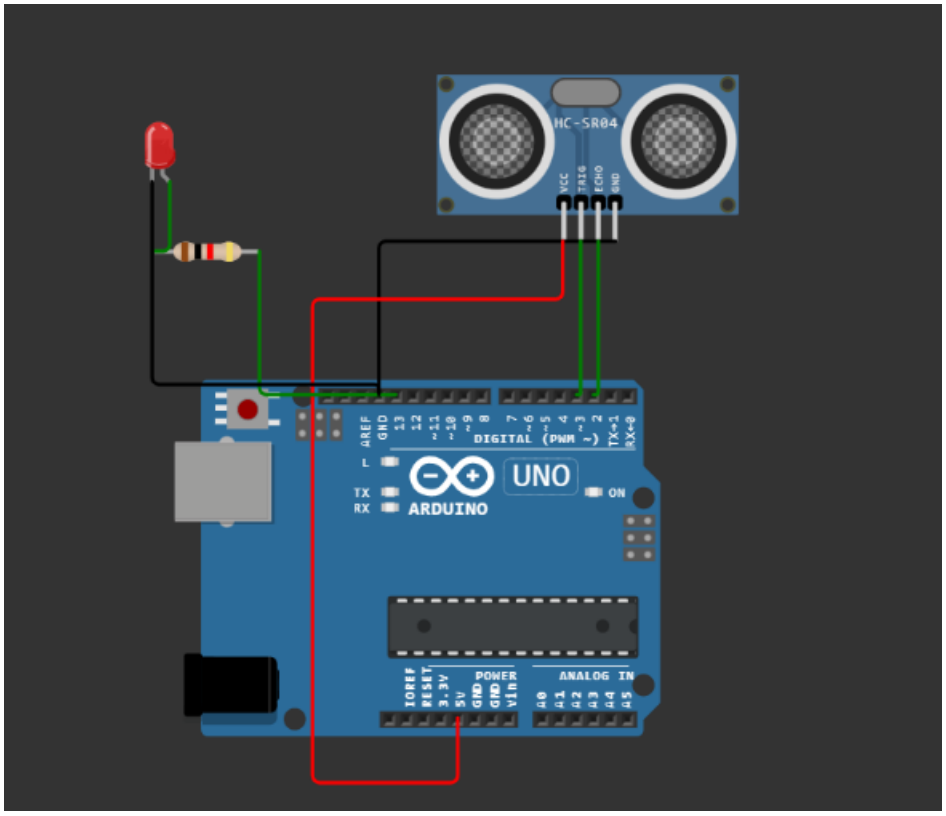
```
  delay(100);

}
```

- **Diagram.json**

```
{
  "version": 1,
  "author": "BINIT NAYAK 20BCE7420",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-arduino-uno", "id": "uno", "top": 0, "left": 0,
"attrs": {} },
    { "type": "wokwi-hc-sr04", "id": "ultrasonic1", "top": -172.14,
"left": 148.06, "attrs": {} },
    {
      "type": "wokwi-led",
      "id": "led1",
      "top": -153.29,
      "left": -27.64,
      "attrs": { "color": "red" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r1",
      "top": -77.63,
      "left": -10.85,
      "attrs": { "value": "1000" }
    }
  ],
  "connections": [
    [ "uno:GND.1", "ultrasonic1:GND", "black", [ "v0" ] ],
    [ "uno:5V", "ultrasonic1:VCC", "red", [ "v36.41", "h-82.02", "v-
272.82", "h140.21" ] ],
    [ "ultrasonic1:ECHO", "uno:2", "green", [ "v0" ] ],
    [ "ultrasonic1:TRIG", "uno:3", "green", [ "v0" ] ],
    [ "led1:A", "r1:1", "green", [ "v0" ] ],
    [ "r1:2", "uno:13", "green", [ "v0" ] ],
    [ "uno:GND.1", "led1:C", "black", [ "v-5.76", "h-144.78" ] ]
  ],
  "dependencies": {}
}
```

Diagram:



Output:

- **Distance less than 100cm**

```
#define ECHO_PIN 2
#define TRIG_PIN 3

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
}

float readDistanceCM() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  int duration = pulseIn(ECHO_PIN, HIGH);
  return duration * 0.008 / 2;
}

void loop() {
  // put your main code here, to run repeatedly:
```

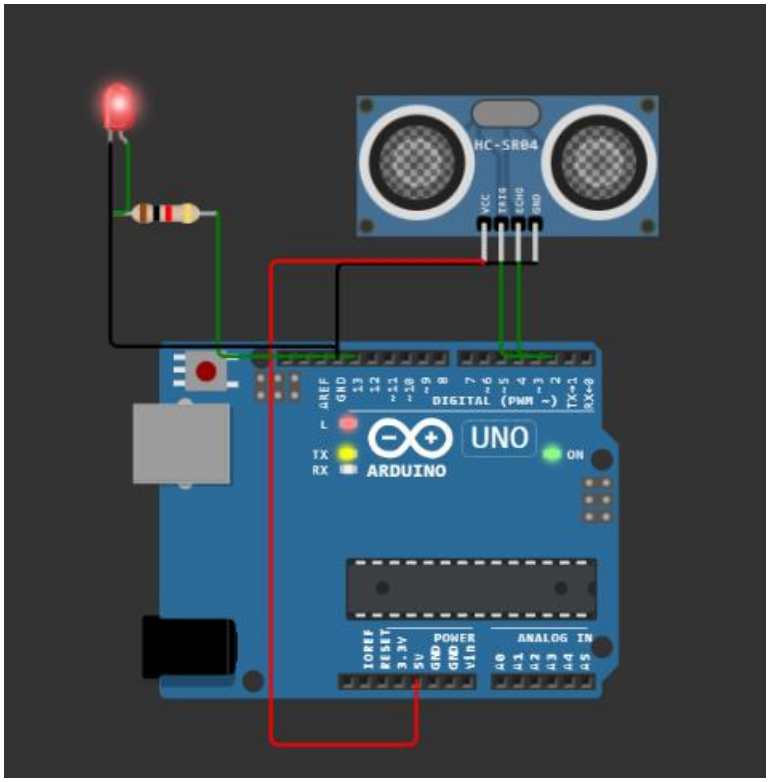
```
  float distance = readDistanceCM();

  bool isNearby = distance < 100;
  digitalWrite(LED_BUILTIN, isNearby);

  Serial.print("Measured distance: ");
  Serial.println(readDistanceCM());

  delay(100);

}
```



- **Distance more than 100cm**

```
#define ECHO_PIN 2
#define TRIG_PIN 3

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
}

float readDistanceCM() {
```

```
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  int duration = pulseIn(ECHO_PIN, HIGH);
  return duration * 0.023 / 2;
}

void loop() {
  // put your main code here, to run repeatedly:
  float distance = readDistanceCM();

  bool isNearby = distance < 100;
  digitalWrite(LED_BUILTIN, isNearby);

  Serial.print("Measured distance: ");
  Serial.println(readDistanceCM());

  delay(100);

}
```