1. Introduction:

LEDs (Light Emitting Diodes) are widely used in electronics to display visual signals. This experiment focuses on two basic tasks using Arduino. First, we blink an LED using simple code, and second, we control the LED with sound input, specifically a clap, using a KY-038 sound sensor. This hands-on activity introduces the fundamental Arduino programming, digital output, and sensor-based input systems.

2. Objective:

The primary objective of this experiment is to understand how to blink an LED using an Arduino Uno. Additionally, this activity aims to teach how to detect sound using the KY-038 sound sensor and how to control the LED state (ON/OFF) based on the sound detected.

3. Prerequisites:

- A basic understanding of Arduino and electronic components
- Arduino IDE installed on your computer
- USB connection between Arduino Uno and PC tested and working

3. Components Required:

- 1 × Arduino Uno R3
- 1 × Breadboard
- 1 × LED
- 1 × 330Ω Resistor
- 1 × KY-038 Sound Sensor
- 4 × Jumper Wires
- 1 × USB Cable for Arduino

4. Procedure:

Part 4.1: Blink LED using Arduino

1. Place the LED on the breadboard.
2. Connect the long leg (anode) of the LED to pin 2 of Arduino through a 330Ω resistor.
3. Connect the short leg (cathode) of the LED to GND.
4. Open Arduino IDE and upload the following code:

```
void setup() {
  pinMode(2, OUTPUT);   // Set pin 2 as output
}

void loop() {
  digitalWrite(2, HIGH); // Turn LED ON
  delay(1000);           // Wait 1 second
  digitalWrite(2, LOW);  // Turn LED OFF
  delay(1000);           // Wait 1 second
}
```

5. After uploading, observe the LED blinking at a 1-second interval.
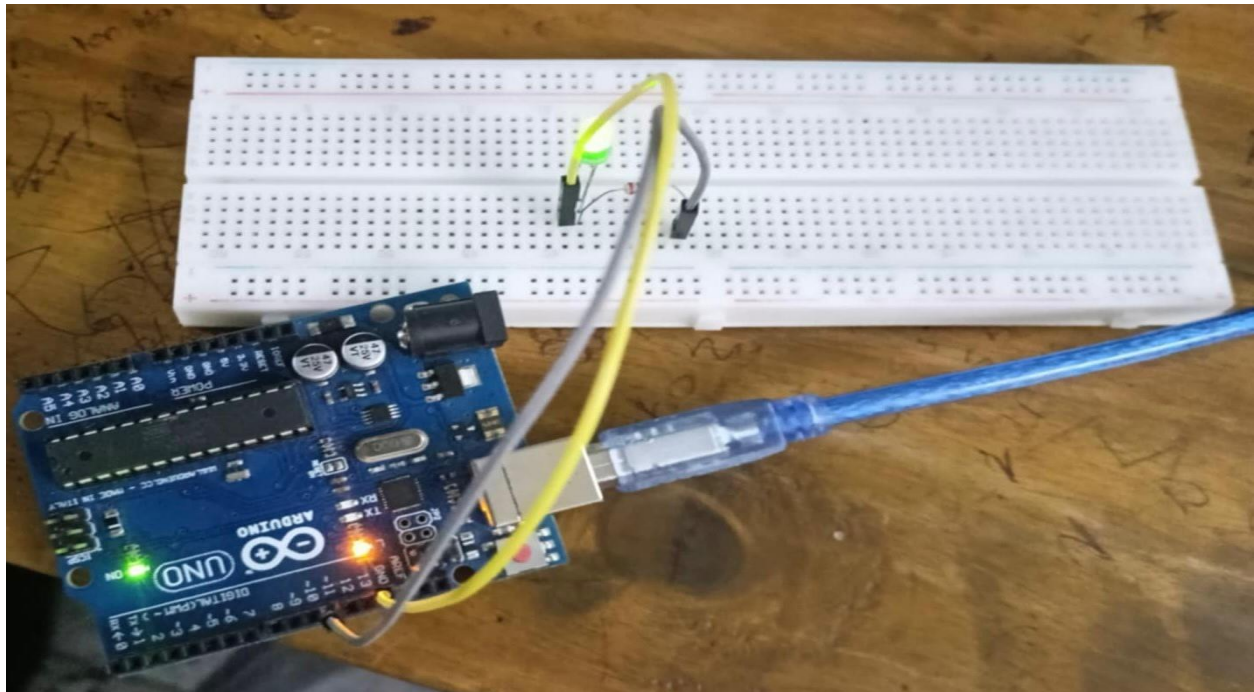
Part 4.2: Control LED using Clap with KY-038

1. Connect VCC of the KY-038 to 5V on Arduino.
2. Connect GND of the KY-038 to GND on Arduino.

3. Connect DO (Digital Out) of KY-038 to pin 3 on Arduino.
4. Keep the LED connected to pin 2 as in Part A.
5. Upload the following code:

```
int ledPin = 2;
int soundSensor = 3;
int soundState = 0;

void setup() {
  pinMode(ledPin, OUTPUT);
  pinMode(soundSensor, INPUT);
}

void loop() {
  soundState = digitalRead(soundSensor);
  if (soundState == HIGH) {
    digitalWrite(ledPin, HIGH); // Turn on LED if sound
detected
    delay(500);                  // Short delay to avoid
noise bouncing
  } else {
    digitalWrite(ledPin, LOW);  // Turn off LED
  }
}
```

6. Clap near the sensor. The LED should turn ON briefly when a sound is detected.
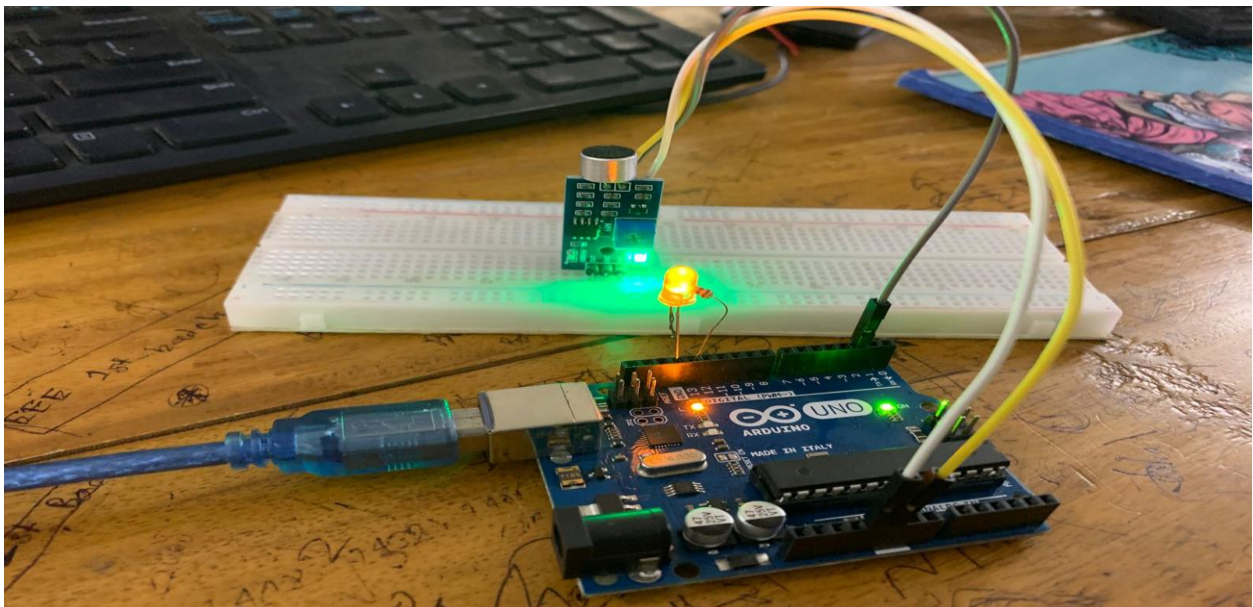
5. Result:

The results of this experiment confirm the expected outcomes. In first part, the LED successfully blinked at regular 1-second intervals, verifying that the Arduino can control output devices using basic timing functions. In second part, the LED responded to claps or sharp sounds, turning on momentarily when sound was detected by the KY-038 sensor.

6. Conclusion:

Through this experiment, we successfully learned how to control an LED using Arduino by both timed logic and external sound input. The blinking LED task introduced the basics of digital output and timing using the delay() function. The clap-controlled LED task introduced sensor integration and digital input processing. Overall, this experiment demonstrates how microcontrollers can interact with the real world through sensors and outputs, forming a foundation for more advanced embedded system and automation projects.