Lab Report No: 03

Lab Title: Blinking LED Controlled by External Sound

Introduction:

This project demonstrates a sound-controlled system where an LED toggles (ON/OFF) in response to external sounds, such as claps or snaps. A sound sensor detects audio signals, which are processed by a microcontroller (Arduino) to determine if the sound exceeds a predefined threshold. If met, the LED changes its state. This project showcases practical applications of sound-responsive systems, emphasizing sensor integration, signal processing, and control logic.

Objectives:

The objective of this project is to design and implement a sound-responsive system that toggles an LED (ON/OFF) based on external sound inputs, such as claps or snaps. The project aims to:

- Demonstrate the use of a sound sensor to detect and convert sound waves into electrical signals.
- Utilize a microcontroller to process the sound signal and control an LED.
- Explore practical applications of sound-controlled systems in real-world scenarios, such as automation and interactive devices.
- Provide hands-on experience with sensor integration, signal thresholding, and control mechanisms.

Components and Equipment:

Hardware:

- Arduino board (e.g., Uno, Nano, Mega, etc.)
- LED (Green color)
- Grove Sound sensor module
- Connecting wires
- Breadboard

Software:

- Arduino IDE

Theory:

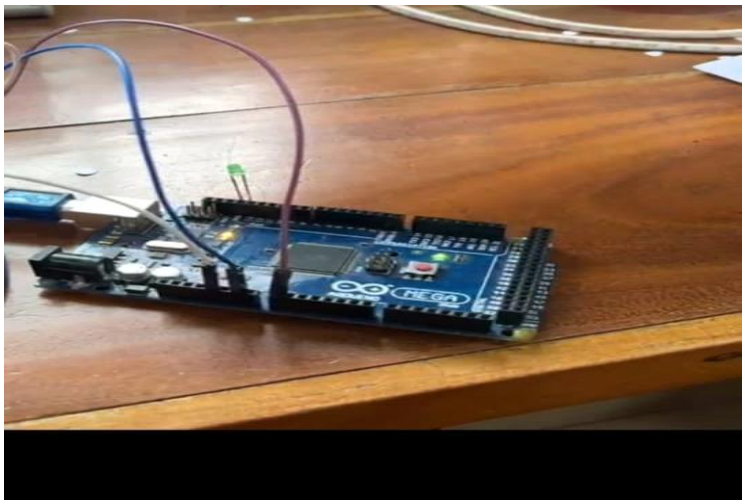
Sound Sensor Operation:

- Converts sound vibrations into an electrical signal.
- The output is compared to a threshold to detect the presence of sound.
- Microcontroller Function:
- Reads the sound sensor signal.
- Turns the LED on/off based on predefined sound conditions (e.g., sound level threshold or clapping pattern).

Working Procedure:

Hardware Setup:

- Connect the sound sensor output pin to a digital/analog pin on the Arduino.
- Here the GND pin of sound sensor will be connected to the GND pin of Arduino.
- The VCC pin of sound sensor will be connected to the 5V pin of Arduino.
- The Signal pin of sound sensor will be connected to the A0 pin of Arduino.
- Connect the LED's short leg to GND of Arduino and long leg to another digital pin 11 on the Arduino



Programming:

- Write an Arduino sketch:
- Read the sound sensor value.
- Compare it to a threshold.
- Toggle the LED state if the threshold is exceeded.

Arduino code:

```
const int soundSensorPin=A0;
const int ledPin=11;
const int threshold=500;
bool ledState=false;
int lastSoundLevel=0;

void setup() {
  PinMode(ledPin,OUTPUT);
  SERIAL.BEGIN(9600);
}

void loop() {
  int SoundLevel=analogRead(SoundSensorPin);
  Serial.println(SoundLevel);
  if(SoundLevel>threshold && lastSoundLevel <= threshold){
    ledState=ledstate;
    digitalWrite(ledPin ,ledState ? HIGH :LOW);
    Serial.println(ledState ? "LED ON" : "LED OFF");
    delay(500);
  }
   lastSoundLevel=soundLevel;
   deley(10);
}
```

Testing:

- Generate sound (e.g., clap or snap).
- Observe the LED behavior in response to the sound input.

Key Observations:

- The sensor readings varied significantly depending on the environment.
- Threshold values needed calibration based on background noise.
- Adding a debounce delay prevented multiple triggers for a single sound event.
- Challenges faced during the implementation.

Results:

- The LED successfully toggled its state in response to specific sound inputs.
- Threshold adjustments were required to filter background noise.
- We get various value by changing the threshold value .

Table 01: Sound Sensor Input vs. LED State

Test No.	Environment	Sound Input (e.g., Clap, Snap)	Sound Sensor Reading	Threshold Value	LED State (On/Off)	Remarks
1	Quiet Room	Clap	550	500	ON	LED turned on as expected.
2	Quiet Room	Snap	480	500	OFF	Sound level below threshold.
3	Moderate Background Noise	Clap	620	600	ON	Adjusted threshold works well.
4	Moderate Background Noise	Snap	590	600	OFF	Background noise filtered out.
5	Loud Environment	Clap	700	650	ON	LED toggled despite noise.
6	Loud Environment	Random noise	680	650	OFF	Misfire avoided by debounce.

Applications:

- Sound-activated lights.
- Noise-responsive toys or devices.
- Security systems with audio triggers.

Conclusion:

The project successfully demonstrated the implementation of a sound-controlled LED system, where the LED toggled (ON/OFF) in response to external sound inputs such as claps or snaps. The sound sensor effectively detected audio signals, which were processed by the microcontroller to determine whether the sound intensity exceeded a predefined threshold.

The system's performance was reliable under controlled conditions, with minimal false triggers when calibrated properly. Debouncing techniques were employed to prevent multiple triggers from a single sound input, enhancing the system's accuracy. The project highlighted the importance of threshold adjustments to filter background noise in varying environments.