**Phase 3 deployment part 1**

**Name:** Gurram Jagadeesh

**Reg no:** 711121106027

**Things used in this project:**

**1.Hardware components**

**a.LM393 sound sensor**

**Sound Detection Sensor Module Features & Specifications**

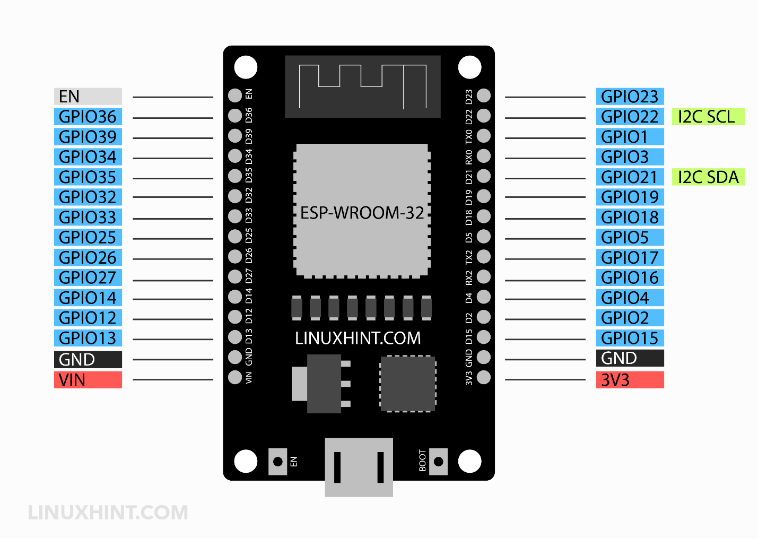
* Operating Voltage: 3.3V to 5V DC
* LM393 comparator with threshold preset
* PCB Size: 3.4cm \* 1.6cm
* Induction distance: 0.5 Meter
* Operating current:  4~5 mA
* Microphone Sensitivity (1kHz): 52 to 48 dB
* Easy to use with Microcontrollers or even with normal Digital/Analog IC

### ****How to Use Sound Detection Sensor Module****

Sound Detection Sensor Module consists of four pins i.e. VCC, GND, DO, AO. Digital out pin is connected to the output pin of LM393 comparator IC while the Analog pin is connected to Microphone.

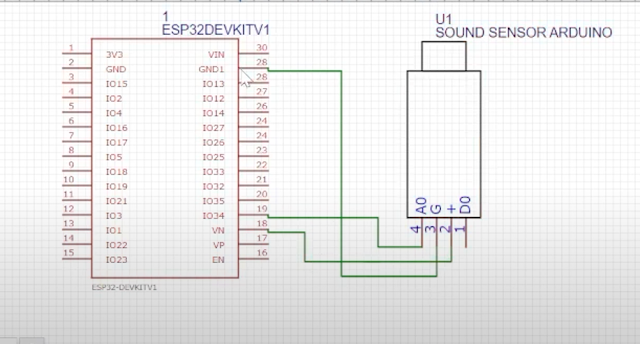
Using Sound Detection Sensor Module with a microcontroller is very easy. Connect the Analog/Digital Output pin of the module to the Analog/Digital pin of Microcontroller. Connect VCC and GND pins to 5V and GND pins of Microcontroller. When the sound level exceeds the setpoint, an LED on the module is illuminated and the output is set low.

**b.ESP32 Development board**



The ESP32 microcontroller has Wi-Fi and Bluetooth functionality, Bluetooth Low Energy (BLE) communication, independent timers, analog to digital and digital to analog converters (ADCs and DACs), capacitive touch sensors, and a Hall effect sensor. The ESP32 microcontroller includes two 240MHz cores, each with a Tensilica Xtensa 32-bit LX6 microprocessor. The ESP32 microcontroller is incorporated in several formats, ranging from a development board to an integrated watch with touchscreen and GPS.

**Circuit diagram**



**Source code**

*const int soundSensorPin = A0; // Analog pin connected to LM393 sound sensor*

*const int ledPin = 13; // Pin number for onboard LED*

*void setup() {*

*Serial.begin(9600);*

*pinMode(soundSensorPin, INPUT);*

*pinMode(ledPin, OUTPUT);*

*}*

*void loop() {*

*int sensorValue = analogRead(soundSensorPin);*

*Serial.print("Sound Level: ");*

*Serial.println(sensorValue);*

*// You might need to adjust this threshold based on your sensor and environment*

*int threshold = 500; // Adjust this threshold based on your environment*

*if (sensorValue > threshold) {*

*digitalWrite(ledPin, HIGH); // Turn on the LED if noise level exceeds the threshold*

*Serial.println("Noise detected!");*

*// You can add code here to send data to a server or perform other actions when noise is detected*

*} else {*

*digitalWrite(ledPin, LOW); // Turn off the LED if noise level is below the threshold*

*}*

*delay(1000); // Delay for 1 second before taking another reading*

*}*

**About Code:**

Sound Sensor Pin is the analog pin to which the LM393 sound sensor is connected.

Led Pin is the pin number to which the onboard LED of the ESP32 board is connected.

The setup() function initializes the serial communication and sets the pin modes.

The loop() function reads the analog value from the sound sensor, compares it with the threshold (you may need to adjust this threshold based on your sensor and environment), and turns on the LED and prints a message if the noise level exceeds the threshold.

Make sure you have the necessary libraries installed in your Arduino IDE for ESP32 board support. You might also need to install additional libraries for the LM393 sound sensor if required by the manufacturer.

Please consult the datasheets and documentation of your ESP32 board and LM393 sound sensor for specific pin configurations and details about the sensor's output range to set an appropriate threshold for noise detection.

**Schematic diagram :**

