Date:

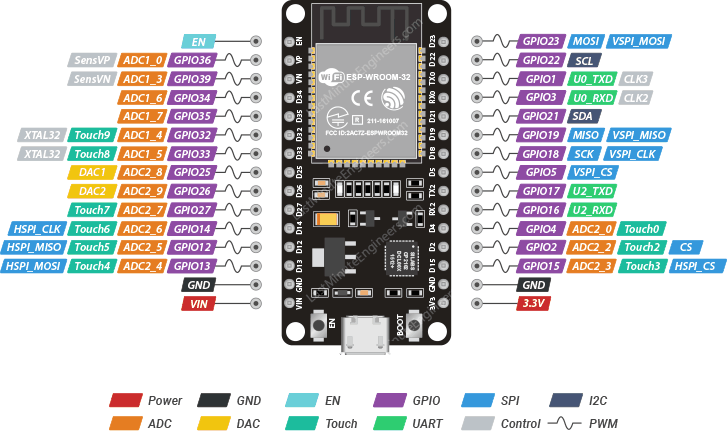
**EXPERIMENT: 10**

**AIM:** Implement interfacing of sensors with Arduino/ESP32. (Ultrasonic Sensor, Temperature and humidity, Light Sensor with Buzzer).

**OBJECTIVES:**

1. To measure distance of object using ultrasonic sensor.
2. To measure temperature and humidity values.
3. To control buzzer depending on light intensity.

**COMPONENTS:**

1. **ESP32:**

ESP32 is a series of low-cost, low-power microcontrollers with integrated Wi-Fi and Bluetooth connectivity. [It is suitable for a wide range of IoT applications and can be programmed using various languages and frameworks](https://www.espressif.com/en/products/socs/esp32).

1. **LED:**

LED stands for light-emitting diode, which is a semiconductor device that emits light when an electric current flows through it.

Grove button is a type of button that can be used with the Grove system, which is a modular and easy-to-use platform for connecting sensors, actuators, and displays.

1. **USB CABLES:**

USB cables are cables that can be used to connect, charge, and transfer data between various devices, such as computers, smartphones, cameras, and more.

1. **JUMPER WIRES:**

Jumper wires are wires that have connectors or pins at each end, which can be used to connect two points in a circuit without soldering.

1. **Ultrasonic Sensor:**



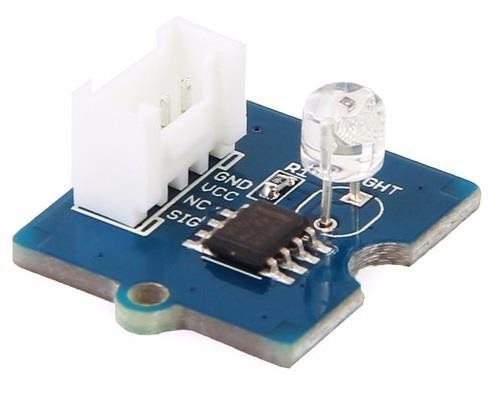
An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves, and converts the reflected sound into an electrical signal.

1. **Temperature and Humidity Sensor:**



This is a multifunctional sensor that gives you temperature and relative humidity information at the same time. It utilizes a DHT11 sensor that can meet measurement needs of general purposes.

1. **Light Sensor:**



The Grove - Light sensor integrates a photo-resistor(light dependent resistor) to detect the intensity of light. The resistance of photo-resistor decreases when the intensity of light increases.

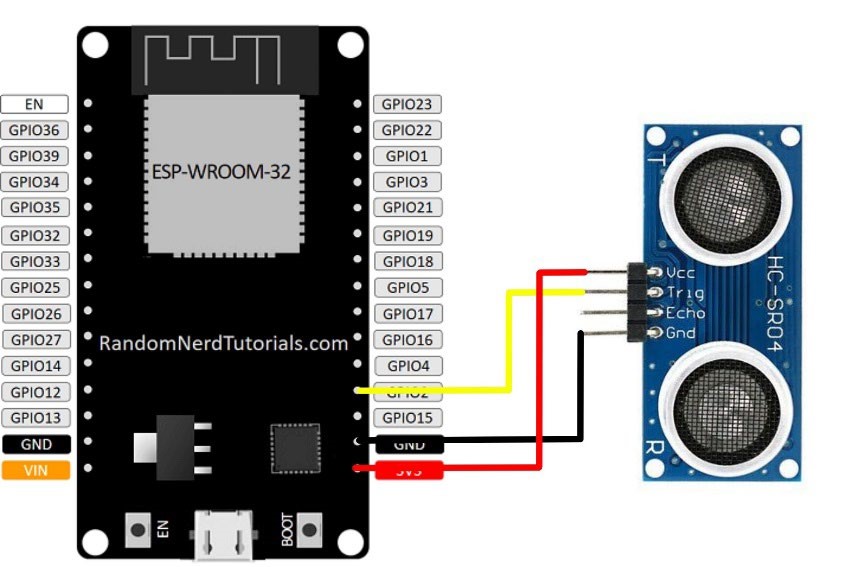
1. **Buzzer:**



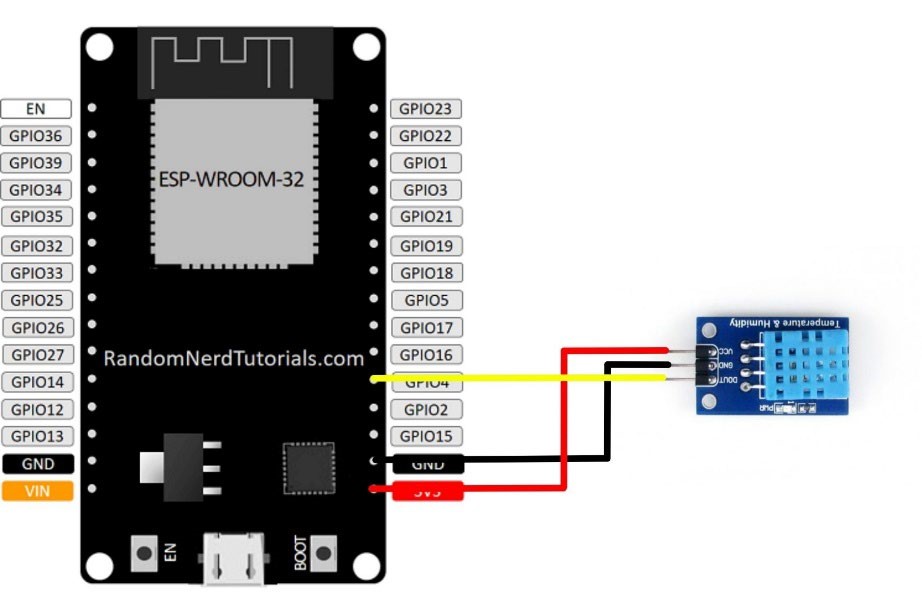
The Grove - Buzzer module has a [piezo buzzer](https://en.wikipedia.org/wiki/Buzzer#Piezoelectric) as the main component. The piezo can be connected to digital outputs, and will emit a tone when the output is HIGH. Alternatively, it can be connected to an analog pulse-width modulation output to generate various tones and effects.

**CONNECTION DIAGRAM:**

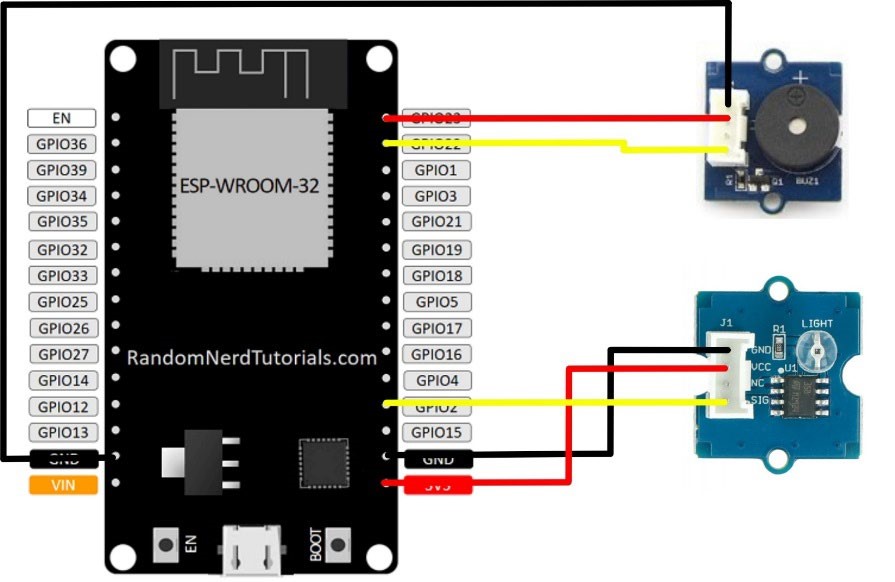
**10.1:**



**10.2:**



**10.3:**



**CODES:**

**10.1:**

**#include "Ultrasonic.h"**

**Ultrasonic ultrasonic(2); // Output Pin void setup()**

**{**

**Serial.begin(9600); // 9600 MHz band**

**}**

**void loop()**

**{**

**long RangeInInches; long RangeInCentimeters;**

**Serial.println("The distance to obstacles in front is: ");**

**RangeInInches = ultrasonic.MeasureInInches();**

**Serial.print(RangeInInches); // 0~157 inches Serial.println(" inch");**

**delay(2000);**

**RangeInCentimeters = ultrasonic.MeasureInCentimeters(); // two measurements should keep an interval**

**Serial.print(RangeInCentimeters); // 0~400cm**

**Serial.println(" cm"); delay(2000);**

**}**

**10.2:**

**#include "DHT.h"**

**#define DHTPIN 2 // what pin we're connected to 5**

**#define DHTTYPE DHT11 //DHT 11**

**DHT dht(DHTPIN, DHTTYPE);**

**#if defined(ARDUINO\_ARCH\_AVR)**

**#define SERIAL Serial**

**#elif defined(ARDUINO\_ARCH\_SAMD) || defined(ARDUINO\_ARCH\_SAM) #define SERIAL SerialUSB**

**#else**

**#define SERIAL Serial**

**#endif void setup()**

**{**

**SERIAL.begin(115200);**

**SERIAL.println("DHT11 test!"); Wire.begin(); dht.begin();**

**}**

**void loop()**

**{**

**float temp\_hum\_val[2] = {0};**

**if (!dht.readTempAndHumidity(temp\_hum\_val))**

**{**

**SERIAL.print("Humidity: ");**

**SERIAL.print(temp\_hum\_val[0]);**

**SERIAL.print(" %\t");**

**SERIAL.print("Temperature: ");**

**SERIAL.print(temp\_hum\_val[1]);**

**SERIAL.println(" \*C");**

**} else {**

**SERIAL.println("Failed to get temprature and humidity value.");**

**}**

**delay(1500);**

**}**

**10.3:**

**int sensorPin = 2; // select the input pin for the potentiometer int ledPin = 15; // select the pin for the LED**

**int sensorValue = 0; // variable to store the value coming from the sensor int suppl = 23; void setup()**

**{**

**// declare the ledPin as an OUTPUT:**

**pinMode(ledPin, OUTPUT); pinMode(suppl, OUTPUT); Serial.begin(115200); digitalWrite(suppl, HIGH);**

**} void loop()**

**{**

**// read the value from the sensor:**

**sensorValue = analogRead(sensorPin); Serial.println(sensorValue);**

**if (sensorValue < 1500)**

**{**

**digitalWrite(ledPin, HIGH);**

**} else**

**{**

**digitalWrite(ledPin, LOW);**

**}**

**// turn the ledPin on**

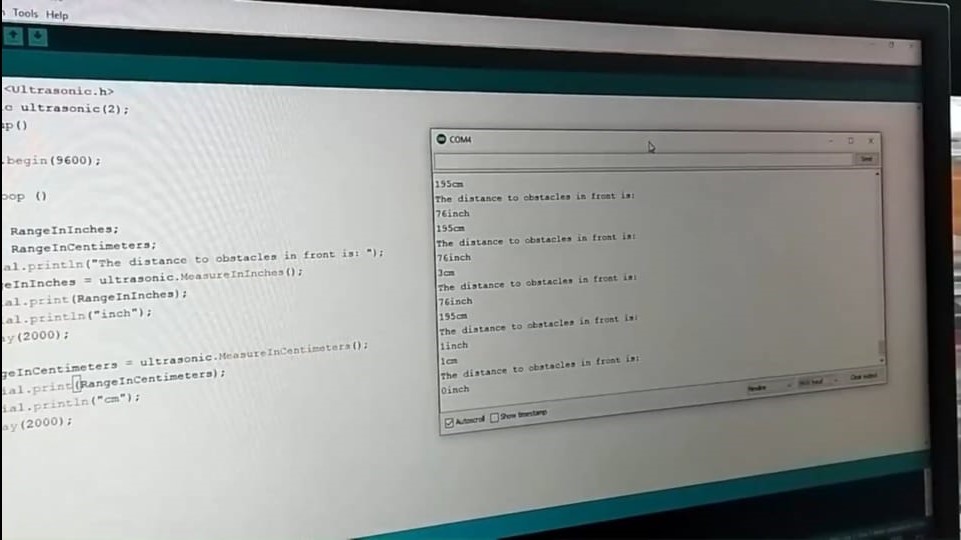
**// stop the program for <sensorValue> milliseconds:**

**// turn the ledPin off:**

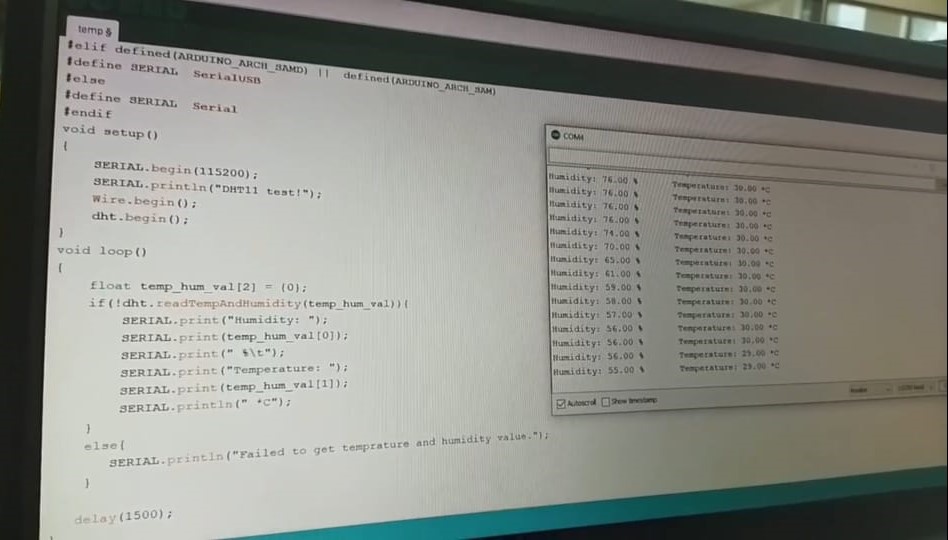
**// stop the program for for <sensorValue> milliseconds: delay(1000);**

**}**

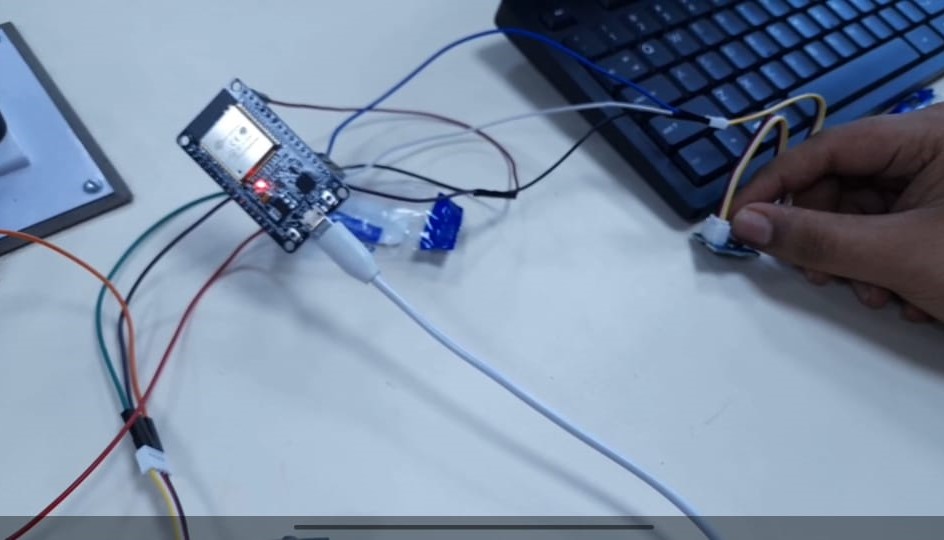
**OUTPUTS:**

**10.1:**

**10.2:**

****

**10.3:**

****

**OBSERVATIONS:**

**CONCLUSION:**

**DRIVE LINK OF VIDEO:**

**10.1:** <https://drive.google.com/file/d/12ayC_dH-YOOqDNBogsWDHpL3PjkuKB9C/view?usp=drivesdk>

**10.2:** <https://drive.google.com/file/d/1HKniFO0LC4EDDvGX-nxnjzV9Sy1HFpqa/view?usp=drivesdk>

**10.3:** <https://drive.google.com/file/d/1IN4F5wtHBDpaJIKBdY8H3HhokKW-6mj-/view?usp=drivesdk>

**SUBMITTED BY:**

1. 23CS041 – DHRUV LOKADIYA
2. 23CS036 – VED KHENI