**IoT Lab Solutions:**

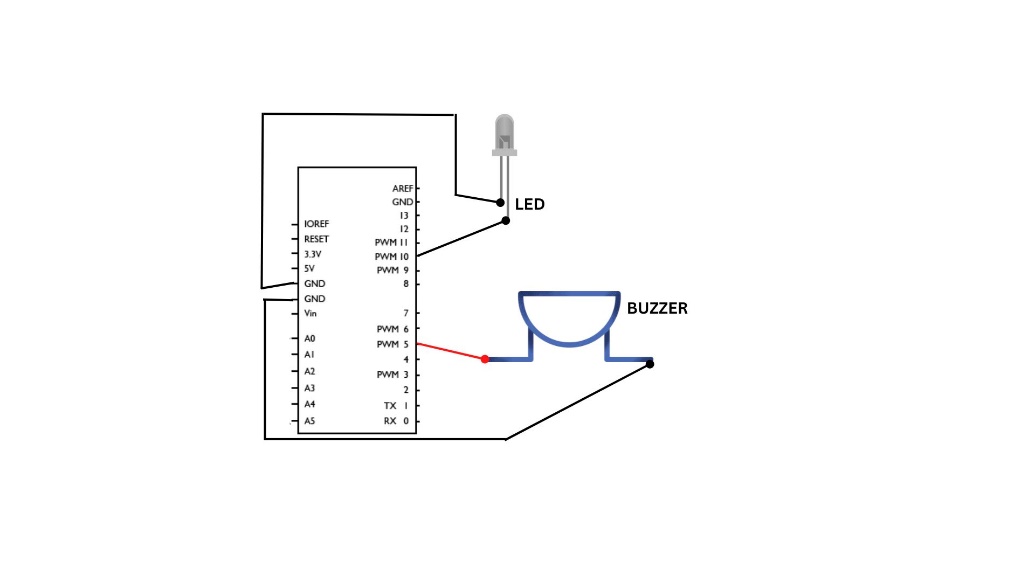
**Experiment No. 1: LED ON/ OFF with some time interval with buzzer ON/OFF.**

Aim: Understand the interface of LED and buzzer with IoT development board and programming.

Components: Arduino Uno, Buzzer, LED, 100 Ohms resister, jumper wires, 5V DC supply.

**Connections:**

|  |  |
| --- | --- |
| From | To |
| Arduino D10 | Anode of LED (long leg) |
| GND | Cathode of LED(short leg) |
| Arduino D5 | Buzzer red wire |
| GND | Buzzer black wire |

****

**Program:**

#define led 10

#define buzzer 5

void setup()

{

pinMode(led,OUTPUT);

pinMode(buzzer,OUTPUT)

Serial.begin(9600);

Serial.println(“LED ON/OFF system”);

void loop()

{

Serial.println(“LED is ON now”);

digital.write(led, HIGH);

digital.write(buzzer, HIGH);

delay(2000);

Serial.println(“LED is OFF now”);

digital.write(led, LOW);

digital.write(buzzer, LOW);

delay(2000);

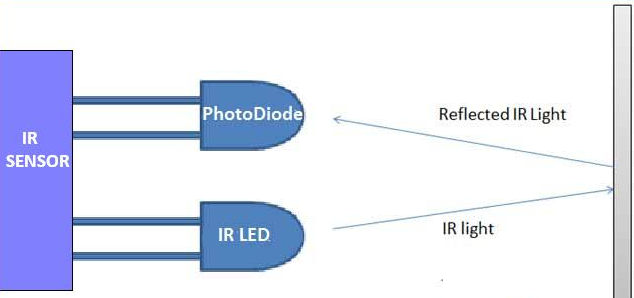
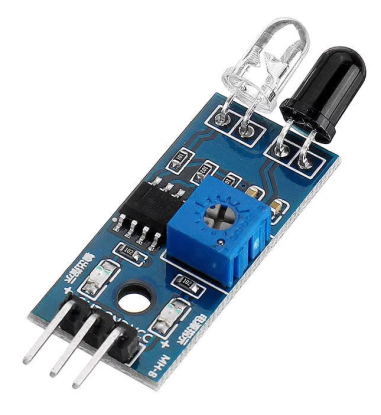
}

**Experiment No. 2: Count number of persons entering a room IR sensor.**

**Aim:** To understand working of IR sensors and its applications.

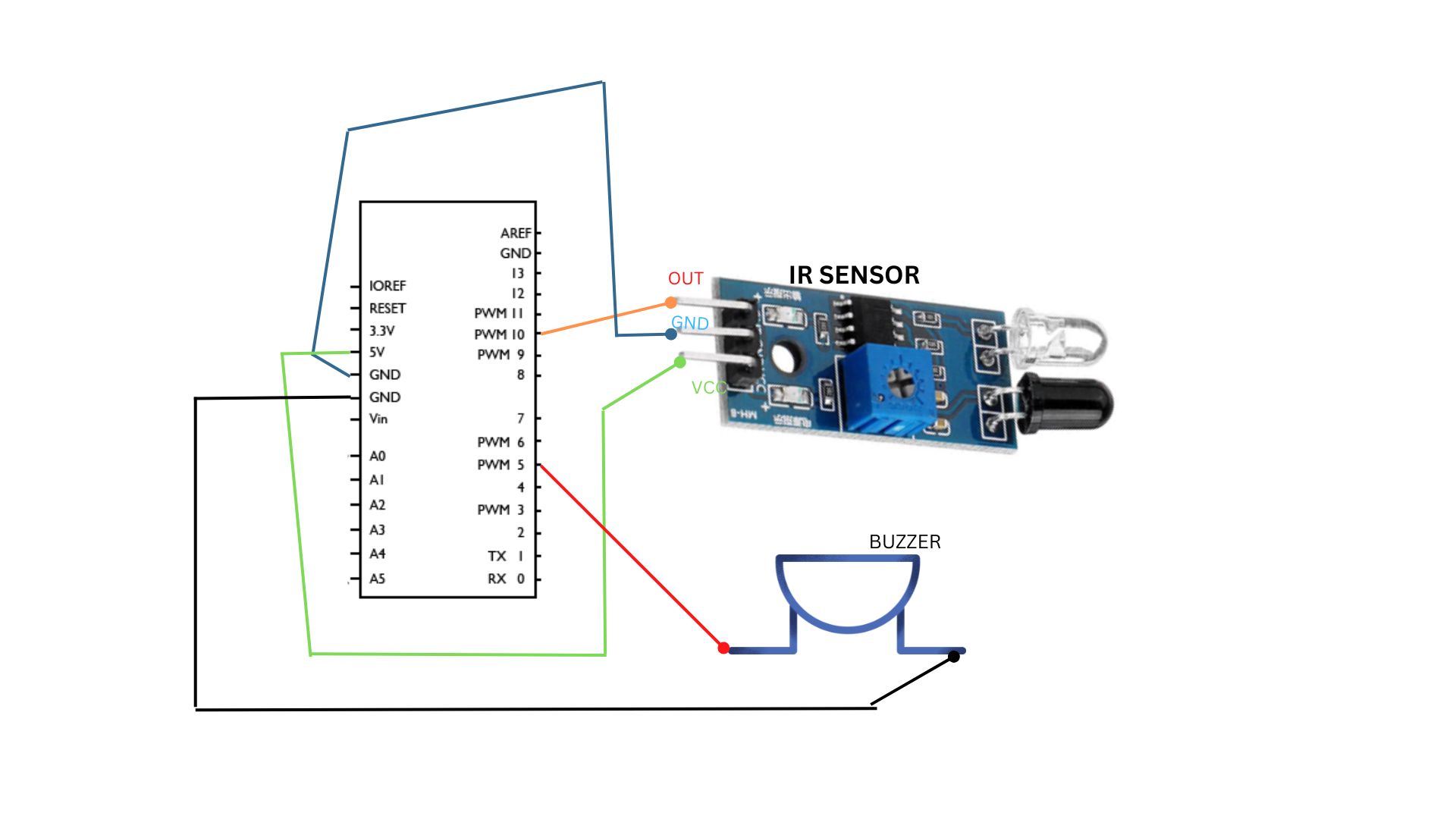
**Components:** IR sensor, Arduino, 5V DC supply

**IR sensor:**



**Connections:**

|  |  |
| --- | --- |
| From | To |
| Arduino D10 | OUT IR sensor |
| Arduino GND | GND IR sensor |
| Arduino 5V pin | Vcc IR Sensor |
| Arduino D5 | Buzzer red wire |
| GND | Buzzer black wire |

****

**Program:**

#define IRout 10

#define led 5

int counter=0;

void setup()

{ pinMode(led,OUTPUT);

pinMode(IRout,INPUT);

Serial.begin(9600); }

void loop()

{

if (digitalRead(IRout)== LOW) {

digitalWrite(led,HIGH); counter ++; delay(10); }

else

{ digitalWrite(led,LOW); delay(10); }

Serial.println(“Total number of persons entered the room”,counter);

} }

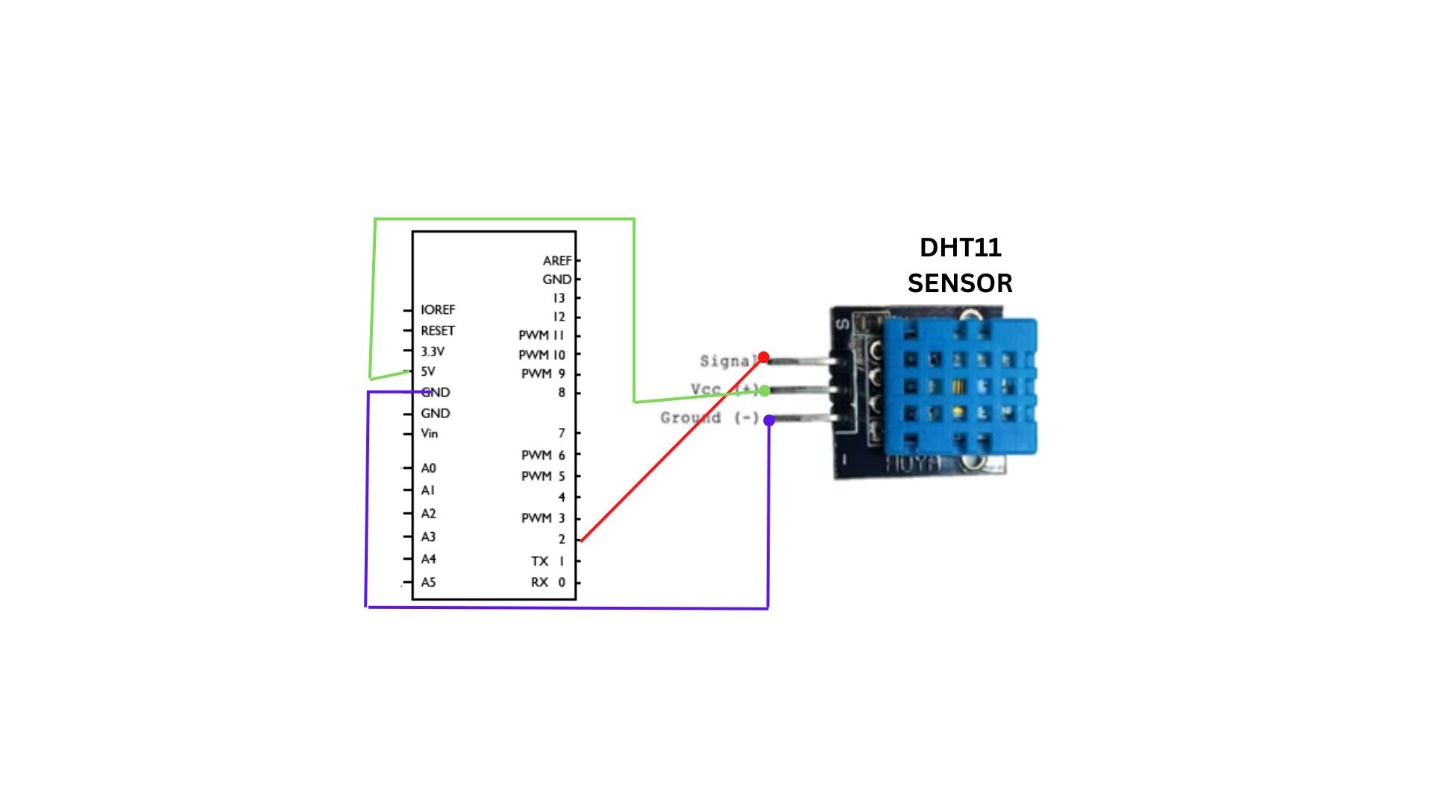
**Experiment No. 3: Measure temperature and humidity in air using DHT11 Sensor.**

Aim: To understand working of DHT11 sensor to automatically monitor temperature and humidity of the surrounding.

**Components:** Arduino Uno, DHT11 sensor, jumper wires, 5V DC supply.

**Connections:**

|  |  |
| --- | --- |
| From | To |
| Arduino D2 | Signal pin of DHT11 |
| Arduino GND | GND of DHT11 |
| Arduino 5V pin | Vcc DHT11 |

****

**Program:**

// REQUIRES the following Arduino libraries:

// - DHT Sensor Library: https://github.com/adafruit/DHT-sensor-library

// - Adafruit Unified Sensor Lib: https://github.com/adafruit/Adafruit\_Sensor

#include "DHT.h"

#define DHTPIN 2 // Digital pin connected to the DHT sensor

#define DHTTYPE DHT11 // DHT 11

DHT dht(DHTPIN, DHTTYPE); // Initialize DHT sensor.

void setup() {

Serial.begin(9600);

dht.begin();

}

void loop() {

delay(2000);

float h = dht.readHumidity();

float t = dht.readTemperature(); // Read temperature as Celsius (the default)

Serial.print(F(" Humidity: "));

Serial.print(h);

Serial.print(F("% Temperature: "));

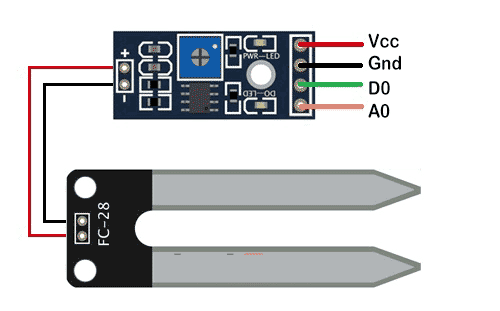
Serial.print(t);

}

**Experiment No. 4: Measure moisture in soil and use suitable LEDs to indicate the moisture levels like below and above required level.**

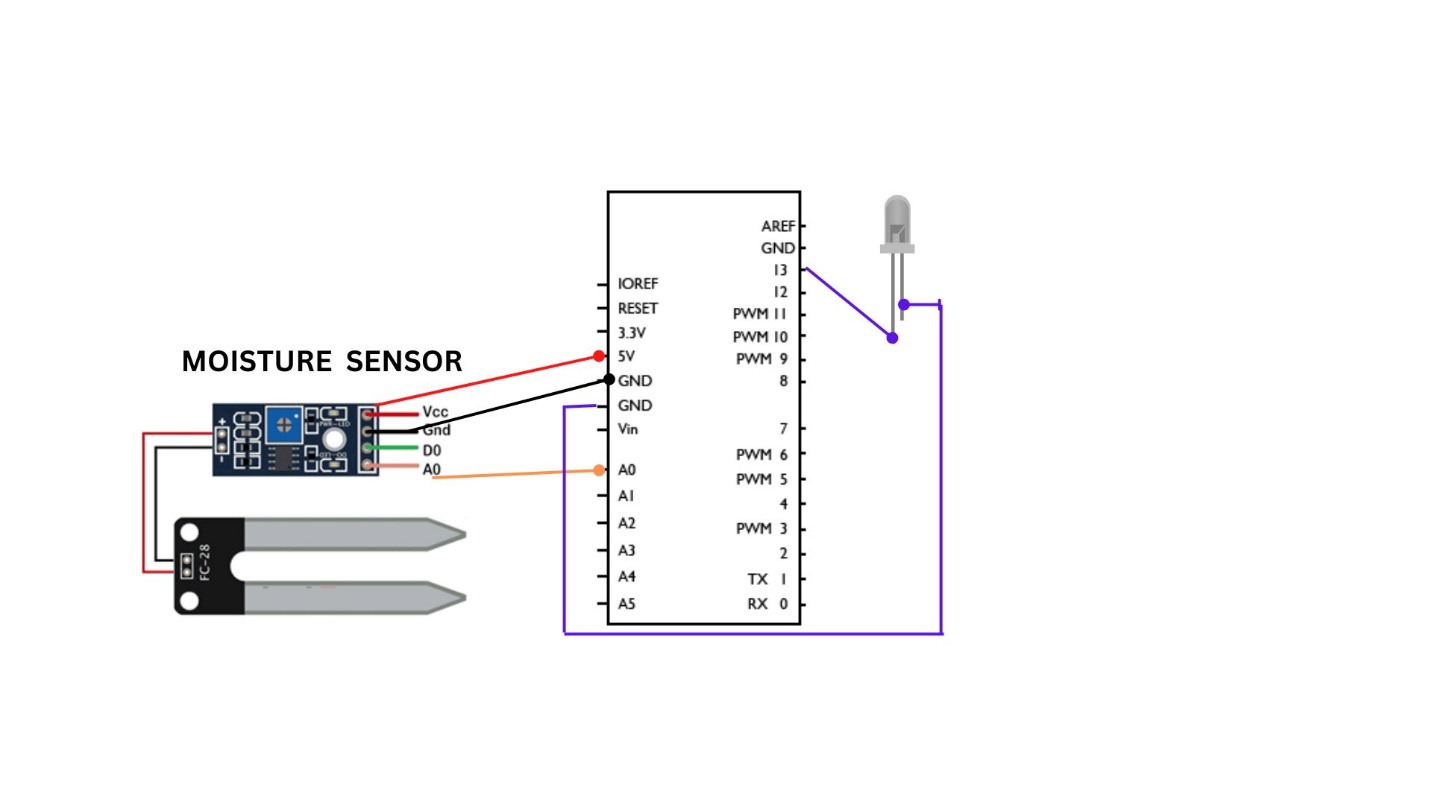
**Components:** Arduino, NodeMCU, Moisture sensor, LED, jumper wires

**Moisture sensor:**



**Connections:**

|  |  |
| --- | --- |
| From | To |
| Arduino A0 | A0 of Moisture sensor |
| Arduino GND | GND of Moisture sensor |
| Arduino 5V pin | Vcc of Moisture sensor |
| Arduino D13 | Anode of LED (ling pin) |
| Arduino GND | Cathode of LED (short pin) |

****

**Program:**

#define sensorPin A0

#define led 13

void setup()

{

Serial.begin(9600);

pinMode(led, OUTPUT); }

void loop() {

sensorValue = analogRead(sensorPin);

Serial.println("Analog Value : ");

Serial.println(sensorValue);

if (sensorValue<limit)

{ digitalWrite(led, HIGH); }

else

{ digitalWrite(led, LOW); }

delay(1000);

}

**Experiment No. 5:** Display messages on LCD module on each button press event.

**Aim:** To understand and implement a output display system using 16x2 LCD . It is very much essential for a working IOT system to show its output.

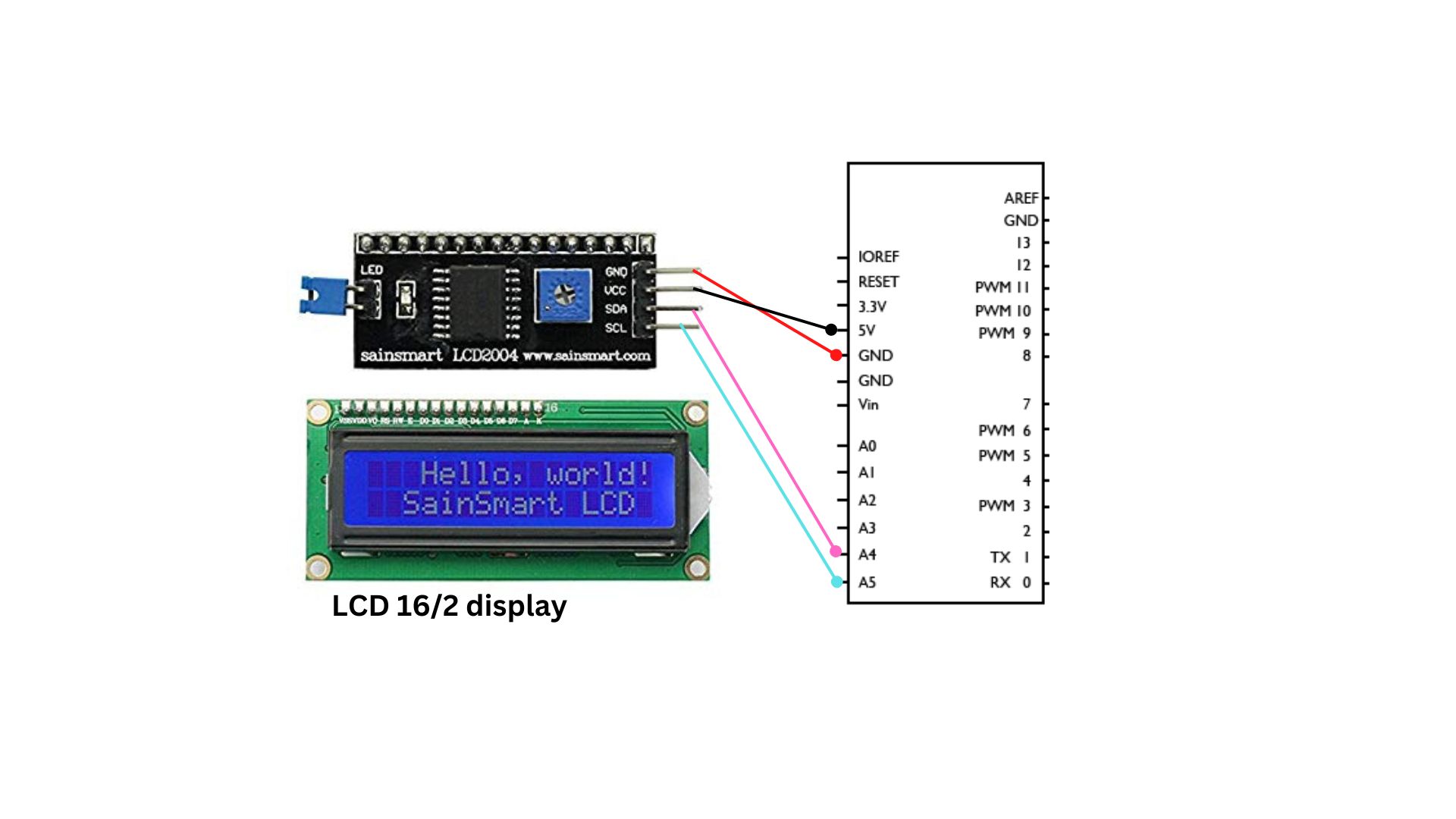
**Components:**

1. LCD 16/2 display
2. Arduino Software
3. Arduino UNO R3
4. Connecting wires.

**Connections:**

To wire your LCD screen to your board, connect the following pins:

* LCD RS pin to digital pin 12
* LCD Enable pin to digital pin 11
* LCD D4 pin to digital pin 5
* LCD D5 pin to digital pin 4
* LCD D6 pin to digital pin 3
* LCD D7 pin to digital pin 2
* LCD R/W pin to GND
* LCD VSS pin to GND
* LCD VCC pin to 5V
* LCD LED+ to 5V through a 220 ohm resistor
* LCD LED- to GND



Program:

#include <LiquidCrystal.h>

const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;

int kount=0;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup()

{

lcd.begin(16, 2);

lcd.print("hello, world!");

}

void loop()

{

lcd.setCursor(0, 1);

lcd.print(“CSE Department”);

lcd.setCurson(15,0); // column 15, First line

lcd.print(kount); // counter

delay(10000);

lcd.setCursor(0, 1);

lcd.print(“PDA College”);

delay(10000);

kount++;

}

**Experiment No. 6: Make to glow three different color LEDs by switching through Relays.**

**Aim:** To understand how to build control systems using relays.

**Components:**

Arduino board, three relays (5V), three different color LEDs, Jumpers, bread board.

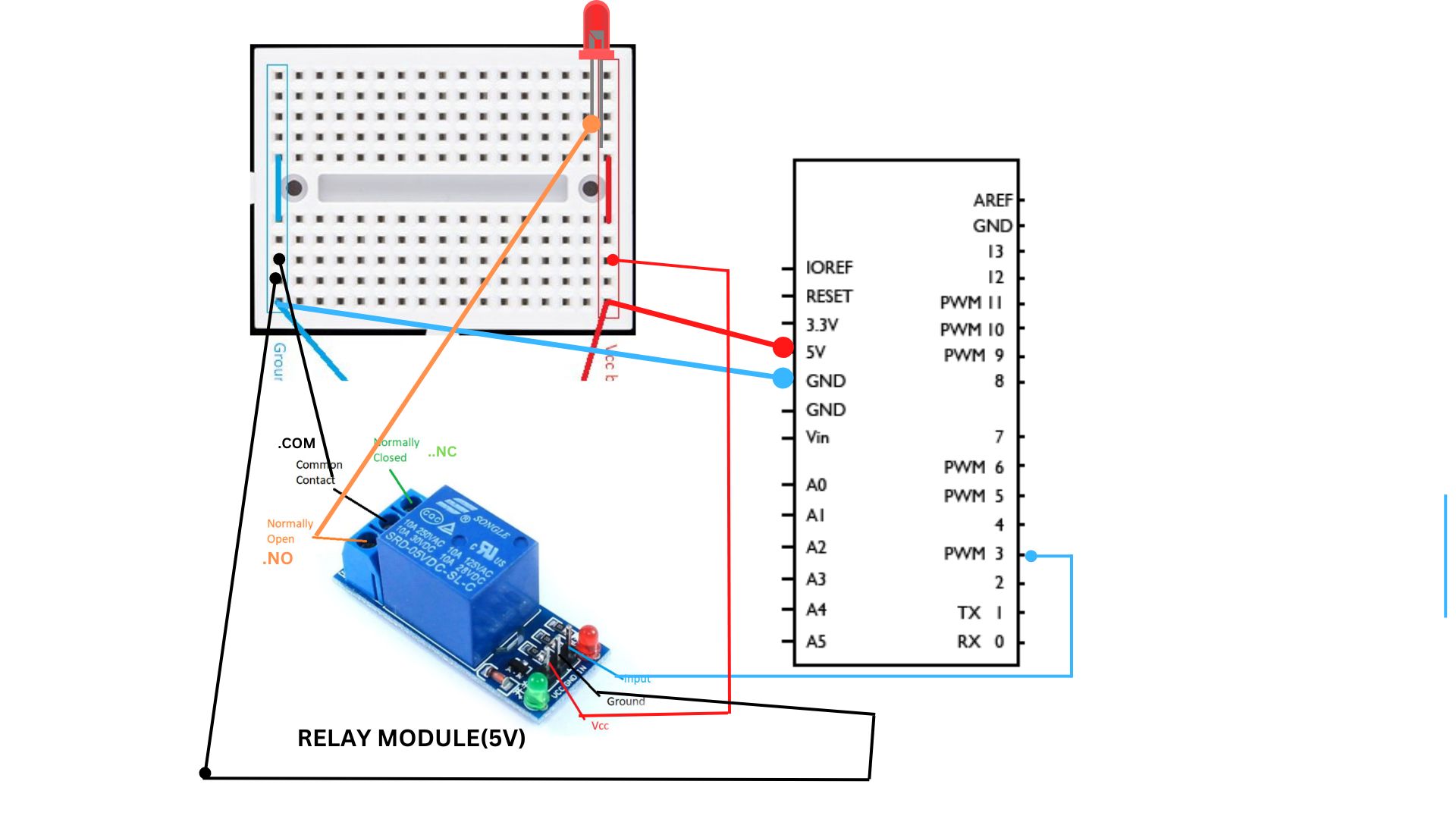
Relay (5v) and connections:

**The relay module has 3pin screw terminal.**

**a). NO - Normally open**

**b). COM - Common**

**c). NC - Normally closed**

****

**Program:**

int Relaypin1= 3; // Define input pin for relay

int Relaypin2= 4; // Define input pin for relay

int Relaypin3= 5; // Define input pin for relay

void setup() {

// put your setup code here, to run once:

pinMode(Relaypin1, OUTPUT);

pinMode(Relaypin2, OUTPUT); // Define the Relaypin2 as output pin

pinMode(Relaypin3, OUTPUT); // Define the Relaypin3 as output pin

}

void loop()

{

digitalWrite(Relaypin1, HIGH); // Sends high signal

delay(1000); // Waits for 1 second

digitalWrite(Relaypin1, LOW); // Makes the signal low

delay(1000); // Waits for 1 second

digitalWrite(Relaypin2, HIGH); // Sends high signal

delay(1000); // Waits for 1 second

digitalWrite(Relaypin2, LOW); // Makes the signal low

delay(1000); // Waits for 1 second

digitalWrite(Relaypin3, HIGH); // Sends high signal

delay(1000); // Waits for 1 second

digitalWrite(Relaypin3, LOW); // Makes the signal low

delay(1000); // Waits for 1 second

}

**Experiment No. 8: Acoustic controlled motor.**

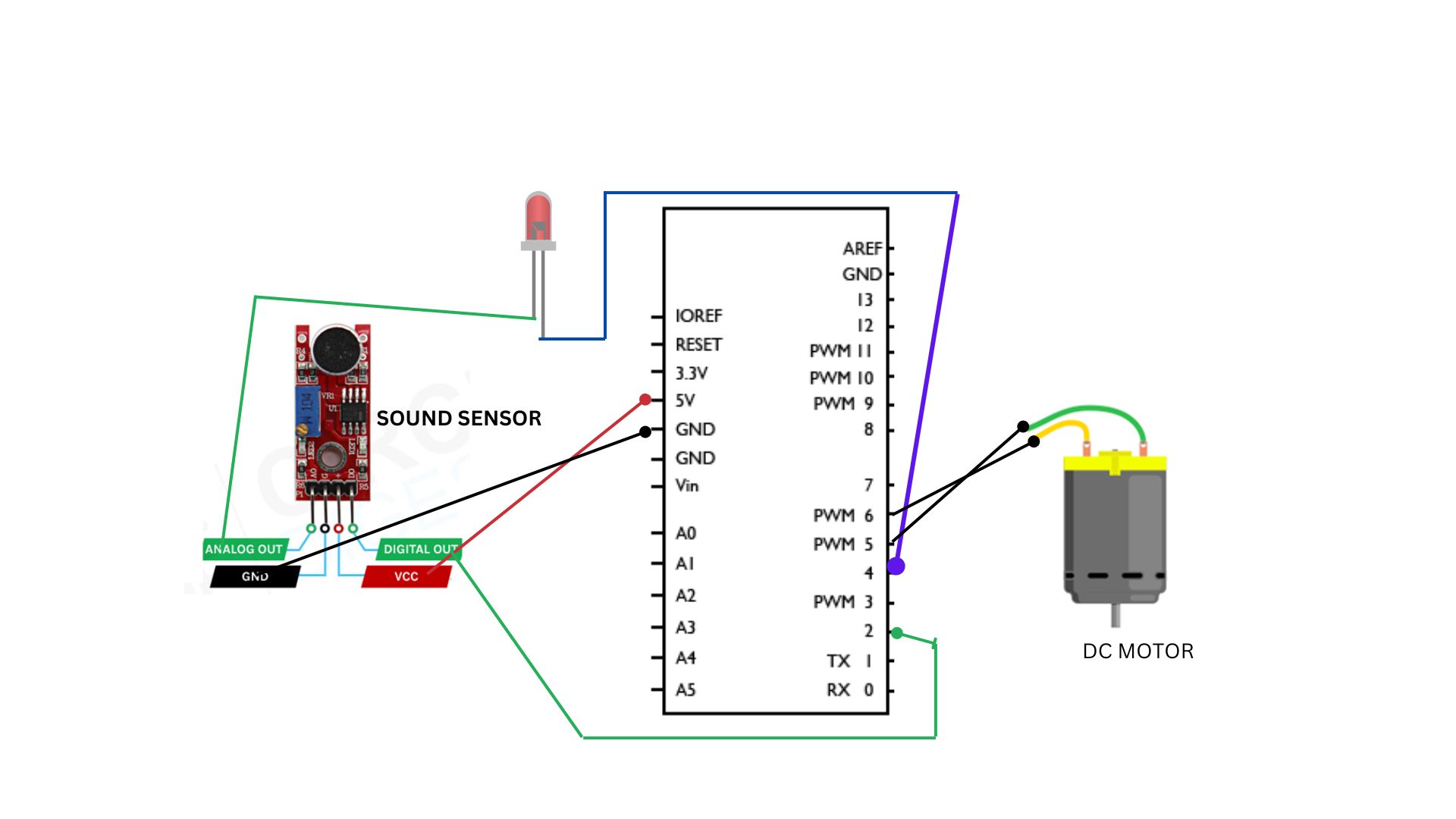
**Aim:** This experiment will provide idea about a control system that is controlled by acoustic (sound) signal.

**Components required:**

* Arduino UNO
* Sparkfun sound detecter
* Jumper wires
* DC motor (generic)
* LED
* Resister 200 Ohms
* Breadboard

**Connections:**

|  |  |
| --- | --- |
| Arduino Pin D2 | Sound sensor Digital Out pin |
| Arduino 5V | Sound sensor Vcc |
| Arduino GND | Sound sensor GND |
| Arduino Pin D4 | LED positive (long leg) |
| Arduino Pin 5 | One wire of DC motor |
| Arduino Pin 6 | Another wire of DC motor |



/\*For this project you will have to connect external power supply of 9V or 5V to the 12V pin of driver.\*/

int soundSensor=2;

int LED=4;

void setup() {

pinMode(soundSensor,INPUT);

pinMode(LED,OUTPUT);

Serial.begin(9600);

}

void loop()

{

int SensorData=digitalRead(soundSensor);

if(SensorData==1)

{

Serial.println(“sound detected”);

digitalWrite(LED, HIGH);

delay(1000);

}

}

**Experiment NO. 9:** Develop a working model to Switch ON/OFF LED based on Switch ON/Switch OFF commands received from smart phone.

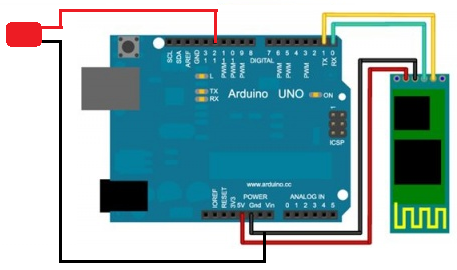
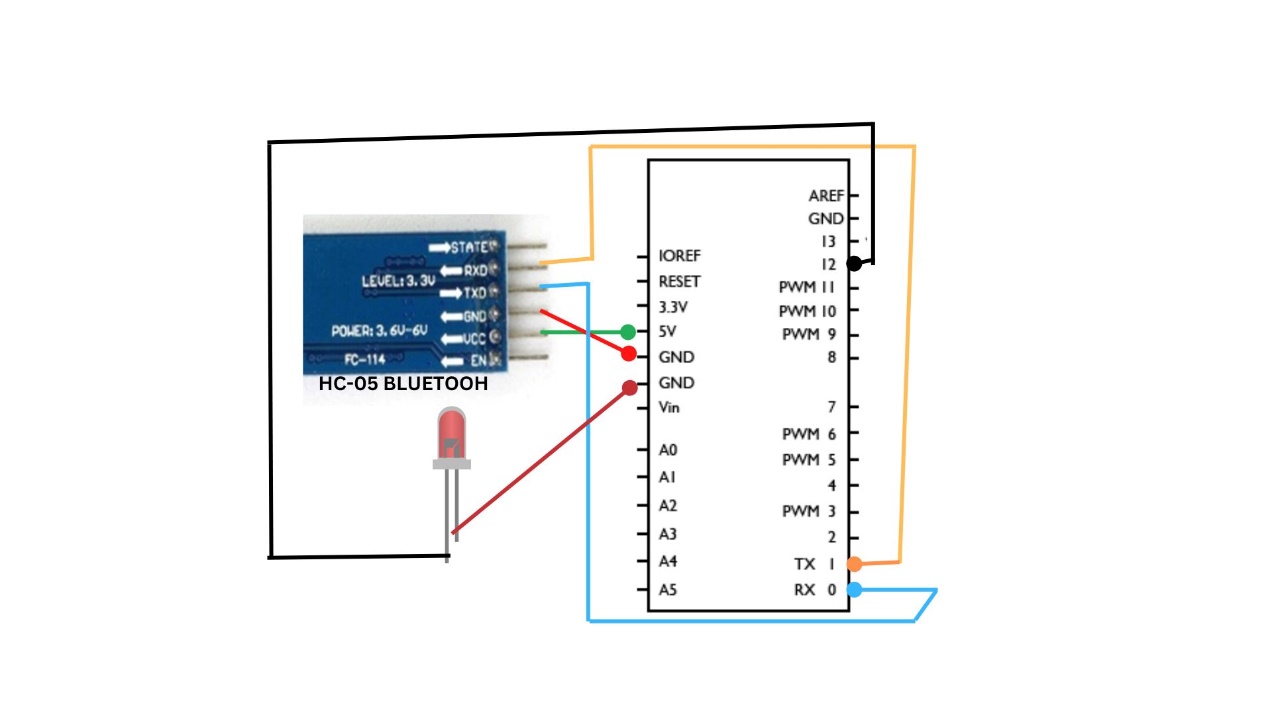


**Components required:**

* Arduino board
* HC-05 bluetooth module
* Bluetooth SPPO Pro APK (android program ) – Download from Google Paly Store.
* Jumper wires
* LEDs
* 5 V Power supply

**Connections:**

* Arduino 5v to VCC Bluetooth module
* Arduino GND to GND of Bluetooth module
* Arduio RXD to TXD of Bluetooth module
* Arduio TXD to RXD of Bluetooth module
* LED anode to PIN 12 of Arduino
* LED cathode to GND of Arduino



**Program:**

#include <HC05.h>

#include <SoftwareSerial.h>

#define LED 12

SoftwareSerial mySerial(0, 1);

int command=-1; //Int to store app command state.

void setup()

{

pinMode(LED, OUTPUT); //Set the LED pin.

digitalWrite(LED, HIGH);

mySerial.begin(38400); //Set the baud rate to your Bluetooth module.

}

void loop()

{

if (mySerial.available() > 0)

{

command = mySerial.read();

switch (command)

{

case '0': // zero for LED off

{

digitalWrite(LED, LOW);

delay(1000);

}

break;

case '1': // one for LED on

{

digitalWrite(LED, HIGH);

delay(1000);

}

break;

} // switch

command=-1;

} }

**Experiment No. 11:** Develop a working model to mark attendance using RFID module and store data in cloud.

Aim: To understand the interface of arduino with RFID for developing useful applications.

**PinWiring to Arduino Uno:**

SDA------------------------Digital 10

SCK------------------------Digital 13

MOSI----------------------Digital 11

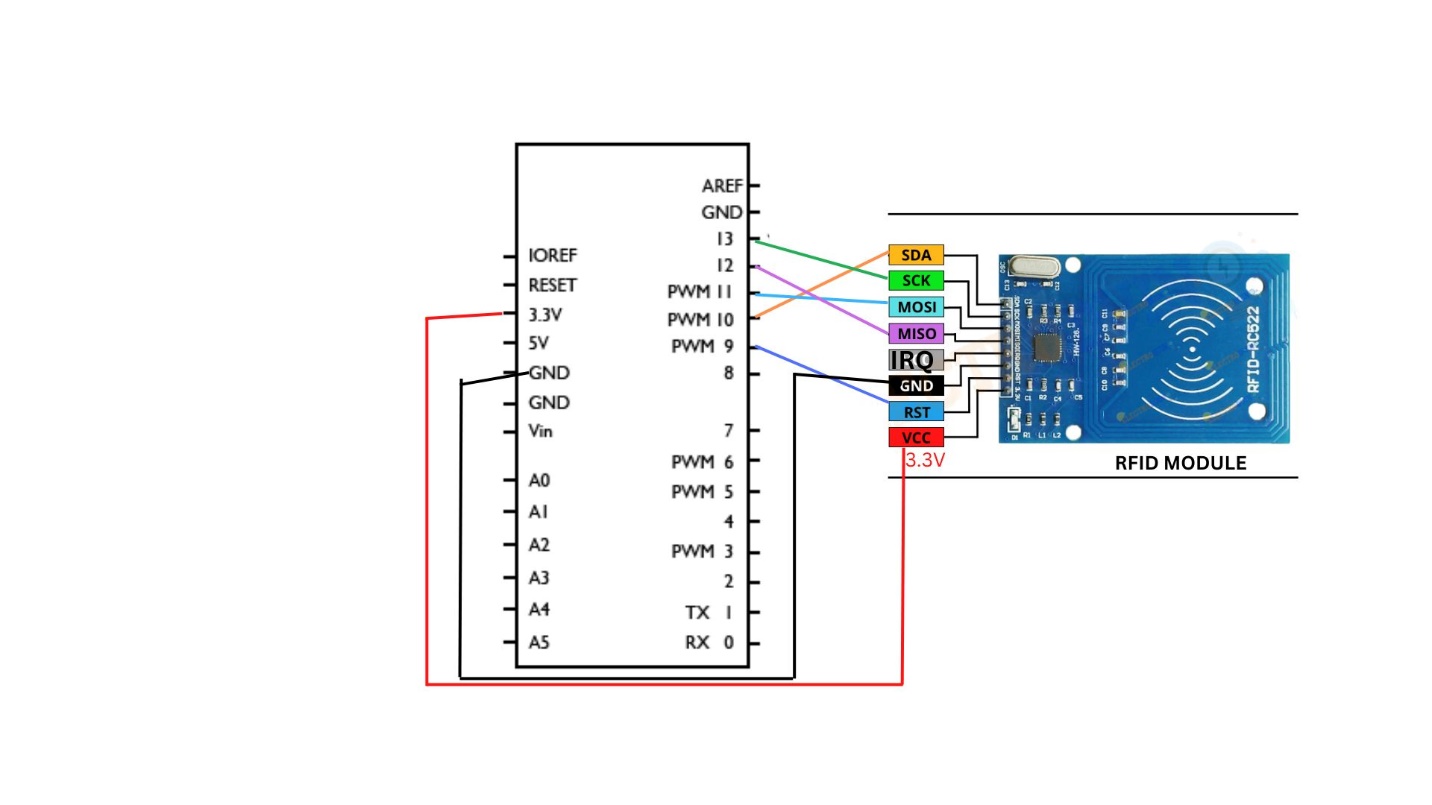
MISO----------------------Digital 12

IRQ------------------------unconnected

GND-----------------------GND

RST------------------------Digital 9

3.3V------------------------3.3V (DO NOT CONNECT TO 5V)



**Program:**

#include <SPI.h>

#include <MFRC522.h>

#define SS\_PIN 10

#define RST\_PIN 9

MFRC522 mfrc522(SS\_PIN, RST\_PIN); // Create MFRC522 instance.

void setup()

{

Serial.begin(9600); // Initiate a serial communication

SPI.begin(); // Initiate SPI bus

mfrc522.PCD\_Init(); // Initiate MFRC522

Serial.println("Approximate your card to the reader...");

Serial.println();

}

void loop()

{

// Look for new cards

if ( ! mfrc522.PICC\_IsNewCardPresent())

{ return; }

// Select one of the cards

if ( ! mfrc522.PICC\_ReadCardSerial())

{ return; }

//Show UID on serial monitor

Serial.print("UID tag :");

String content= "";

byte letter;

for (byte i = 0; i < mfrc522.uid.size; i++)

{

Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");

Serial.print(mfrc522.uid.uidByte[i], HEX);

content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));

content.concat(String(mfrc522.uid.uidByte[i], HEX));

}

Serial.println();

Serial.print("Message : ");

content.toUpperCase();

if (content.substring(1) == "BD 31 15 2B") //change here the UID of the

card/cards that you want to give access

{

Serial.println("Authorized access");

Serial.println();

delay(3000);

}

else {

Serial.println(" Access denied");

delay(3000);

}

}

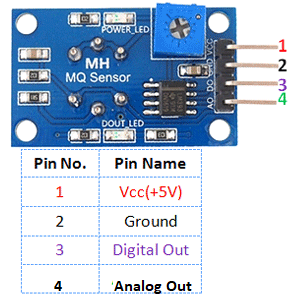
**Experiment No. 12:** Detect Gas leakage and enable alarm on detection of leakage of gas.

Aim: To understand working of Gas sensor with arduino

**Components:**

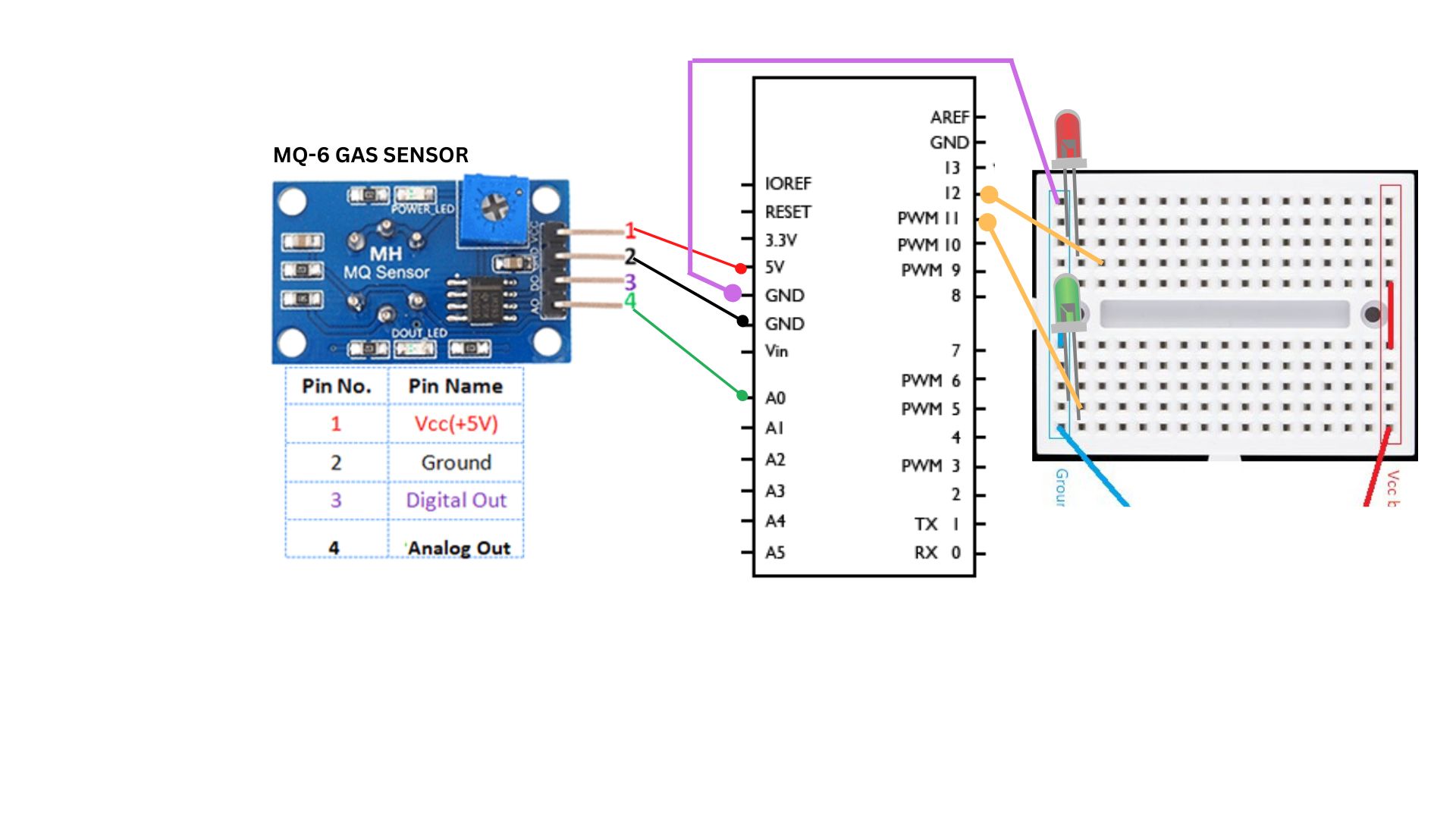
* Arduino UNO
* MQ-6 gas sensor
* Red and green LEDs
* Jumper wires and a breadboard
* Bread board

**MQ-6 Gas sensor and its pinout diagram:**

****

**Connections:**

|  |  |
| --- | --- |
| Arduino A0 | Analog Out pin of MQ-6 |
| Arduino 5V | Vcc of MQ-6 |
| Arduino GND | GND pin of MQ-6 |
| Arduino D11 | Green LED long leg  Green LED Short leg to Arduino GND |
| Arduino D12 | Red LED long leg  Red LED Short leg to Arduino GND |

****

**Program:**

int GAS\_VAL = 0;

void setup()

{

pinMode(A0, INPUT); // MQ-6 A0 Pin

Serial.begin(9600);

pinMode(11,OUTPUT); // LED Green

pinMode(12,OUTPUT); // LED Red

}

void loop()

{

GAS\_VAL = analogRead(A0);

Serial.println(GAS\_VAL);

if (GAS\_VAL > 500)

{

Serial.println(" LPG Detected ");

digitalWrite(11,HIGH);

digitalWrite(12,LOW);

}

else

{

Serial.println("LPG Not Detected ");

digitalWrite(11,LOW);

digitalWrite(12,HIGH);

}

delay(10);

}