DOCUMENTATION---

PROJECT ON INTEGATION OF TEMPERTURE SENSOR ,SMOKE SENSOR WITH LCD DISPLAY IN ARDUINO BASED PROJECT……………

CONNECTIONS………………..

LCD---ARDUINO UNO………………….

1 .GND—GND.

2 .VCC—5VOLT.

3.PIN-3----------RESISTANCE/POTENTIOMETER(Used for controlling brightness of display)---GND.

4.PIN4—DIGITAL 12.

5.PIN5—GND.

6.PIN6—DIGITAL 11.

7.PIN7,8,9,10—are not connected.

8.PIN11—DIGITAL 5.

9.PIN12—DIGITAL 4.

10.PIN13—DIGITAL 3.

11.PIN14—DIGITAL 2.

12.PIN15— VCC.

13.PIN16— GND.

DHT11-----ARDUINO UNO.

1.VCC—5 VOLT.

2.DATA—DIGITAL 6.

3.GND---GND.

SMOKE SENSOR(MQ2)-----ARDUINO UNO.

1.VCC—5 VOLT.

2.GND—GND.

3.D0—NOT CONNECTED.

4.A0---ANALOG(A0).

After all connection become completed then move on arduino ide and code will present in that formate written below……….

CODE OF TEMPERATURE SENSOR (DHT11)……………………

```
#include <LiquidCrystal.h>

#include <SimpleDHT.h>

//Declaring digital pin no 6 as the dht11 data pin

int pinDHT11 = 6;
SimpleDHT11 dht11;

//Declaring the lcd pins

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

void setup() {
// Don't forget to choose 9600 at the port screen

  Serial.begin(9600);

//Telling our lcd to start up

  lcd.begin(16, 2);


}

void loop() {

  //These serial codes are for getting readings on the port screen aswell as the LCD
display, since they'll offer us a more detailed interface


  Serial.println("=================================");
  Serial.println("DHT11 readings...");


  byte temperature = 0;
  byte humidity = 0;
  int err = SimpleDHTErrSuccess;

  //This bit will tell our Arduino what to do if there is some sort of an error at
getting readings from our sensor
  if ((err = dht11.read(pinDHT11, &temperature, &humidity, NULL)) !=
SimpleDHTErrSuccess) {
    Serial.print("No reading , err="); Serial.println(err);delay(1000);
    return;
  }

  Serial.print("Readings: ");
  Serial.print((int)temperature); Serial.print(" Celcius, ");
  Serial.print((int)humidity); Serial.println(" %");

  //Telling our lcd to refresh itself every 0.75 seconds
  lcd.clear();

  //Choosing the first line and row
  lcd.setCursor(0,0);
  //Typing Temp: to the first line starting from the first row
  lcd.print("Temp: ");
  //Typing the temperature readings after "Temp: "
  lcd.print((int)temperature);
  //Choosing the second line and first row
  lcd.setCursor(0,1);
  //Typing Humidity(%): to the second line starting from the first row
  lcd.print("Humidity(%): ");
  //Typing the humidity readings after "Humidity(%): "
  lcd.print((int)humidity);




  delay(750);
```

CODE OF SMOKE SENSOR IN LCD DISPLAY………………..

```
#include <MQ2.h>

#include <Wire.h>

#include <LiquidCrystal.h>

//I2C pins declaration

//LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);

int Analog_Input = A0;

int lpg, co, smoke;

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

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

MQ2 mq2(Analog_Input);
```

```
void setup(){
  Serial.begin(9600);
  lcd.begin(16,2);//Defining 16 columns and 2 rows of lcd display
//  lcd.backlight();
  mq2.begin();
}
void loop(){
  float* values= mq2.read(true); //set it false if you don't want to print the values in the Serial
  //lpg = values[0];
  lpg = mq2.readLPG();
  //co = values[1];
  co = mq2.readCO();
  //smoke = values[2];
  smoke = mq2.readSmoke();
  lcd.setCursor(0,0);
  lcd.print("LPG:");
  lcd.print(lpg);
  lcd.print(" CO:");
  lcd.print(co);
  lcd.setCursor(0,1);
  lcd.print("SMOKE:");
  lcd.print(smoke);
  lcd.print(" PPM");
  delay(1000);
}
```

INTEGRATED CODE OF ALL SENSOR WITH LCD…………………..

```
#include <MQ2.h>
#include <Wire.h>

#include <LiquidCrystal.h>

#include <SimpleDHT.h>

//Declaring digital pin no 6 as the dht11 data pin

int pinDHT11 = 6;
SimpleDHT11 dht11;

//Declaring the lcd pins

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

int Analog_Input = A0;
int lpg, co, smoke;
MQ2 mq2(Analog_Input);

void setup() {
  // Don't forget to choose 9600 at the port screen
  Serial.begin(9600);

  //Telling our lcd to start up

  lcd.begin(16, 2);
mq2.begin();

}

void loop() {

  //These serial codes are for getting readings on the port screen aswell as the LCD display, since they'll offer us a more detailed interface

float* values= mq2.read(true); //set it false if you don't want to print the values in the Serial
  //lpg = values[0];
  lpg = mq2.readLPG();
  //co = values[1];
  co = mq2.readCO();
  //smoke = values[2];
  smoke = mq2.readSmoke();
  lcd.setCursor(0,0);
  lcd.print("LPG : ");
  lcd.print(lpg);
  //lcd.print(" CO:");
  //lcd.print(co);
  lcd.setCursor(0,1);
  lcd.print("SMOKE:");
  lcd.print(smoke);
  lcd.print(" PPM    ");
  delay(1000);

  //CODE FOR TEMPERATURE SENSOR..............................
  //Serial.println("================================");
  //Serial.println("DHT11 readings...");



  byte temperature = 0;
  byte humidity = 0;
    int err = SimpleDHTErrSuccess;

//  This bit will tell our Arduino what to do if there is some sort of an error at getting readings from our sensor
    if ((err = dht11.read(pinDHT11, &temperature, &humidity, NULL)) != SimpleDHTErrSuccess) {
  Serial.print("No reading , err="); Serial.println(err); delay(1000);
  return;
}

Serial.print("Readings: ");
Serial.print((int)temperature); Serial.print(" Celcius, ");
Serial.print((int)humidity); Serial.println(" %");
```

```
//Telling our lcd to refresh itself every 0.75 seconds
//lcd.clear();


//Choosing the first line and row
lcd.setCursor(0, 0);
//Typing Temp: to the first line starting from the first row
lcd.print("Temp: ");
//Typing the temperature readings after "Temp: "
lcd.print((int)temperature);
//Choosing the second line and first row
lcd.setCursor(0, 1);
//Typing Humidity(%): to the second line starting from the first row
lcd.print("Humidity(%): ");
//Typing the humidity readings after "Humidity(%): "
lcd.print((int)humidity);




delay(750);
}
```

Ending of this project…………………………………….