IBM ASSIGNMENT – 1

DHANESH K - 2019504514

CIRCUIT DIAGRAM

CODE:

#include <LiquidCrystal.h>

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

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

const int PIR = 8;

const int bulb = 7;

const int tempPin = A1;

const int fan = 10;

int const PINO\_SGAS = A0;

int temp;

int tempMin = 30;

int tempMax = 60;

int fanSpeed;

int PIRState = 0;

int trigger\_pin = 13;

int echo\_pin = 6;

int buzzer\_pin = 9;

int time;

int distance;

void setup() {

pinMode(PIR, INPUT);

pinMode(bulb, OUTPUT);

pinMode(fan, OUTPUT);

pinMode(tempPin, INPUT);

lcd.begin(16, 2);

Serial.begin (9600);

pinMode (trigger\_pin, OUTPUT);

pinMode (echo\_pin, INPUT);

pinMode (buzzer\_pin, OUTPUT);

}

void loop()

{

//Using PIR to automate lights

PIRState = digitalRead(PIR);

if (PIRState == HIGH)

{

digitalWrite(bulb, HIGH);

}

if (PIRState == LOW)

{

digitalWrite(bulb, LOW);

}

if

(temp = readTemp());

if (temp < tempMin)

{

fanSpeed = 0;

analogWrite(fan, fanSpeed);

digitalWrite(fan, LOW);

}

if ((temp >= tempMin) && (temp <= tempMax))

{

fanSpeed = temp;

fanSpeed = 1.5 \* fanSpeed;

analogWrite(fan, fanSpeed);

}

lcd.setCursor(0, 0);

lcd.print("TEMP:");

lcd.print(temp);

lcd.print(" C ");

delay(200);

int color = analogRead(PINO\_SGAS);

lcd.setCursor(0,1);

//lcd.print("");

if(color <= 85){

lcd.print("Gas is Low ");

} else if(color <= 120){

lcd.print("Gas is Med ");

} else if(color <= 200){

lcd.print("Gas is High ");

} else if(color <= 300){

lcd.print("Gas is Ext ");

}

delay(250);

digitalWrite (trigger\_pin, HIGH);

delayMicroseconds (10);

digitalWrite (trigger\_pin, LOW);

time = pulseIn (echo\_pin, HIGH);

distance = (time \* 0.034) / 2;

if (distance <= 10)

{

Serial.println (" Door Open ");

Serial.print (" Distance= ");

Serial.println (distance);

digitalWrite (buzzer\_pin, HIGH);

delay (500);

}

else {

Serial.println (" Door closed ");

Serial.print (" Distance= ");

Serial.println (distance);

digitalWrite (buzzer\_pin, LOW);

delay (500);

}

}

int readTemp()

{

temp = analogRead(tempPin);

return temp \* 0.48828125;

}

