|  |
| --- |
| SMOKE DETECTION ALARM SYSTEM AND REMOTE EXHAUSTING FAN |
|  |
|  |
| July 15  Authored by: LAHIRU MADHUSHANKA  2017/CSC/025 |



|  |
| --- |
| Introduction A **smoke detector** is a device that senses smoke, typically as an indicator of [fire](https://en.wikipedia.org/wiki/Fire). Commercial security devices issue a signal to a [fire alarm control panel](https://en.wikipedia.org/wiki/Fire_alarm_control_panel) as part of a [fire alarm system](https://en.wikipedia.org/wiki/Fire_alarm_system), while household smoke detectors, also known as **smoke alarms**, generally issue a local audible or visual [alarm](https://en.wikipedia.org/wiki/Alarm) from the detector itself or several detectors if there are multiple smoke detectors interlinked.  What happens in this system is that when a toxic chemical or flammable gas entries the house or storeroom or hotel where the system installed, It sends the signal to an LCD display, Red light and warning alarm for alert authorized people.    The advantage of this system has a remote control system. The remote control then activates an exhaust fan to remove toxin gases from the system, Then it can be minimizing the damage.    After reducing the spreading gas into the room system going to be normal.After displaying a safe message in LCD display and activate the green light and automatically off the exhausting fan |
|  |
| * **Component List**      * **Design**      * **Pseudocoda**     **BEGIN**  Initialize pin 7 to IR sensor  Initialize receiver object as IR\_PIN  Initialize decode\_results results  Initialize pin 2 to redLed light  Initialize pin 3 to greenLed light  Initialize pin 4 to buzzer  Initialize pin A0(Analog) to gas\_sensor  Initialize sensor threshold value 350 to sensorThresh  Initialize pin 12 to fan(DC motor)  Function setup  set pin 2 as OUTPUT  set pin 3 as OUTPUT  set pin 4 as OUTPUT  set pin 12 as OUTPUT  set pin A0 as INPUT    Begin serial communication  Begin LCD process  Begin the receiving process  End Function  Loop  Initialize analogRead(gas\_sensor) to analogValue    IF(analogValue > sensorThresh )THEN  Activate red LED  Deactivate green LED  Activate BUZZER  Clear LCD display  Set LCD display cursor from (0,0)  Display “WARNING”  Wait 1000 milliseconds  Clear LCD display  Set LCD display cursor from (0,1)  Display “GAS DETECTING”  Wait 1000 milliseconds  IF(irrecv.decode(&results = true)THEN  Display receive value in decimal on serial monitor  IF(results.value = 16580863)THEN  Activate the fan  ENDIF  Ready to receive next value  ENDIF  ELSE  Activate green LED  Deactivate red LED  Deactivate BUZZER  Clear LCD dispay  Set LED display cursor from (0,0)  Display “SAFE”  Wait 1000 milliseconds  Clear LCD display  Set LCD display cursor from (0,1)  Display “ALL CLEAR”  Wait 1000 milliseconds  Deactivate the fan  Repeat LOOP  END   * Sketch     #include <IRremote.h>  #include <LiquidCrystal.h>  LiquidCrystal lcd(5,6,8,9,10,11);  int IR\_PIN = 7;  IRrecv irrecv(IR\_PIN);  decode\_results results;  int redLed = 2;  int greenLed = 3;  int buzzer = 4;  int gas\_sensor = A0;  int sensorThresh = 350;  int fan=12;  void setup(){  pinMode(redLed, OUTPUT);  pinMode(greenLed,OUTPUT);  pinMode(buzzer,OUTPUT);  pinMode(fan,OUTPUT);  pinMode(gas\_sensor,INPUT);  Serial.begin(9600);  lcd.begin(16,2);  irrecv.enableIRIn();  }  void loop(){  int analogValue = analogRead(gas\_sensor);    if(analogValue>sensorThresh){  digitalWrite(redLed,HIGH);  digitalWrite(greenLed,LOW);  digitalWrite(buzzer,HIGH);  lcd.clear();  lcd.setCursor(0,0);  lcd.print("WARNING");  delay(1000);  lcd.clear();  lcd.setCursor(0,1);  lcd.print("GAS DETECTING");  delay(1000);  if (irrecv.decode(&results)) {  Serial.println(results.value);  if (results.value == 16580863){  digitalWrite(fan,HIGH);  }  irrecv.resume();  }  }else{  digitalWrite(greenLed,HIGH);  digitalWrite(redLed,LOW);  digitalWrite(buzzer,LOW);  lcd.clear();  lcd.setCursor(0,0);  lcd.print("SAFE");  delay(1000);  lcd.clear();  lcd.setCursor(0,1);  lcd.print("ALL CLEAR");  delay(1000);  digitalWrite(fan,LOW);  }  }    **END OF THE REPORT**  **THANK YOU!** |