

Report – Semester Project  
Internet of Things

Student: ***Nikita Ferents***

KLAIPĖDA

2023

**Smoke Detector using Arduino and TinkerCAD**

Smoke Detectors arevery useful in detecting smoke or fire in buildings, and so are the important safety parameters. So I have build a **Smoke Detector Circuit** which not only sense the smoke in the air but also reads and **displays the level of Smoke in the Air in PPM** (parts per million). This circuit triggers the Buzzer when Smoke level becomes higher than 100 ppm, increasing rapidity and changing the patter of the beeper the more particle it gets. This threshold value can be changed in the Code according to the requirement.

Here are some examples of where it can be used:

**Homes:**

 A DIY gas detector can help to detect and prevent carbon monoxide poisoning, which is a leading cause of accidental death. Carbon monoxide is a colorless, odorless gas that can be produced by incomplete combustion of fuel. It can build up in homes and businesses if there is a problem with the ventilation or if there is a leak in a gas appliance.

**Industrial settings:**

A DIY gas detector can be used to detect leaks of flammable or toxic gases in industrial settings. This can help to prevent accidents and protect workers from exposure to dangerous gases.

**Scientific research:**

With a benefit of a screen with precise particles amount and the custom signals for the different concentration. A DIY gas detector can be used for scientific research to measure the concentration of gases in the air. This can be used to study atmospheric composition or to track the release of pollutants.

Here are some examples of who could benefit from using a DIY gas detector:

**Homeowners:** A DIY gas detector is a relatively inexpensive way to add an extra layer of safety to your home.

**Business owners:** A DIY gas detector can help to protect your employees from exposure to dangerous gases and help to prevent accidents.

**Scientists:** A DIY gas detector can be a valuable tool for scientific research.

Here are some of the benefits of using a DIY gas detector:

**It is relatively inexpensive**. You can purchase all of the components for a DIY gas detector for an up to 100 euros.

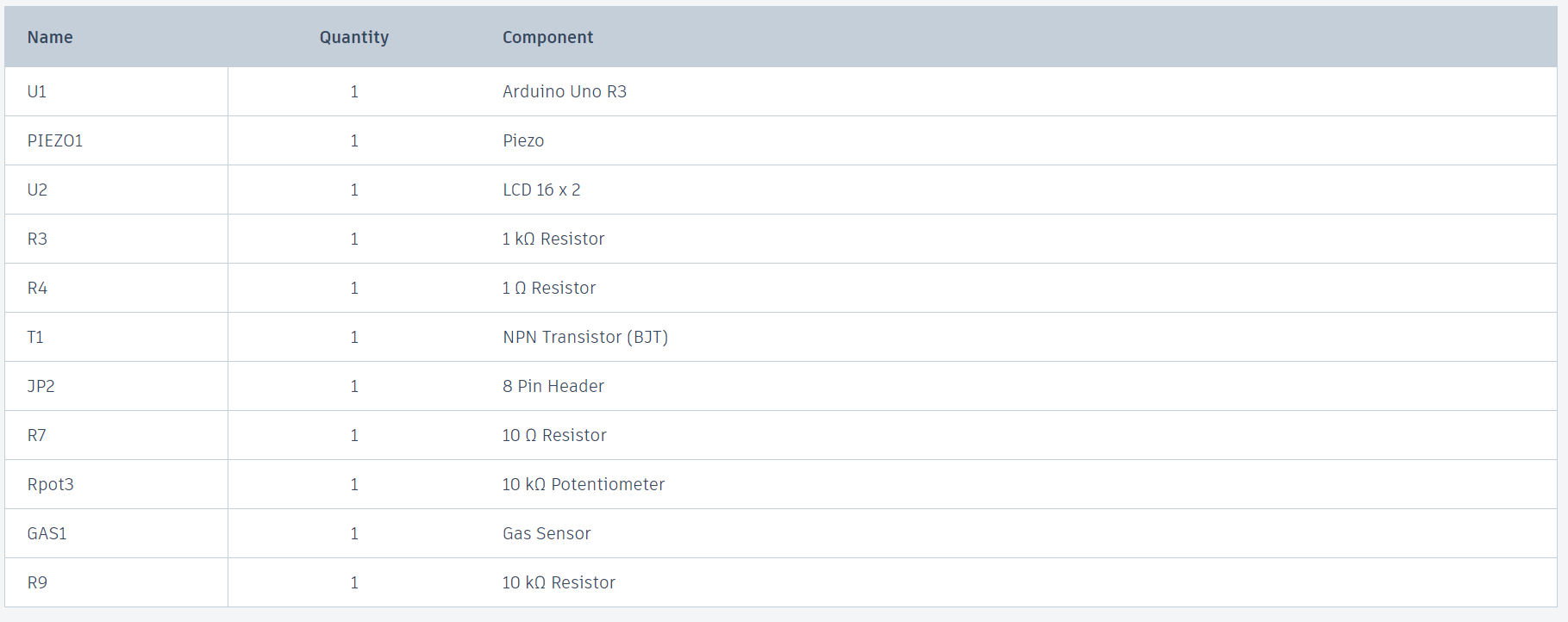
**It is easy to build.** The circuit for a DIY gas detector is relatively simple and can be assembled in a few hours.

**It is portable.** You can easily take a DIY gas detector with you wherever you go.

**It is versatile.** A DIY gas detector can be used to detect a variety of gases.

**Components Required:**

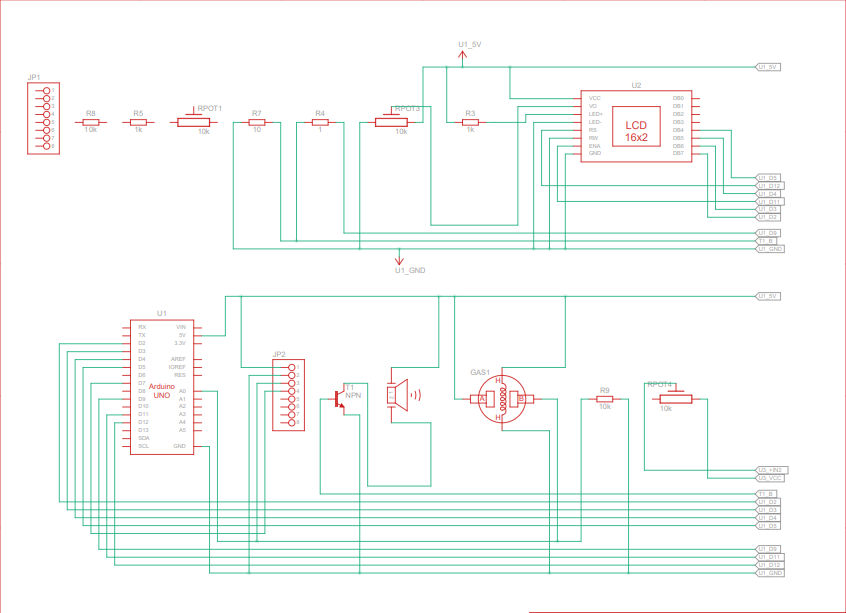
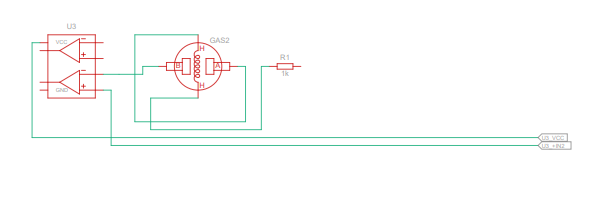
* Smoke Sensor
* Resistors (10K and 1K)
* Buzzer
* 16x2 LCD
* 10k POT
* TinkerCAD
* Arduino UNO module



**Circuit Explanation:**

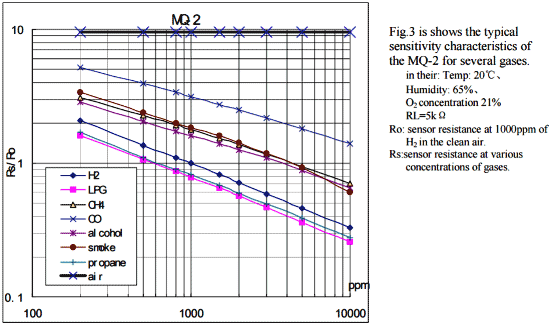
In this Smoke Detector Circuit with Arduino, we have used a  **Gas Sensor** to detect preset smoke in the air. A **16x2 LCD is used for displaying the PPM value of Smoke**. A **buzzer** is placed as an alarm which gets triggered when smoke level goes beyond 100 PPM and changes the pattern of the sound with the increasing level of danger.

Circuit connections for this project is relatively simple, we have a **Comparator Circuit** for comparing output voltage of smoke sensor with preset voltage (output connected at pin D7). Also smoke sensor output is connected at an analog pin of Arduino (A0). Buzzer is connected at Pin D9. And LCD connections are same as Arduino LCD examples that are available in Arduino IDE (12, 11, 5, 4, 3, 2). Remaining connections are shown in the circuit diagram.

**Programming Explanation:**

Program of this project is little difficult to make. User needs to read Smoke Sensor datasheet. In this we have to read slop or curve of smoke concentration in air with respect to clean air. After reading datasheet, we get some values that we will need in the Code to calculate ppm of smoke in air. Here mostly we need curve values (we take two points from the curve), sensor resistance (will be calculated in code), clean air constant (9.83) and Load Resistance (I used 10K). We can find curve values from the datasheet and we can put load resistance 5k-54k and then we will calculate sensor resistance by these value and smoke samples.



Take two points from the curve and take log of them like point one: (lg200, lg3.4)=(2.3,0.53) and  point two: (lg10000,lg0.63)=(4,-0.20). Then find slope of the curve using Formula: (y2-y1)/(x2-x1), then take one point and slope (-0.44) and use them in the Program (x, y, slope). Further check the Code below to understand the Calculation.

First we need to include header file for LCD and define pins for the same. Then define curve values and load resistance.

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

#define buzzer 9

#define sensor A0

#define load\_Res 10

#define air\_factor 9.83

float SmokeCurve[3] ={2.3,0.53,-0.44};

float Res=0;

Now in *void setup()*, we need to Calibrate the module by using *SensorCalibration* function:

void setup()

{

lcd.begin(16,2);

lcd.print("Calibrating.....");

Res = SensorCalibration();

lcd.print("Calibration done.");

lcd.setCursor(0,1);

lcd.print("Res=");

lcd.print(Res);

lcd.print("kohm");

delay(2000);

lcd.clear();

pinMode(buzzer, OUTPUT);

}

float SensorCalibration()

{

int i;

float val=0;

val=resistance(50,500);

val = val/air\_factor;

return val;

}

Then in *void loop()* function, we have calculated the PPM of smoke by using *resistance* Function:

void loop()

{

lcd.setCursor(0,0);

lcd.print("SMOKE:");

float res=resistance(5,50);

res/=Res;

int result=pow(10,(((log(res)-SmokeCurve[1])/SmokeCurve[2]) + SmokeCurve[0]))\* 100;

lcd.print(result);

lcd.print( " ppm ");

if(result>15000)

{

tone(buzzer, 500);

delay(1000);

noTone(buzzer);

}

else if(result>4000)

{

tone(buzzer, 500);

delay(200);

noTone(buzzer);

delay(1000);

}

else if(result>100)

{

tone(buzzer, 500);

delay(100);

noTone(buzzer);

delay(2000);

}

else

{

noTone(buzzer);

delay(500);

}}

float resistance(int samples, int interval)

{

int i;

float res=0;

for (i=0;i<samples;i++)

{

int adc\_value=analogRead(sensor);

res+=((float)load\_Res\*(1023-adc\_value)/adc\_value);

delay(interval);

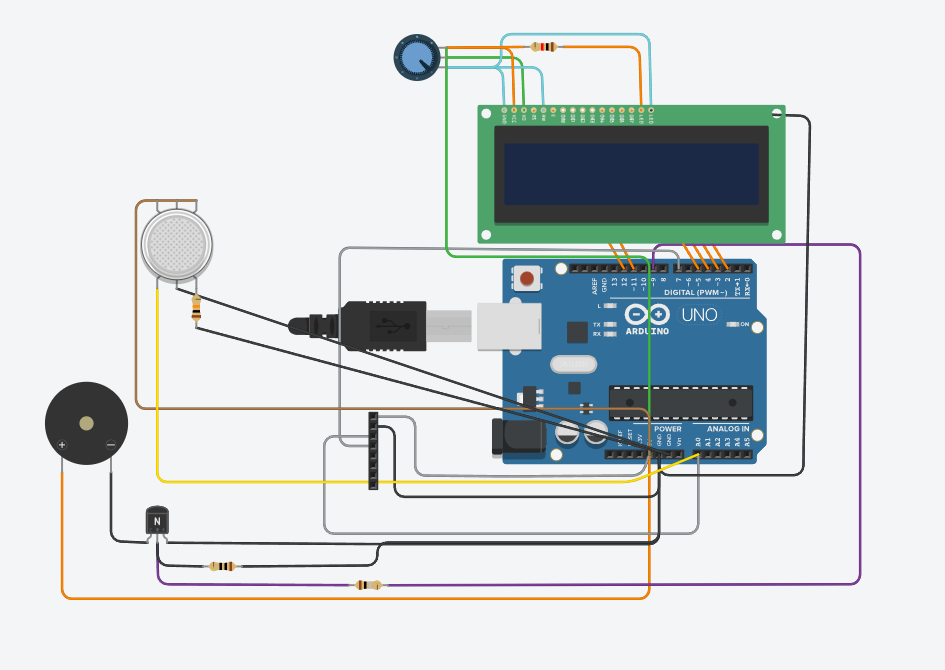
}

res/=samples;

return res;

}

The look of the project:



**Summary**

My DIY smoke detector a device capable of detecting smoke and generating an audible alarm. Also, it has a informative LCD with all needed information. Additionaly, you can adjust the brightness of this screen. It is able to help any user with the home safety excluding the problem with the fire and air pollution in the home.