7/27/2020

DLITHE INTERNSHIP

PROJECT WORK-2

TOPIC:ALCOHOL DETECTION

BHARATH D
Email: 4nm18ec038@nmamit.in / bharathdnayar@gmail.com
Contact : 7899216194

Table of Contents

1.INTRODUCTION:

In this project, we will go over how to build an alcohol sensor with an arduino.

The alcohol sensor we will use is the MQ-3 sensor. This is a sensor that is not only sensitive to alcohol, particularly ethanol, which is the type of alcohol which is found in wine, beer, and liquor.

This type of sensor circuit can be used as a breathalyzer to check a person's blood alcohol level. Just as we exhale carbon dioxide when we breathe out, we also will breathe out some alcohol if we have alcohol in our blood. Any alcometer device can measure this alcohol content.

The more ethanol in your blood, the more there is in the air on exhalation. This alcohol content gives a good indication for if a person is drunk and how drunk they are.

The amount of alcohol exhaled into the air is proportional to the amount of alcohol which will be found in a person's blood. Alcometers use a built-in formula to estimate blood alcohol content from exhaled air alcohol content.

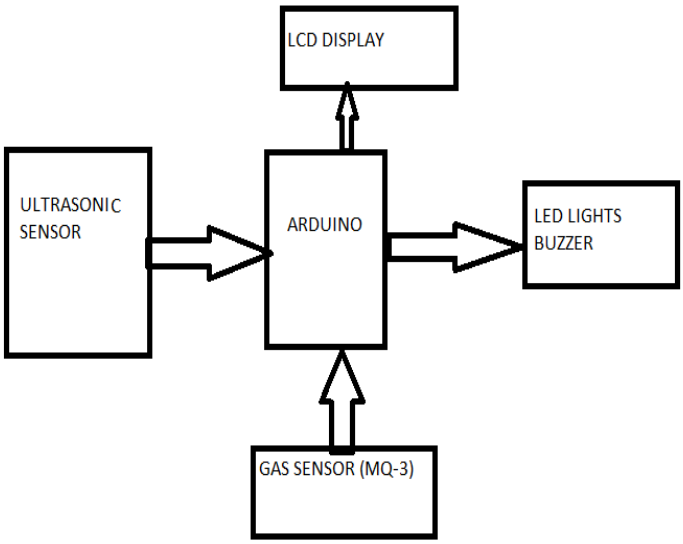
For different countries, the level of alcohol in the blood that defines a person as over the limit for driving varies. The range ranges from 0.01 to 0.10. Most countries have a limit of about 0.05. For example, Greece, Greenalnd, and Iceland all have limits of 0.05. Canada has a higher limit set at 0.08. In the United States, it is also 0.08. This means that if the alcometer reading measures above this, the person can receive a DUI.

For our circuit, it can function as an alcometer so that we get an estimate of a person's blood alcohol level.

To connect the sensor, there are 4 leads. 2 of them are for power. The +5V terminal of the sensor connects into the 5V terminal of the arduino board. The GND terminal of the sensor connects into the GND terminal of the arduino. This establishes power for the sensor.

The other 2 connections are the analog and digital output of the sensor. These connect to analog pin A0 and digital pin D8, respectively.

2.DESIGN / BLOCK DIAGRAM:



3.CIRCUIT DIAGRAM:

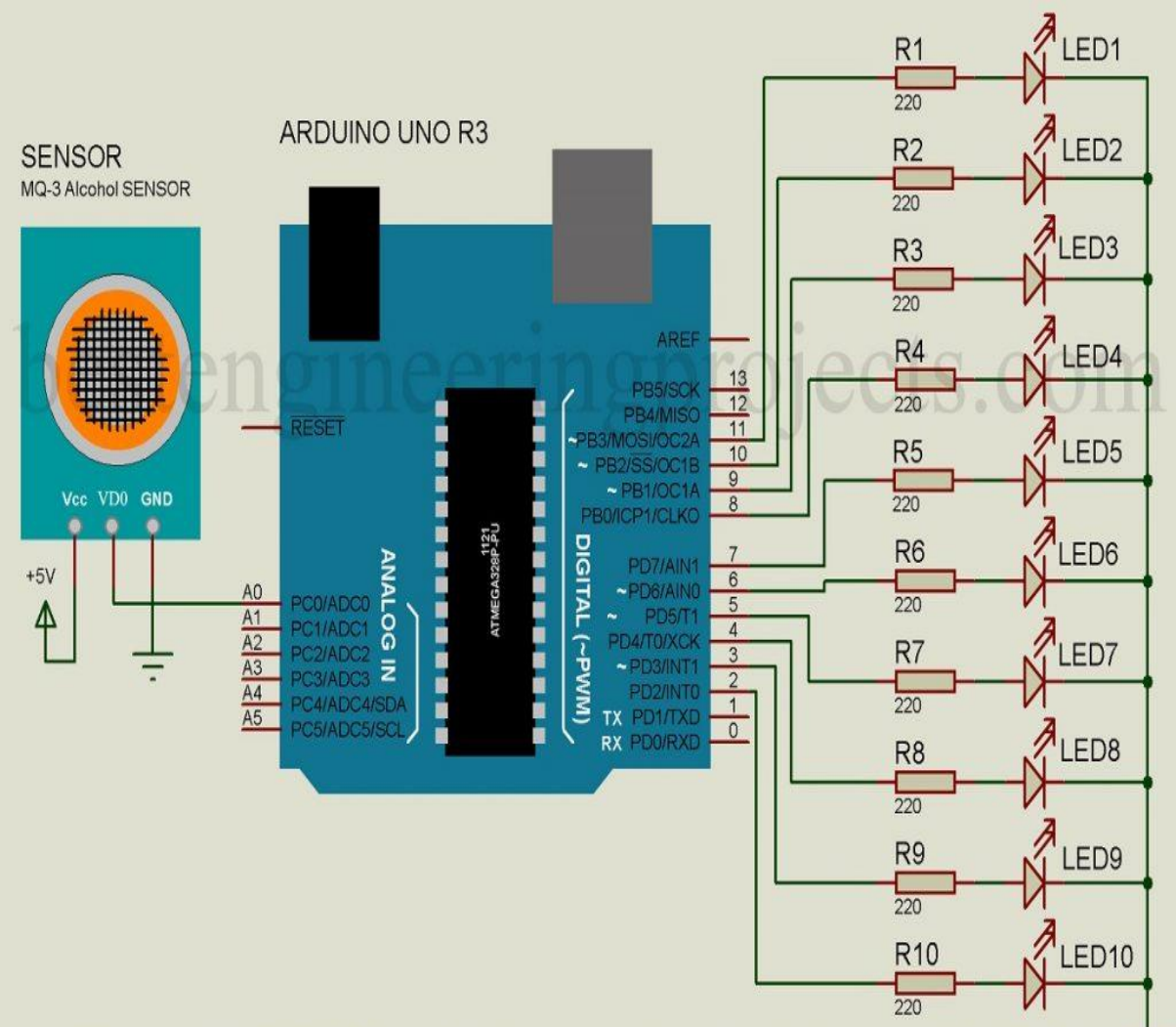
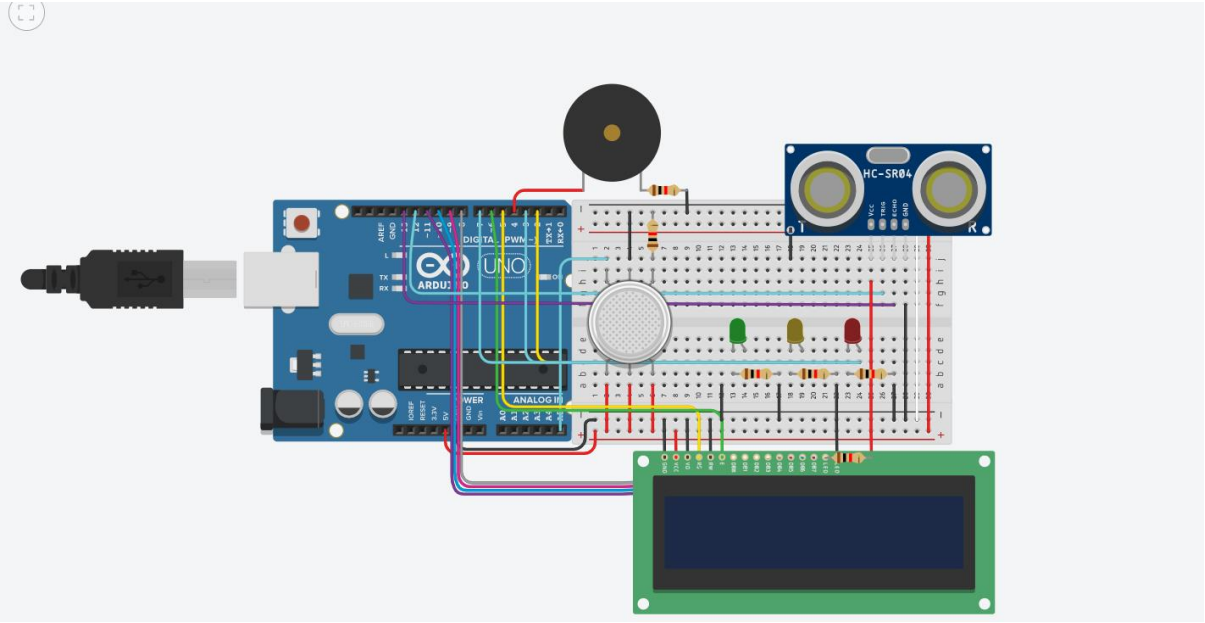


Figure 1: DIY Breathalyzer using Arduino and MQ-3

4.SIMULATION CIRCUIT:



5.COMPONENTS LIST

Name	Quantity	Component
U1	1	Arduino Uno R3
GAS1	1	Gas Sensor
R1 R2 R6 R7 R8 R11	6	1 kΩ Resistor
U2	1	LCD 16 x 2
D3	1	Green LED
D4	1	Yellow LED
D5	1	Red LED
DIST2	1	Ultrasonic Distance Sensor
PIEZO1	1	Piezo

WORKING:

In this project ultrasonic distance sensor is used . Which determines the distance of the person from the sensor . If the distance is less then particular value then it allow MQ-3 gas sensor to detect the amount of alcohol present in the breath . If the distance less then particular value then we can here the buzzer sound. It has a lcd display which notify the amount of alcohol present in breath in the scale of 10. It has 3 led's (green, yellow, red) ,if alcohol content is less then 2 green led will glow. If alcohol content is less then 5 yellow led along with green led will glow. If alcohol content is less then or equal to 10 red ,yellow, green led's will glow.

6.ABOUT SOME COMPONENTS:

GAS SENSORS:

Gas sensors are typically classified into various types based on the type of the sensing element it is built with. Below is the classification of the various types of gas sensors based on the sensing element that are generally used in various applications:

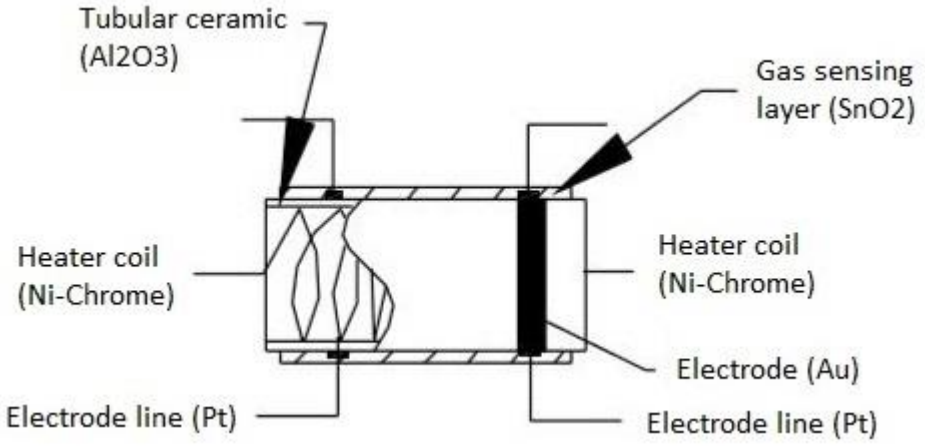
- Metal Oxide based gas Sensor.
- Optical gas Sensor.
- Electrochemical gas Sensor.
- Capacitance-based gas Sensor.
- Calorimetric gas Sensor.
- Acoustic based gas Sensor.

Gas Sensor Construction

Of all the above-listed types, the most commonly used gas sensor is the Metal oxide semiconductor based gas sensor. All Gas sensors will consist of a sensing element which comprises of the following parts.

1. Gas sensing layer
2. Heater Coil
3. Electrode line
4. Tubular ceramic
5. Electrode

The below image illustrates the parts present in a metal oxide gas sensor



The purpose of each of these elements is as below:

Gas sensing layer: It is the main component in the sensor which can be used to sense the variation in the concentration of the gases and generate the change in electrical resistance. The gas sensing layer is basically a chemiresistor which changes its resistance value based on the

The concentration of particular gas in the environment. Here the sensing element is made up of a Tin Dioxide (SnO2) which is, in general, has excess electrons (donor element). So whenever toxic gases are being detected the resistance of the element changes and the current flown through it varies which represents the change in concentration of the gases.

Heater coil: The purpose of the heater coil is to burn-in the sensing element so that the sensitivity and efficiency of the sensing element increases. It is made of Nickel-Chromium which has a high melting point so that it can stay heated up without getting melted.

Electrode line: As the sensing element produces a very small current when the gas is detected it is more important to maintain the efficiency of carrying those small currents. So Platinum wires come into play where it helps in moving the electrons efficiently.

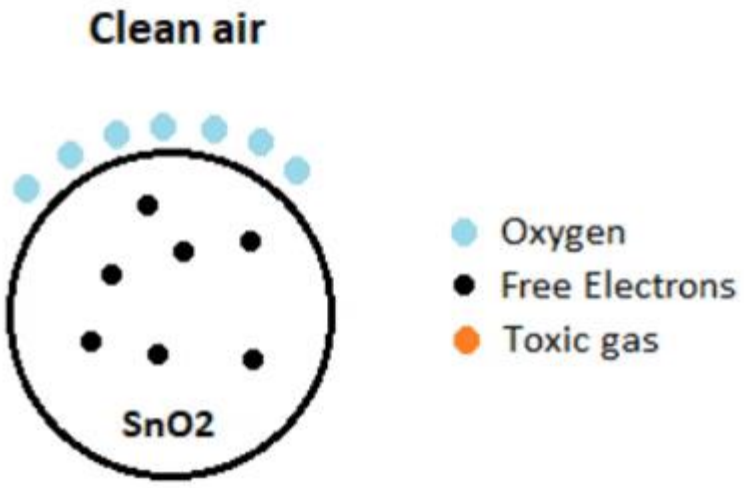
Electrode: It is a junction where the output of the sensing layer is connected to the Electrode line. So that the output current can flow to the required terminal. An electrode here is made of Gold (Au –Aurum) which is a very good conductor.

Tubular ceramic: In between the Heater coil and Gas sensing layer, the tubular ceramic exists which is made of Aluminum oxide (Al2O3). As it has high melting point, it helps in maintaining the burn-in (preheating) of the sensing layer which gives the high sensitivity for the sensing layer to get efficient output current.

Mesh over the sensing element: In order to protect the sensing elements and the setup, a metal mesh is used over it, which is also used to avoid/hold the dust particles entering into the mesh and prevent damaging the gas sensing layer from corrosive particles.

Gas Sensor Working

The ability of a Gas sensor to detect gases depends on the **chemiresister** to conduct current. The most commonly used chemiresistor is Tin Dioxide (SnO2) which is an n-type semiconductor that has free electrons (also called as donor). Normally the atmosphere will contain more oxygen than combustible gases. The oxygen particles attract the free electrons present in SnO2 which pushes them to the surface of the SnO2. As there are **no free electrons** available output current will be zero. The below gif shown the oxygen molecules (blue color) attracting the free electrons (black color) inside the SnO2 and preventing it from having free electrons to conduct current.



When the sensor is placed in the toxic or combustible gases environment, this reducing gas (orange color) reacts with the adsorbed oxygen particles and breaks the chemical bond between oxygen and free electrons thus **releasing the free electrons**. As the free electrons are back to its initial position they can now conduct current, this conduction will be proportional the amount of free electrons available in SnO2, if the gas is highly toxic more free electrons will be available.

List of Gas Sensors and What Gases They Sense

Sensor Name	Gas to measure
MQ-2	Methane, Butane, LPG, Smoke
MQ-3	Alcohol, Ethanol, Smoke
MQ-4	Methane, CNG Gas
MQ-5	Natural gas, LPG
MQ-6	LPG, butane
MQ-7	Carbon Monoxide
MQ-8	Hydrogen Gas
MQ-9	Carbon Monoxide, flammable gasses
MQ131	Ozone
MQ135	Air Quality
MQ136	Hydrogen Sulphide gas
MQ137	Ammonia
MQ138	Benzene, Toluene, Alcohol, Propane, Formaldehyde gas, Hydrogen
MQ214	Methane, Natural Gas
MQ216	Natural gas, Coal Gas
MQ303A	Alcohol, Ethanol, smoke
MQ306A	LPG, butane
MQ307A	Carbon Monoxide

MQ309A	Carbon Monoxide, flammable gas

ULTRASONIC SENSOR:

An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves, and converts the reflected sound into an electrical signal. Ultrasonic waves travel faster than the speed of audible sound (i.e. the sound that humans can hear). Ultrasonic sensors have two main components: the transmitter (which emits the sound using piezoelectric crystals) and the receiver (which encounters the sound after it has travelled to and from the target).

In order to calculate the distance between the sensor and the object, the sensor measures the time it takes between the emission of the sound by the transmitter to its contact with the receiver. The formula for this calculation is $D = ½ T \times C$ (where D is the distance, T is the time, and C is the speed of sound ~ 343 meters/second). For example, if a scientist set up an ultrasonic sensor aimed at a box and it took 0.025 seconds for the sound to bounce back, the distance between the ultrasonic sensor and the box would be:

D = 0.5 x 0.025 x 343

or about 4.2875 meters.

ARDUINO:

The Arduino Uno is a microcontroller board based on the ATmega328. It has 20 digital input/output pins (of which 6 can be used as PWM outputs and 6 can be used as analog inputs), a 16 MHz resonator, a USB connection, a power jack, an in-circuit system programming (ICSP) header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features an ATmega16U2 programmed as a USB-to-serial converter. This auxiliary microcontroller has its own USB bootloader, which allows advanced users to reprogram it.

The Arduino has a large support community and an extensive set of support libraries and hardware add-on “shields” (e.g. you can easily make your Arduino wireless with our Wixel shield), making it a great introductory platform for embedded electronics.

The USB controller chip changed from ATmega8U2 (8K flash) to ATmega16U2 (16K flash). This does not increase the flash or RAM available to sketches.

Three new pins were added, all of which are duplicates of previous pins. The I2C pins (A4, A5) have been also been brought out on the side of the board near AREF. There is a IOREF pin next to the reset pin, which is a duplicate of the 5V pin.

The reset button is now next to the USB connector, making it more accessible when a shield is used.

7.CODE:

```
//libraries used in this code//

#include<LiquidCrystal.h>

LiquidCrystal lcd(5,6,8,9,10,11);//assigning the digital pin number which connected to lcd

long duration;//declaring the variable as LONG

int distance;//declaring variable as int

int safetyDistance;//declaring variable as int

int trigPin = 12;//declaring the variable as int and tells it is connected to digitalpin 12

int echoPin = 13;//declaring the variable as int and tells it is connected to digitalpin 13

int red2 = 2;//declaring the variable as int and tells it is connected to digitalpin 2

int red3 = 3;//declaring the variable as int and tells it is connected to digitalpin 3
```

```
int red4 = 7;//declaring the variable as int and tells it is connected to digitalpin 7
int analogValue=0;//initializing the variable as 0
int buzzer =4;//declaring the variable as int and tells it is connected to digitalpin 4
void setup()
{
  pinMode(red2, OUTPUT);//declaring variable as output
  pinMode(red3, OUTPUT);//declaring variable as output
  pinMode(red4, OUTPUT);//declaring variable as output
pinMode(buzzer, OUTPUT);//declaring variable as output
pinMode(trigPin, OUTPUT);//declaring variable as output
pinMode(echoPin, INPUT);//declaring variable as output
pinMode(A5, INPUT);//declaring variable as input
Serial.begin(9600);//command to bein serial monitor with data trasffering rate of 9600
lcd.begin(16,2);//tells lcd to bgin as 16*2
}
void loop()
{
analogValue=analogRead(A5);//assigning the analog value of A5 into new variable
Serial.println(analogValue);//print the value in serial monitor
  digitalWrite(trigPin, LOW);//to send signal from ultasonic sensors trig pin first we have to set it as low
delayMicroseconds(2);//which gives 2us delay
digitalWrite(trigPin, HIGH);//which emitts the signal
delayMicroseconds(10);//which gives delay
digitalWrite(trigPin, LOW);//stops sending the signal
duration = pulseIn(echoPin, HIGH);//pulsein is a function which helps directly getting analog value
distance= duration*0.034/2;//formula to calculate distance
safetyDistance=distance;//assigning the vale to new variable
if(safetyDistance<=10)
{
  tone(buzzer,1000,10000);//if safety distance less than 10 controll flows inside if statement
```

```
if(analogValue>80&&analogValue<150)//the value of variable in the range of  80 to 150
{lcd.setCursor(7,1);
 digitalWrite(red2, HIGH);//first light will glow
 digitalWrite(red3, LOW);
 digitalWrite(red4, LOW);//lcd shows a message
    lcd.print(" ");//clear screen
//lcd.clear();
lcd.setCursor(0,0);//sets cursor at the specified position
lcd.print("ALCOHOL CONTENT");
//lcd.clear();
lcd.setCursor(0,1);
lcd.print("2in scale of  10");
//delay(1000);
}
else if(analogValue>150&&analogValue<300)//if value of variable lies in the range 150 to 300
{ digitalWrite(red2, HIGH);//2 led will glow with suitable message
 digitalWrite(red3, HIGH);
 digitalWrite(red4, LOW);
  lcd.setCursor(7,1);
    lcd.print(" ");
//lcd.clear();
lcd.setCursor(0,0);
lcd.print("ALCOHOL CONTENT");
//delay(1000);
//lcd.clear();
lcd.setCursor(0,1);
lcd.print("5in scale of  10");
//delay(1000);
}
  else if(analogValue>300&&analogValue<500)//if value of variable lies in the scale of 300 to 500
{ digitalWrite(red2, HIGH);// all led will glow by giving suitable message
```

```
 digitalWrite(red3, HIGH);
 digitalWrite(red4, HIGH);
lcd.setCursor(7,1);
   lcd.print(" ");
lcd.setCursor(0,0);
lcd.print("ALCOHOL CONTENT");
//delay(1000);
//lcd.clear();
lcd.setCursor(0,1);
lcd.print("10in scale of 10");
//delay(1000);
}

  else//if any other value detected no led will glows
{
//lcd.clear();
   lcd.setCursor(7,1);
lcd.setCursor(0,0);
lcd.print("SAFE");
//delay(1000);
//lcd.clear();
lcd.setCursor(0,1);
lcd.print("NO ALCOHOL detected");
//delay(1000);
}
}
  else//if the distance more than 10 then no led will glow
  {
    //lcd.clear();//gives suitable message
    noTone(buzzer);
    lcd.setCursor(0,0);
```

```
    lcd.print("SAFE          ");
    lcd.setCursor(0,1);
    lcd.print("NO ALCOHOL DETECTED");
  }
}
```

8.LIBRARY USED AND LINK

LiquidCrystal library:

<LiquidCrystal.h>

This library allows an Arduino board to control LiquidCrystal displays (LCDs) based on the Hitachi HD44780 (or a compatible) chipset, which is found on most text-based LCDs. The library works with in either 4- or 8-bit mode (i.e. using 4 or 8 data lines in addition to the rs, enable, and, optionally, the rw control lines).

To use this library

#include <LiquidCrystal.h>

- Autoscroll: Shift text right and left.
- Blink: Control of the block-style cursor.
- Cursor: Control of the underscore-style cursor.
- Display: Quickly blank the display without losing what's on it.
- Hello World: Displays "hello world!" and the seconds since reset.
- Scroll: Scroll text left and right.
- Serial Display: Accepts serial input, displays it.
- Set Cursor: Set the cursor position.
- Text Direction: Control which way text flows from the curso

Function

- LiquidCrystal()
- begin()
- clear()
- home()
- setCursor()
- write()
- print()
- cursor()
- noCursor()
- blink()
- noBlink()
- display()

- noDisplay()
- scrollDisplayLeft()
- scrollDisplayRight()
- autoscroll()
- noAutoscroll()
- leftToRight()
- rightToLeft()
- createChar()

Link to download LiquidCrystal.h:

http://downloads.arduino.cc/libraries/github.com/arduino-libraries/LiquidCrystal-1.0.7.zip

Tinker cad link: https://www.tinkercad.com/things/b6KtloQvwjT-alcohol-detector

Some images