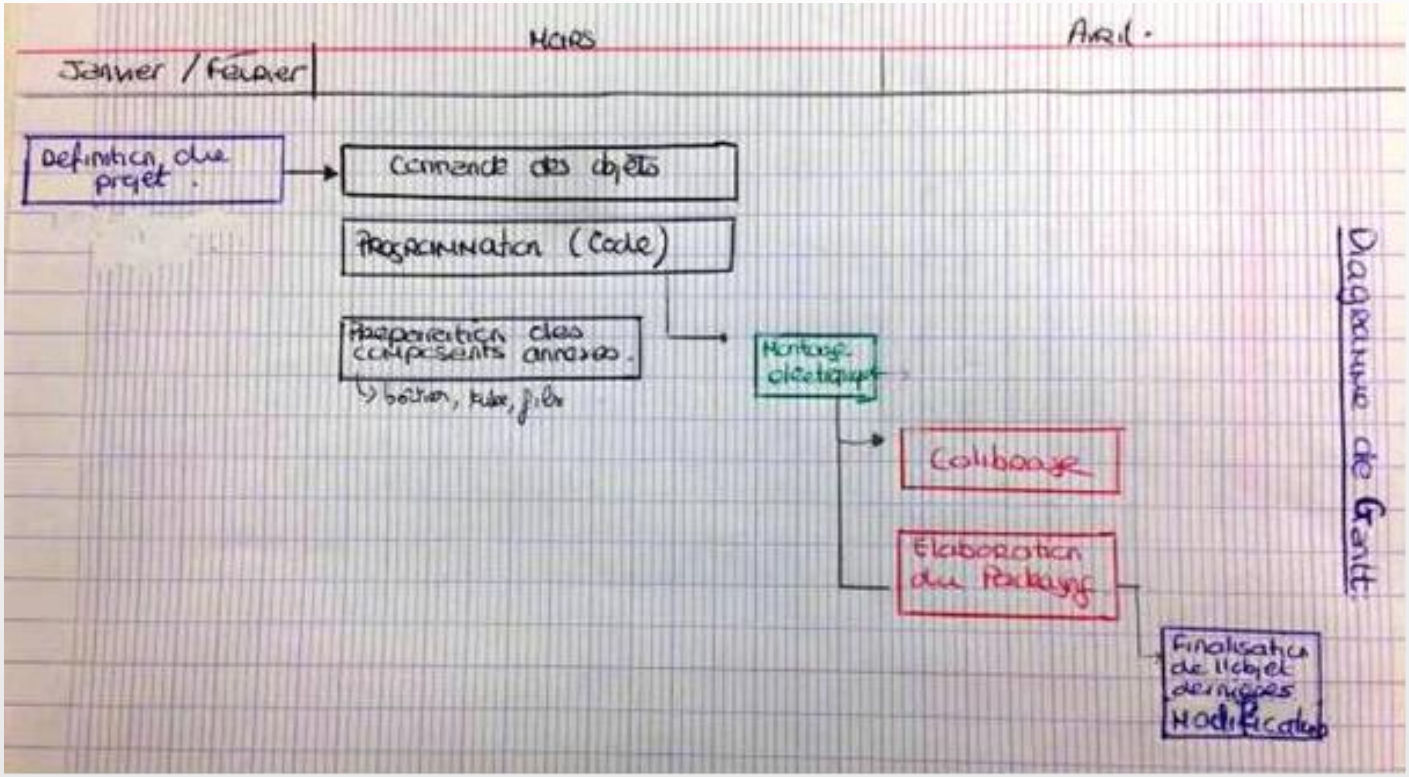
Éthylotest 21.7

Sommaire

1. Introduction
2. Modélisation
3. Assemblage
4. Programmation
5. Calibration
6. Conclusion

Introduction



Modélisation

Afficheur LCD

Arduino Uno

Capteur MQ-03

Connecteur de montage

Connecteur batterie

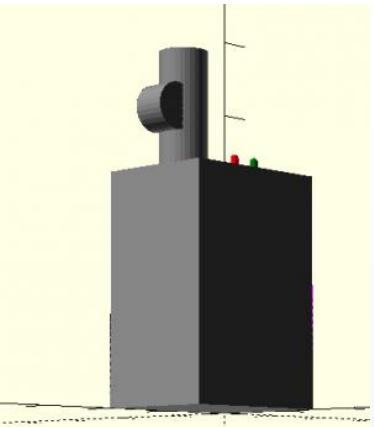
Interrupteur

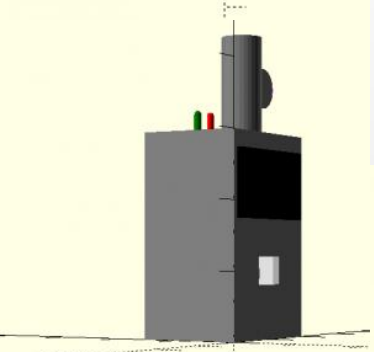
Potentiomètre

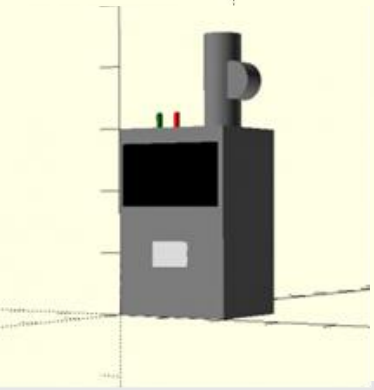
Resistances

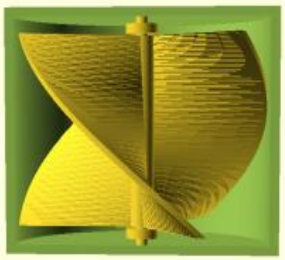
LEDs

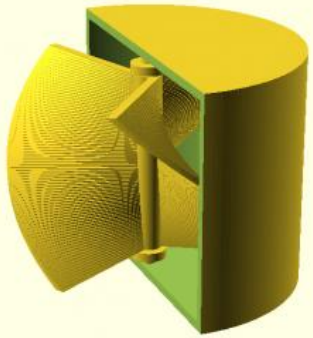
Modélisation (2)

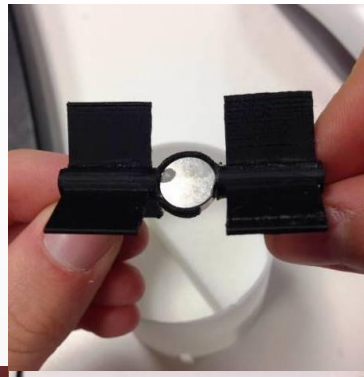


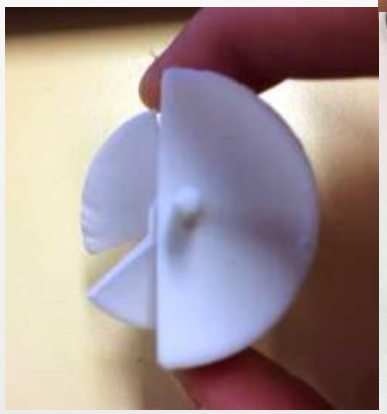


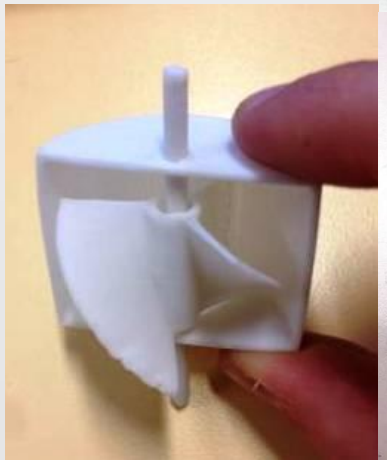


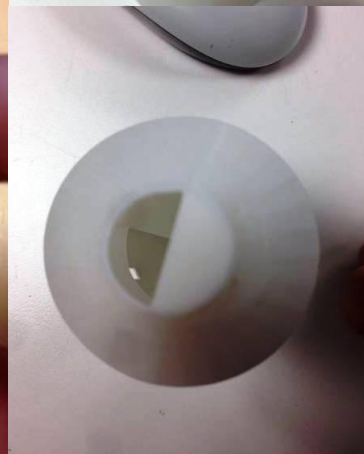




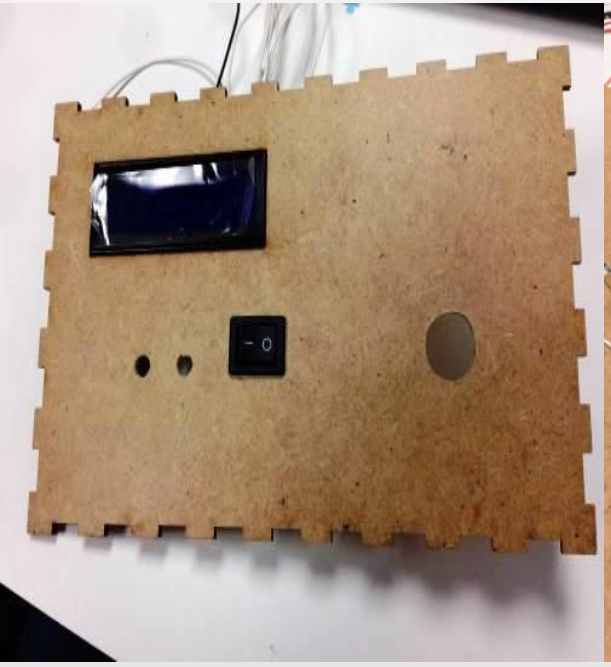


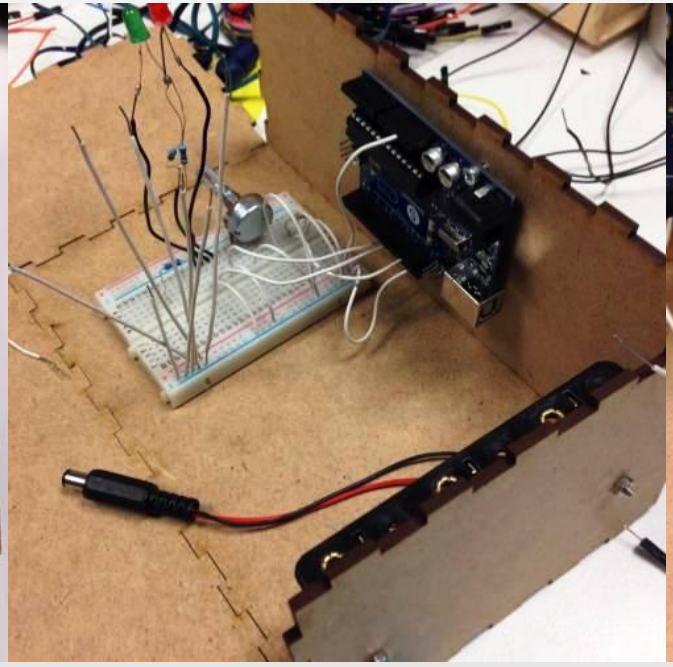


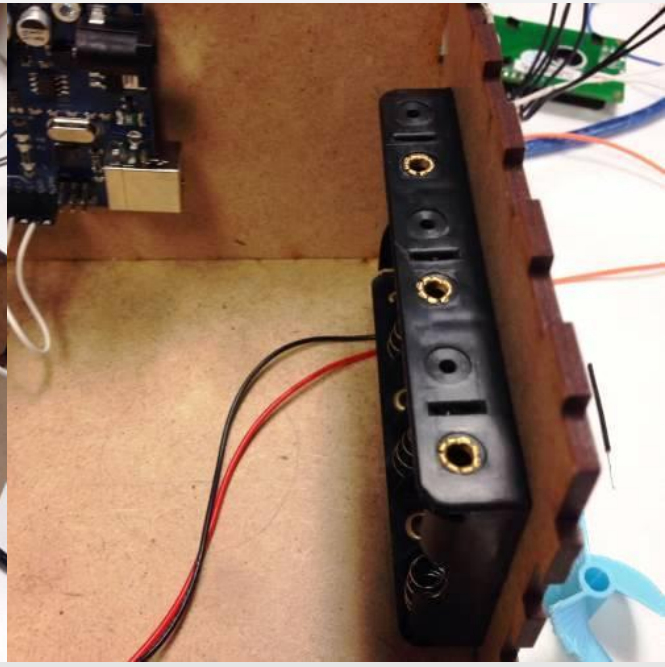




Assemblage





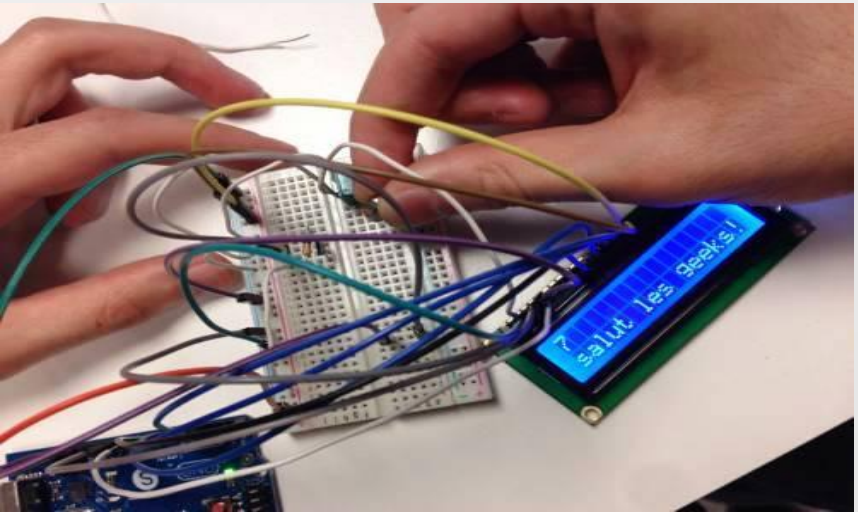


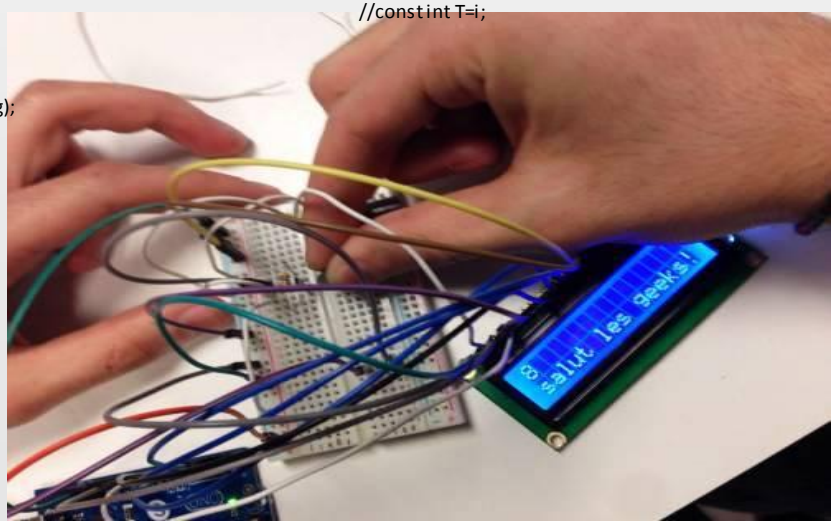
Programmation

```
#include <LiquidCrystal.h>      // on inclut la bibliothèque
#define LED 8
#define LEDa 9                          // on déclare les 2 LED en D9 et D8

const int analogPin = A0;                   // le capteur MQ-3 en A0
const int hallPin = 12;              // the number of the hall effect sensor pin
const int ledPin =  13;
int i=0;                          // variables will change:
int hallState = 0;                  // variable for reading the hall sensor status
int ref = 1;
int D=0;
float S=0.0004;
float P=3.1415;//9265358979323846;
float R=0.01;
int j=0;

LiquidCrystal lcd(11, 10, 5, 4, 3, 2);

// définit un objet lcd avec les entrées sorties de l'Arduino en paramètre
//RS_pin EN_pin BUS1 BUS2 BUS3 BUS4
long time;            // pour pouvoir compter jusqua 60s

 void setup() {

  pinMode(LED,  OUTPUT);
  pinMode(LEDa, OUTPUT);
 pinMode(ledPin, OUTPUT);     // initialize the LED pin as an output:
  pinMode(hallPin, INPUT);     // initialize the hall effect sensor pin as an input:
lcd.begin(16, 2);     // défini le nombre de caractère et de ligne de l'écran
Serial.begin(9600);   // on configure les 2 leds et on prépare le moniteur série
```

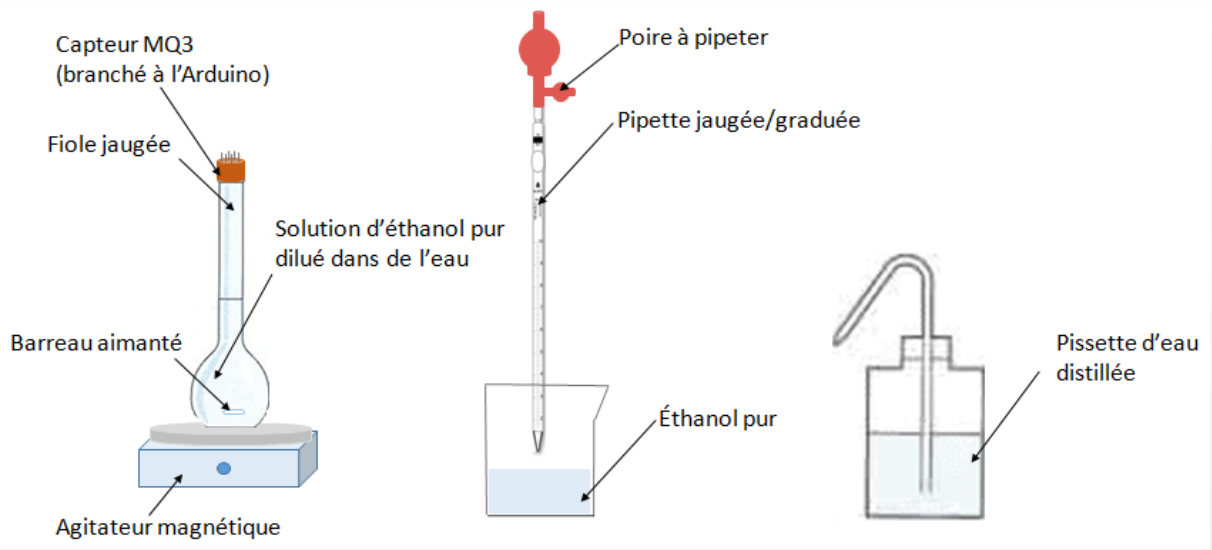
```
void loop() {
/*
while (time<=10000) {

    lcd.clear();
    hallState = digitalRead(hallPin);       // read the state of the hall effect sensor:
    if (hallState == LOW && ref==HIGH) {    // truc de compte binaire pour les tours
    digitalWrite(ledPin, HIGH);             // turn LED on:
    i=i+1;
}
   //lcd.clear();                    //afichage des tour
   // lcd.print(hallState);
   // lcd.setCursor(6, 0);
   // lcd.print(i);
    delay(5);                       // pour perception visualiser

   ref = hallState;
    time = millis();
  // lcd.setCursor(0, 1);
  // lcd.print("Time: ");
  // lcd.setCursor(6, 1);
  //lcd.print(time/1000);     //prints time since program started
   //delay(10);    // wait a second so as not to send massive amounts of data
   break;
                          float D=((S*R*2*P*i)/60*1000000);
                           lcd.clear();
                           lcd.setCursor(0, 0);
                           lcd.print("Dv =");
                           lcd.setCursor(5, 0);
                           lcd.print(D);
                           lcd.setCursor(10, 0);
                           lcd.print("mL/s");
                            lcd.setCursor(0, 1);
                           lcd.print("Alcool");
                           lcd.setCursor(7, 1);
                           lcd.print(sensorReading);
                           delay(1000);

                           lcd.clear();
                           lcd.print("Volt=");
                           lcd.setCursor(5, 0);
                           lcd.print(sensor_volt);
                           lcd.setCursor(10, 0);
                           lcd.print("RS=");
                           lcd.setCursor(13, 0);
                           lcd.print(RS_gas);
                           lcd.setCursor(0, 1);
                           lcd.print("Rs/R0 = ");
                           lcd.setCursor(8, 1);
                           lcd.print(ratio);
                           delay(1000);
```

```
*/
float sensor_volt;
 float RS_gas; // Get value of RS in a GAS
 float ratio; // Get ratio RS_GAS/RS_air
 float R0 = 0.11;

 int sensorReading = analogRead(analogPin);

sensor_volt=(float)sensorReading/1024*5.0;
 RS_gas = (5.0-sensor_volt)/sensor_volt; // omit *RL

//Replace the name "R0" with the value of R0 in t
he demo of First Test - ratio = RS_gas/R0;  // ratio = RS/R0

 /*
 if (sensorReading>400) {
digitalWrite(LED, LOW);
 digitalWrite(LEDa, HIGH);
 }
 else {
 digitalWrite(LEDa, LOW);

   digitalWrite(LED,HIGH);
  }
 if (time>=10000) {
   */
    //lcd.clear();
    //lcd.print(i);
    //lcd.setCursor(6, 0);
    //lcd.print("tours");

    //unsigned int w=(2*PI*i)/60;
    //unsigned int v=w*R;
    //const int T=i;
```





Calibration



$$C_{m_{EtOH}}^{G} = M_{EtOH} \times \frac{\frac{\frac{C_{m_{EtOH}}^{L}}{M_{EtOH}}}{\frac{C_{m_{EtOH}}^{L}}{M_{EtOH}} + \frac{C_{m_{eau}}^{L}}{M_{eau}}} \times P_{EtOH}^{sat}}{P_{atm}} \times \left(\frac{h_{sat} \times h_r}{M_{eau}} + \frac{1}{M_{air}}\right) \times C_{m_{air}}^{G}}{1 - \frac{\frac{\frac{C_{m_{EtOH}}^{L}}{M_{EtOH}}}{\frac{C_{m_{EtOH}}^{L}}{M_{EtOH}} + \frac{C_{m_{eau}}^{L}}{M_{eau}}} \times P_{EtOH}^{sat}}{P_{atm}}}$$

Conclusion