

ANNEXE 1

PROGRAMME ARDUINO - MESURE DES PARTICULES PM10

```
/*
  Lycée Jacques de VAUCANSON
  TPE SUR LE CAPTEUR DE POUSSIERE
  Décembre 2018
*/
// Inclusion des bibliothèques pour la carte SD
#include <SPI.h>
#include <SD.h>
File dataFile;
String dataString = ""; // Déclaration d'une chaîne de caractère
int PasCarte = 0;
unsigned long Nb = 0;
boolean start = false;

// Inclusion des bibliothèques pour l'afficheur LCD
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x20, 16, 2); // Configuration de l'adresse du LCD à 0x20 pour un
afficheur de 16 caractères sur 2 lignes

// Variables pour le capteur de poussière
unsigned long duration;
unsigned long starttime;
unsigned long sampletime_ms = 5000;
unsigned long lowpulseoccupancy = 0;
float ratio = 0;
float concentration = 0;

void setup()
{
  Serial.begin(9600);
  lcd.init();
  lcd.backlight();
  pinMode(8, INPUT);
  pinMode(A1, INPUT);
  pinMode(A0, OUTPUT);
  // Initialisation de la carte SD
  if (!SD.begin(4)) // Test si la carte SD est présente
  {
    // Carte non présente
    PasCarte = 1;
    lcd.setCursor(0, 0);
    lcd.print("ERREUR CARTE");
    lcd.setCursor(0, 1);
    lcd.print("SD ABSENTE");
    for (char i = 0; i < 500; i++)
    {
```

```

    digitalWrite(A0, 0);
    delay(50);
    digitalWrite(A0, 1);
    delay(50);
}
return;
}
// Carte initialisée
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Carte SD");
lcd.setCursor(0, 1);
lcd.print("INITIALISEE");
delay(1000);

// Si le fichier n'est pas présent, écriture dans le fichier la ligne d'entête
if (!SD.exists("data.csv"))
{
    dataFile = SD.open("data.csv", FILE_WRITE);
    dataFile.close();
    dataFile = SD.open("data.csv", FILE_WRITE);
    dataString = "Numero; Concentration";
    dataFile.println(dataString);
    // CrÉation fichier data
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Creation");
    lcd.setCursor(0, 1);
    lcd.print("data.csv");
    delay(1000);
}
dataFile.close();
}

void loop()
{
    // Mise en marche et arrÊt avec le bouton poussoir
    if (digitalRead(A1))
    {
        lcd.clear();
        if (start)
        {
            start = false;
            lcd.print("STOP ");
        }
        else
        {
            start = true;
            lcd.print("START ");
        }
        while (digitalRead(A1));
    }
}

```

```

// Réinitialisation de la variable starttime et lowpulseoccupancy
lowpulseoccupancy = 0;
starttime = millis();
}

// Mesures
if (start)
{
duration = pulseIn(8, LOW);
lowpulseoccupancy = lowpulseoccupancy + duration;

if ((millis() - starttime) >= sampletime_ms) // Test positif toutes les 5 secondes avec
sampletime_ms = 5000
{
// Allumage de la led
digitalWrite(A0, 1);

// Calcul de la concentration
ratio = lowpulseoccupancy / (sampletime_ms * 10.0); // Integer percentage 0=>100
concentration = (1.1 * pow(ratio, 3) - 3.8 * pow(ratio, 2) + 520 * ratio + 0.62)/0.283.; //
using spec sheet curve

// Affichage de la concentration sur l'afficheur LCD
lcd.setCursor(0, 0);
lcd.print("Capt poussières");
lcd.setCursor(0, 1);
lcd.print("Taux = ");
lcd.print((int)concentration);
lcd.print('.');
lcd.print((int)((float)(concentration - (int)concentration) * 100));
lcd.print(" ");

// Affichage de la concentration sur l'afficheur LCD
Serial.print("concentration = ");
Serial.print(concentration);
Serial.println(" pcs/0.01cf");
Serial.println("\n");

// Ecrire les données sur la carte SD toutes les minutes
if (PasCarte == 0)
{
Nb++;
// Ouvrir le fichier CSV en écriture
dataFile = SD.open("data.csv", FILE_WRITE);

// Si le fichier est présent, écriture dans le fichier
if (dataFile)
{
dataFile.print(Nb);
dataFile.print(';');
}
}
}

```

```

    dataFile.print((int)concentration);
    dataFile.print(',');
    dataFile.println((int)((float)(concentration - (int)concentration) * 100));
    dataFile.close();
  }
}
// Extinction de la led
digitalWrite(A0, 0);

// Réinitialisation de la variable starttime et lowpulseoccupancy
lowpulseoccupancy = 0;
starttime = millis();
}
}
else
{
  // Cycle Ã l'arrêt
  lcd.setCursor(0, 0);
  lcd.print("PRET AU      ");
  lcd.setCursor(0, 1);
  lcd.print("DEPART ");
  if (PasCarte == 1) lcd.print("sans SD  ");
  else      lcd.print("      ");
}
}
}

```

ANNEXE 2

PROGRAMME ARDUINO - MESURE DU CO

Présentation du programme arduino - capture d'écran : mesure du CO

```
void setup()
{
  Serial.begin(9600); //Set serial baud rate to 9600 bps
  pinMode(8,INPUT);
  pinMode(LED_BUILTIN,OUTPUT);
}
void loop()
{
  int val;
  int limit;
  val=analogRead(0);//Read Gas value from analog 0
  limit=digitalRead(8);
  Serial.print("Co value =");
  Serial.println(val,DEC);//Print the value to serial port
  Serial.print("Limit: ");
  Serial.println(limit);
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(500);
  digitalWrite(LED_BUILTIN, LOW);
  delay(500);
}
```