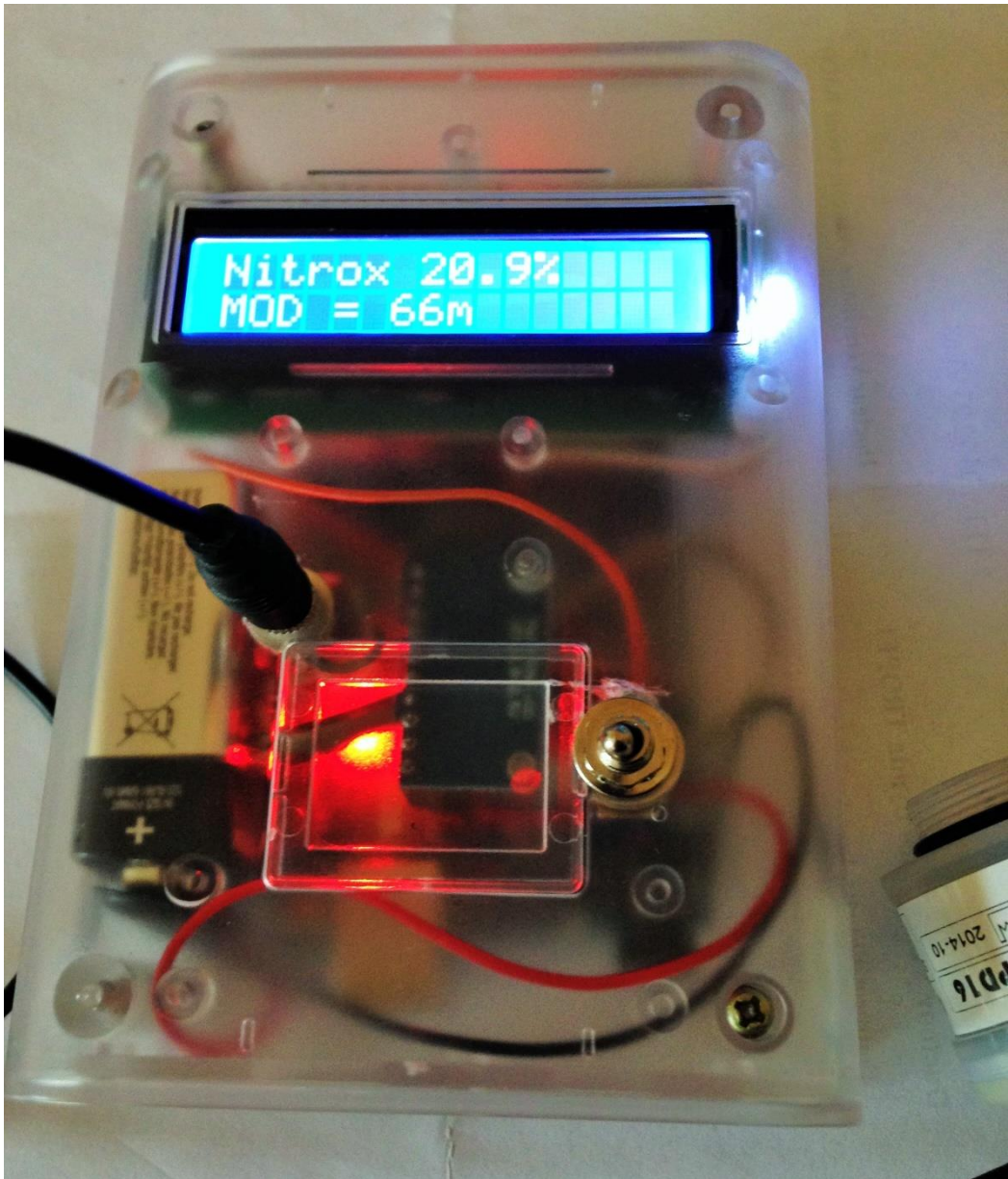


Analyseur Nitrox Arduino :



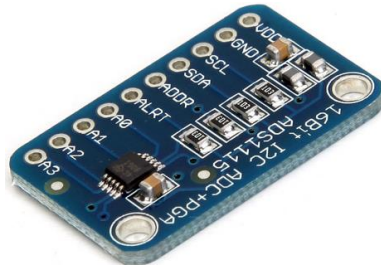
Principe :

- Utilisation d'une cellule oxygène pour mesurer le taux d'O₂ du mélange (30€)
- Utilisation d'une carte Arduino pour gérer logiquement le calibrage du capteur, et afficher le Nitrox analysé et le MOD associé (3€)
- Utilisation d'une carte ADS1115 pour mesurer précisément des tensions entre 5mV et 100mV (ce que ne sait pas faire la carte Arduino nativement) (3€)
- Affichage du taux de Nitrox du mélange et du MOD associé sur un écran LCD 16 caractères sur 2 lignes, avec interface I2C (3€)

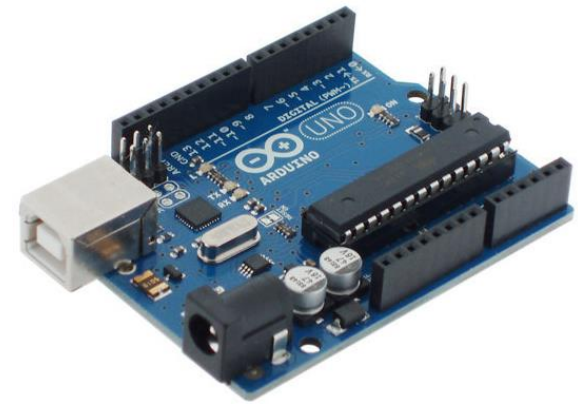
les ingrédients :



écran LCD 2 lignes
I2C 1602



convertisseur ADC 16 bits
ADS1115



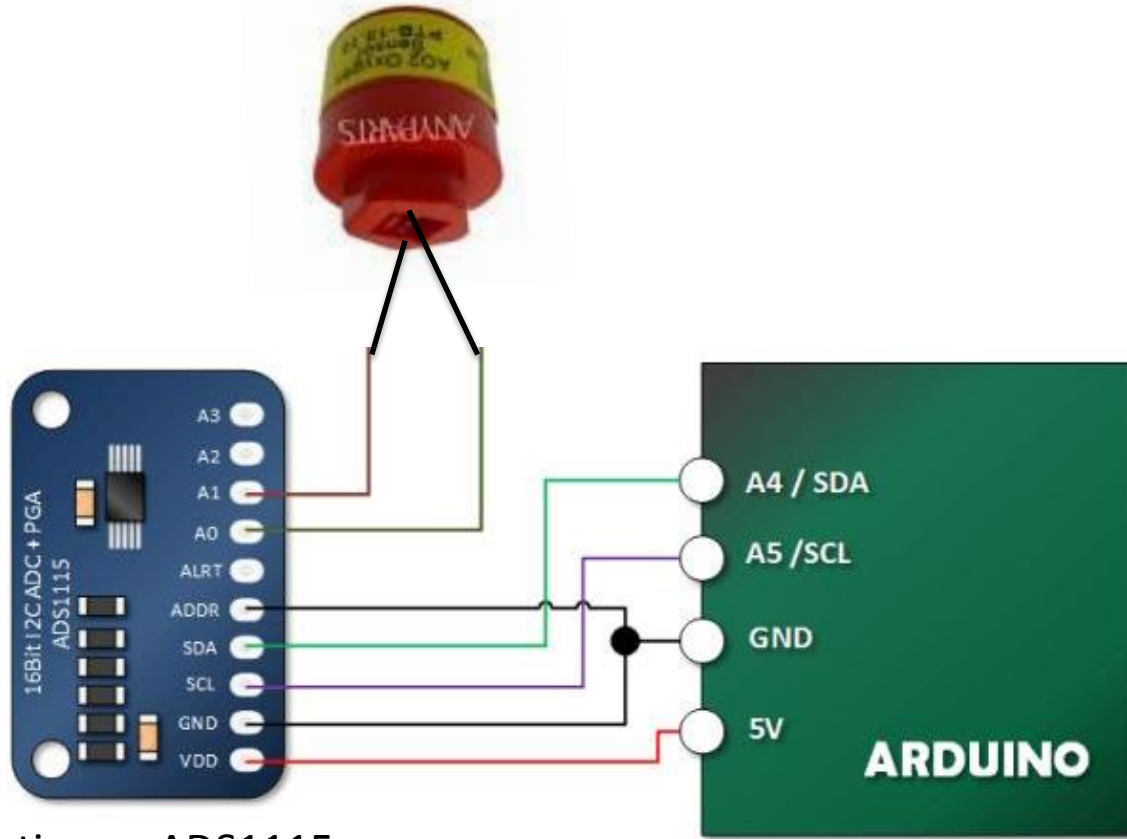
carte Arduino



cellule oxy AO2 Citycell

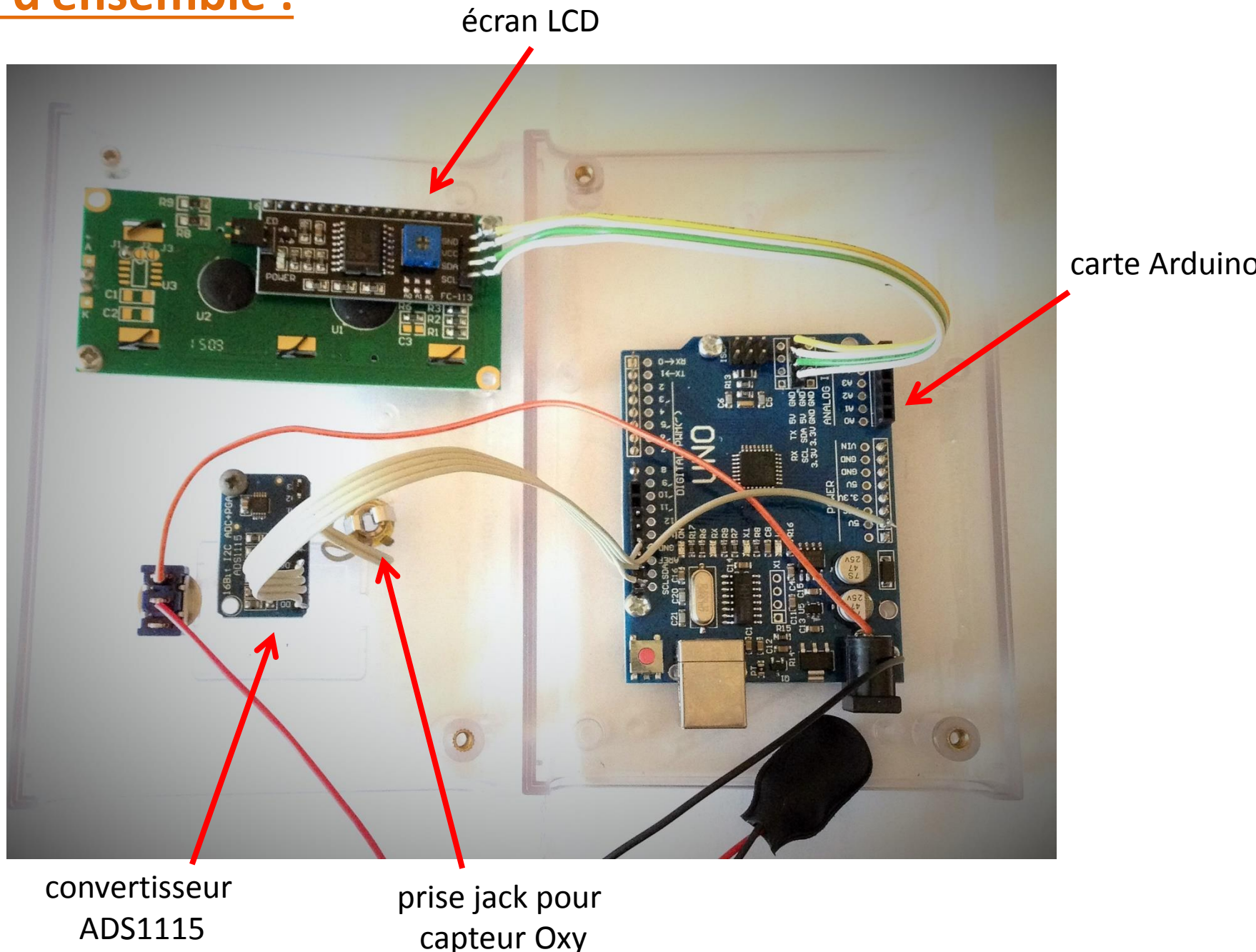
cablage cellule O2 :

cellule Oxy



convertisseur ADS1115

vue d'ensemble :



Le programme dans l'Arduino (déclarations) :

```
#include <Wire.h>
#include <Adafruit_ADS1015.h>
#include <RunningAverage.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,2, 1, 0, 4, 5, 6, 7, 3, POSITIVE); // set the
LCD address to 0x20 for a 16 chars and 2 line display
Adafruit_ADS1115 ads(0x48);

// variables will change:
float TensionCalib = 0; // mise a 0 de la tension de calibrage
float voltage = 0;
int MOD = 0; // valeur de MOD du mélange
float gain = 0.0078125;
RunningAverage RA(10);

// Définition du gaz de référence
float calibgas=20.9;
```

Le programme dans l'Arduino (setup) :

```
void setup() {  
  
  // initialize serial communication at 9600 bits per second:  
  Serial.begin(9600);  
  
  lcd.begin(16,2);  
  lcd.backlight();  
  lcd.clear();  
  lcd.print("Analyseur Nitrox");  
  
  ads.setGain(GAIN_SIXTEEN); // 16x gain 1 bit = 0.0078125mV  
  ads.begin();  
  
  int16_t adc0;  
  adc0 = ads.readADC_Differential_0_1();  
  RA.addValue(adc0);  
  
  voltage = abs(RA.getAverage()*gain);  
  int i = 0;  
  
  // affichage de la tension  
  lcd.setCursor(0,1);  
  lcd.print("V cell : ");  
  lcd.print(voltage,2);  
  lcd.print("mV");  
  delay(2000);  
  
  lcd.clear();  
  lcd.print("Calib. en cours");  
  
  float tensionMoyenne = 0;  
  for(i = 1; i <10 or (abs (voltage - (tensionMoyenne / (i-1)))) > 0.001;  
  i++)  
  {  
    adc0 = ads.readADC_Differential_0_1();  
    RA.addValue(adc0);  
    voltage = abs(RA.getAverage()*gain);  
    tensionMoyenne = tensionMoyenne + voltage;  
    delay(200);  
  }  
  
  lcd.clear();  
  if (voltage < 2) {  
    lcd.print("Capteur Oxy HS !");  
    delay(3000);  
  }  
  else {  
    lcd.print("Calibrage OK");  
  }  
  
  delay(1000);  
  
  tensionMoyenne = tensionMoyenne / (i - 1);  
  TensionCalib = tensionMoyenne;  
  lcd.clear();  
  lcd.print("Tension Calib :");  
  lcd.setCursor(0,1);  
  lcd.print(TensionCalib,2);  
  lcd.print("mV");  
  delay(1000);  
  
  lcd.clear();  
  lcd.print("Nitrox");  
  lcd.setCursor(0,1);  
  lcd.print("MOD = ");  
}
```

Le programme dans l'Arduino (loop) :

```
void loop() {

    int16_t adc0;
    adc0 = ads.readADC_Differential_0_1();
    RA.addValue(adc0);

    voltage = abs(RA.getAverage()*gain);
    float nitrox = 0;

    nitrox = voltage * (20.9 / TensionCalib);
    MOD = 10 * ( (160/nitrox) - 1);

    // si cellule HS
    if (voltage < 2) {
        lcd.clear();
        lcd.print("Capteur Oxy HS !");
        goto CellulesHS;
    }

    lcd.setCursor(7,0);
    lcd.print(nitrox,1);
    lcd.print("% ");
    lcd.setCursor(6,1);
    lcd.print(MOD);
    lcd.print("m ");

    CellulesHS:
    delay(500);    // delay in between reads for stability
}
```


Les éléments à acheter :

- Carte Arduino Uno : [ici](#)
- Ecran LCD 1602 I2C : [ici](#)
- Carte ADS1115 : [ici](#)
- Cellule oxy : [ici](#)

- le boîtier : [ici](#)