

# Régulation du taux d'humidité

UE 803 - Optimisation et Contrôle des Procédés - Master 2 - Ingénierie Chimique

Projet Expérimental Arduino

Objectif du projet :

Réussir à réguler un flux d'air sec et un flux de d'air humidifié à l'aide d'un barboteur pour réguler le taux d'humidité dans l'air à l'aide de 2 pompes et du logiciel Arduino.

Pourquoi avons-nous choisi ce projet ?

La régulation de l'humidité de l'air est un procédé très utilisé en industrie pour réaliser des analyses qualitatives mais aussi dans le but de tester la résistance de certains produits à l'humidité notamment en industrie.

Liste du matériel :

2 x Électrovannes pour l'air

2 x Tuyaux hermétiques pour le montage qui seront découpés en plusieurs morceaux

1 x Barboteur maison

1 x Capteur d'humidité

2 x Pompes

1 x Carte Arduino

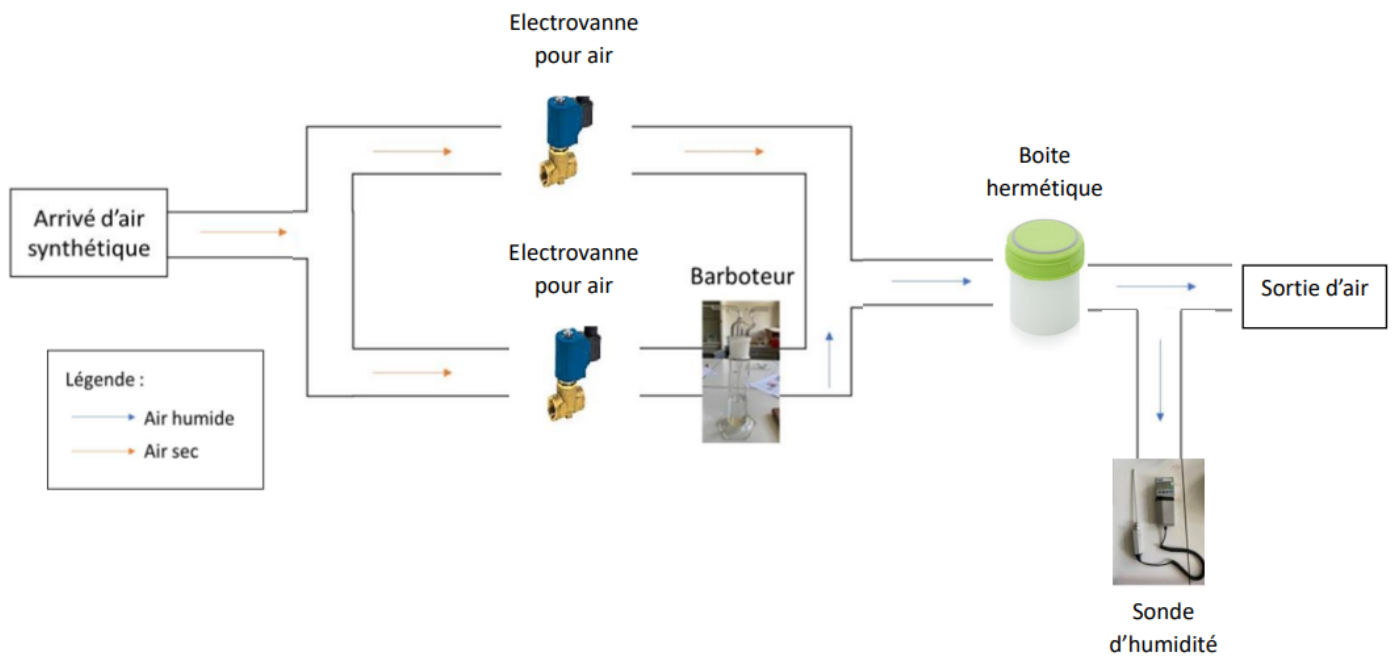
2 x bouteilles

1 x paquet de riz

2 x relais

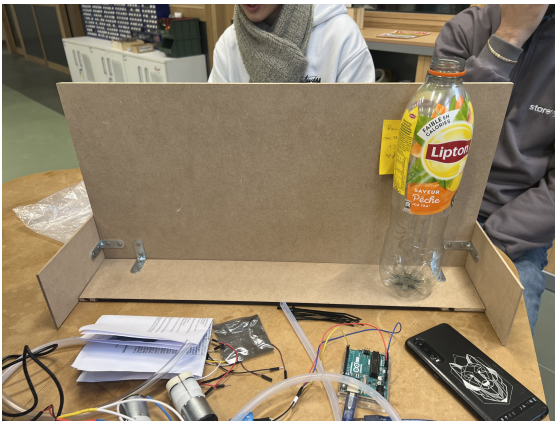
fils électrique pour raccorder les appareils entre eux

- Réalisation d'un schéma prototype du montage (28/10/2022)



On a réalisé un premier schéma du montage afin d'étudier les équipements dont nous aurions besoin pour réaliser notre projet.

- Réalisation du support du projet (12/01/2023)



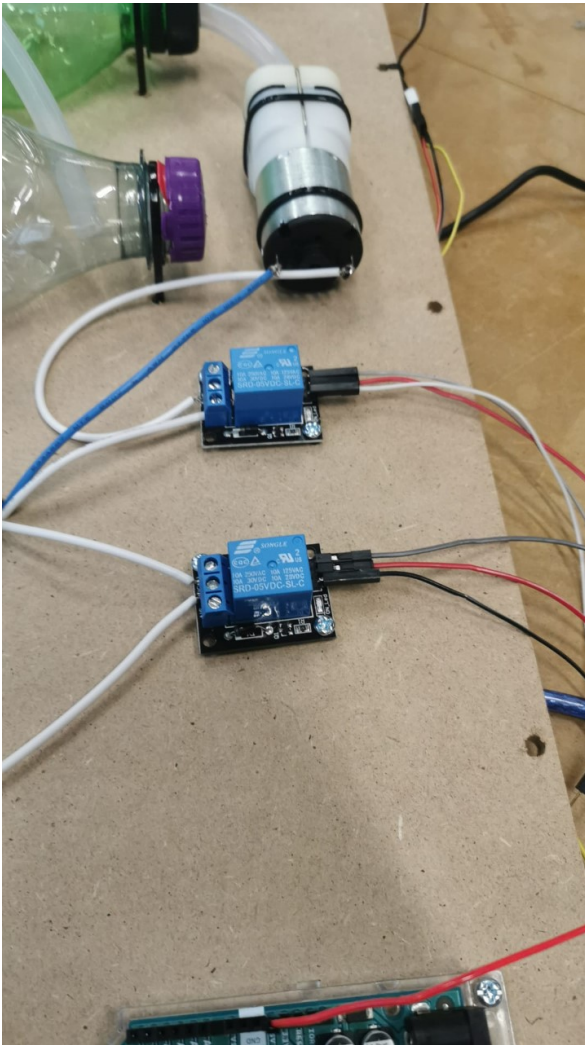
Sur cette première image on peut voir support qui a été assemblé à partir de bois recycler.

- Réalisation du barboteur (12/01/2023)



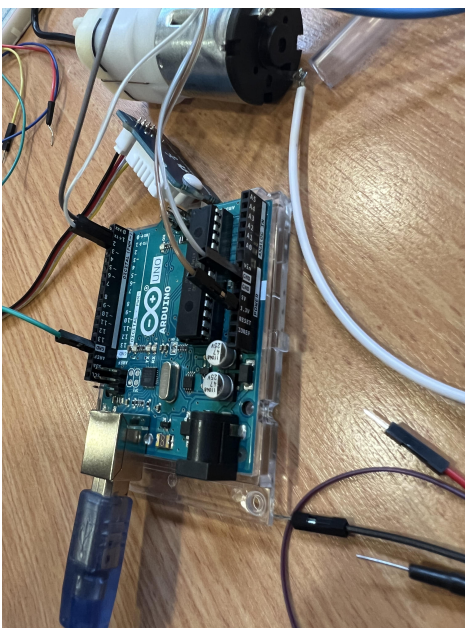
Afin de réaliser notre barboteur nous utilisons une bouteille de soda sur laquelle nous réalisons 2 trous : le 1er en bas de la bouteille qui est la sortie d'air humide et le second en haut de la bouteille au niveau du bouchon pour l'entrée gaz sec. Les trous ont été réalisés à l'aide de la perceuse à colonne qui nous permet d'avoir plus de précision pour la réalisation de nos trous. Pour humidifier l'air il entre par le haut de la bouteille dans un tube qui bullera dans l'eau pour l'humidifier.

- Réalisations des relais pour le contrôle des pompes (12/01/2023)



Des relais ont été ajoutés afin de pouvoir contrôler les pompes individuellement en fonction du signal envoyé par la carte arduino.

- Réalisation du montage Arduino / Code (26/01/2023)



## Code que nous avons trouvé et qui a été testé :

```
*/

#include <DHT.h>

#define brocheDeBranchementDHT 6 // La ligne de communication du DHT22 sera donc branchée sur la pin
D6 de l'Arduino

#define typeDeDHT DHT22 // Ici, le type de DHT utilisé est un DHT22 (que vous pouvez changer en
DHT11, DHT21, ou autre, le cas échéant)

// Instanciation de la librairie DHT
DHT dht(brocheDeBranchementDHT, typeDeDHT);

// =====
// Initialisation programme
// =====
void setup () {

    // Initialisation de la liaison série (pour retourner les infos au moniteur série de l'ordi)
    Serial.begin(9600);
    Serial.println("Programme de test du DHT22");
    Serial.println("=====");
    Serial.println();

    // Initialisation du DHT22;
    dht.begin();
}

// =====
// Boucle principale
// =====
void loop () {

    // Lecture des données
    float tauxHumidite = dht.readHumidity(); // Lecture du taux d'humidité (en %)
    float temperatureEnCelsius = dht.readTemperature(); // Lecture de la température, exprimée en degrés
    Celsius

    // Vérification si données bien reçues
```

```

if (isnan(tauxHumidite) || isnan(temperatureEnCelsius)) {
    Serial.println("Aucune valeur retournée par le DHT22. Est-il bien branché ?");
    delay(2000);
    return;    // Si aucune valeur n'a été reçue par l'Arduino, on attend 2 secondes, puis on redémarre la
fonction loop()
}

// Calcul de la température ressentie
float temperatureRessentieEnCelsius = dht.computeHeatIndex(temperatureEnCelsius, tauxHumidite, false); //
Le "false" est là pour dire qu'on travaille en °C, et non en °F

// Affichage des valeurs
Serial.print("Humidité = "); Serial.print(tauxHumidite); Serial.println(" %");
Serial.print("Température = "); Serial.print(temperatureEnCelsius); Serial.println(" °C");
Serial.print("Température ressentie = "); Serial.print(temperatureRessentieEnCelsius); Serial.println(" °C");
Serial.println();

// Temporisation de 2 secondes (pour rappel : il ne faut pas essayer de faire plus d'1 lecture toutes les 2
secondes, avec le DHT22, selon le fabricant)
delay(2000);
}

```

On réalise les premiers branchements des 2 pompes sur la carte arduino ainsi que le capteur d'humidité qui nous a d'ailleurs posé un peu problème car le capteur possède 4 branches et nous n'avons la possibilité que de brancher 4 branches nous avons donc demandé de l'aide à monsieur Pulpytel pour nous aider pour ce branchement. Suite à ça nous avons recherché le code afin de faire fonctionner le capteur d'humidité et pouvoir récupérer les valeurs.

Code que nous avons réalisé :

```

#include <DHT.h>
#include <stdio.h>
#include <stdlib.h>
#define brocheDeBranchementDHT 6    // La ligne de communication du DHT22 sera donc
branchée sur la pin D6 de l'Arduino
#define typeDeDHT DHT22             // Ici, le type de DHT utilisé est un DHT22 (que vous pouvez
changer en DHT11, DHT21, ou autre, le cas échéant)

// Instanciation de la librairie DHT
DHT dht(brocheDeBranchementDHT, typeDeDHT);

int objectif;
const int relais_pompe1 = 3;

```

```

const int relais_pompe2 = 4;
// =====
// Initialisation programme
// =====
void setup () {

    // Initialisation de la liaison série (pour retourner les infos au moniteur série de l'ordi)
    Serial.begin(9600);
    Serial.println("Programme de test du DHT22");
    Serial.println("=====");
    Serial.println();
    pinMode(BROCHE_CAPTEUR, INPUT_PULLUP);
    pinMode(relais_pompe1, OUTPUT);
    pinMode(relais_pompe2, OUTPUT);
    // Initialisation du DHT22;
    dht.begin();
}

// =====
// Boucle principale
// =====
void loop () {

    // Lecture des données
    float tauxHumidite = dht.readHumidity();           // Lecture du taux d'humidité (en %)
    float temperatureEnCelsius = dht.readTemperature(); // Lecture de la température, exprimée en
    degrés Celsius
    printf("quel est le taux d'humidité souhaité ?");
    scanf("%d",&objectif);
    // Vérification si données bien reçues
    if (isnan(tauxHumidite) || isnan(temperatureEnCelsius)) {
        Serial.println("Aucune valeur retournée par le DHT22. Est-il bien branché ?");
        delay(2000);
        return; // Si aucune valeur n'a été reçue par l'Arduino, on attend 2 secondes, puis on
redémarre la fonction loop()
    }

    // Calcul de la température ressentie
    float temperatureRessentieEnCelsius = dht.computeHeatIndex(temperatureEnCelsius,
    tauxHumidite, false); // Le "false" est là pour dire qu'on travaille en °C, et non en °F

    // Affichage des valeurs
    Serial.print("Humidité = "); Serial.print(tauxHumidite); Serial.println(" %");
    Serial.print("Température = "); Serial.print(temperatureEnCelsius); Serial.println(" °C");
    Serial.print("Température ressentie = "); Serial.print(temperatureRessentieEnCelsius);
    Serial.println(" °C");
}

```

```
Serial.println();
```

```
// Temporisation de 2 secondes (pour rappel : il ne faut pas essayer de faire plus d'1 lecture  
toutes les 2 secondes, avec le DHT22, selon le fabricant)
```

```
delay(2000);
```

```
while(tauxHumidite!=objectif){
```

```
if(tauxHumidite<objectif)
```

```
{
```

```
digitalWrite(relais_pompe1, HIGH);
```

```
digitalWrite(relais_pompe2, LOW);
```

```
}
```

```
else
```

```
{
```

```
digitalWrite(relais_pompe1, LOW);
```

```
digitalWrite(relais_pompe2, HIGH);}
```

```
}
```

Pour la seconde partie du code concernant la régulation des flux d'air sec et d'air humide, un code a été réalisé mais la prise en compte des données mesurées par le capteur dans ce code est un problème que nous n'arrivons pas à résoudre. Ainsi il est difficile de finaliser le code si nous ne prenons pas en compte les données mesurées par le capteur.

on du flux d'air sec (26/01/2023)



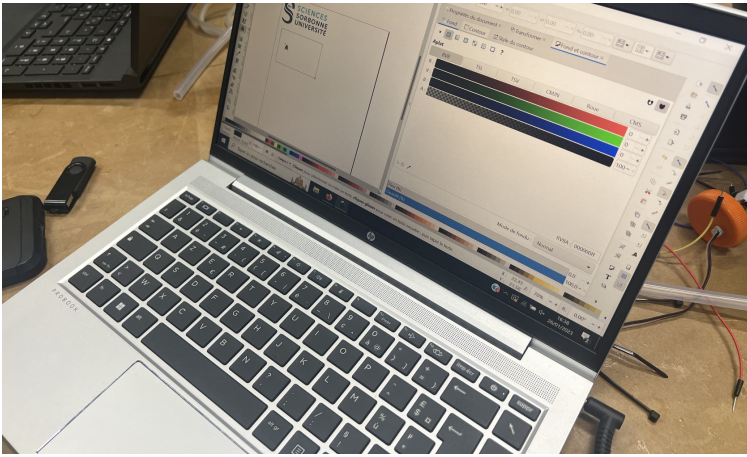


On sait que en moyenne le taux d'hydrométrie de l'air en France est de 50%. Dans le but de faire diminuer ce taux, nous avons eu l'idée de rajouter une bouteille pleine de riz qui d'après nous serait capable d'absorber une partie de l'humidité contenu dans l'air et ainsi nous permettre d'élargir notre gamme de régulation. Afin d'éviter que le riz entre dans nos tuyaux et les bouche nous avons ajouté un morceau de torchon qui nous permet de servir de filtre.

- Assemblage sur le support (26/01/2023)

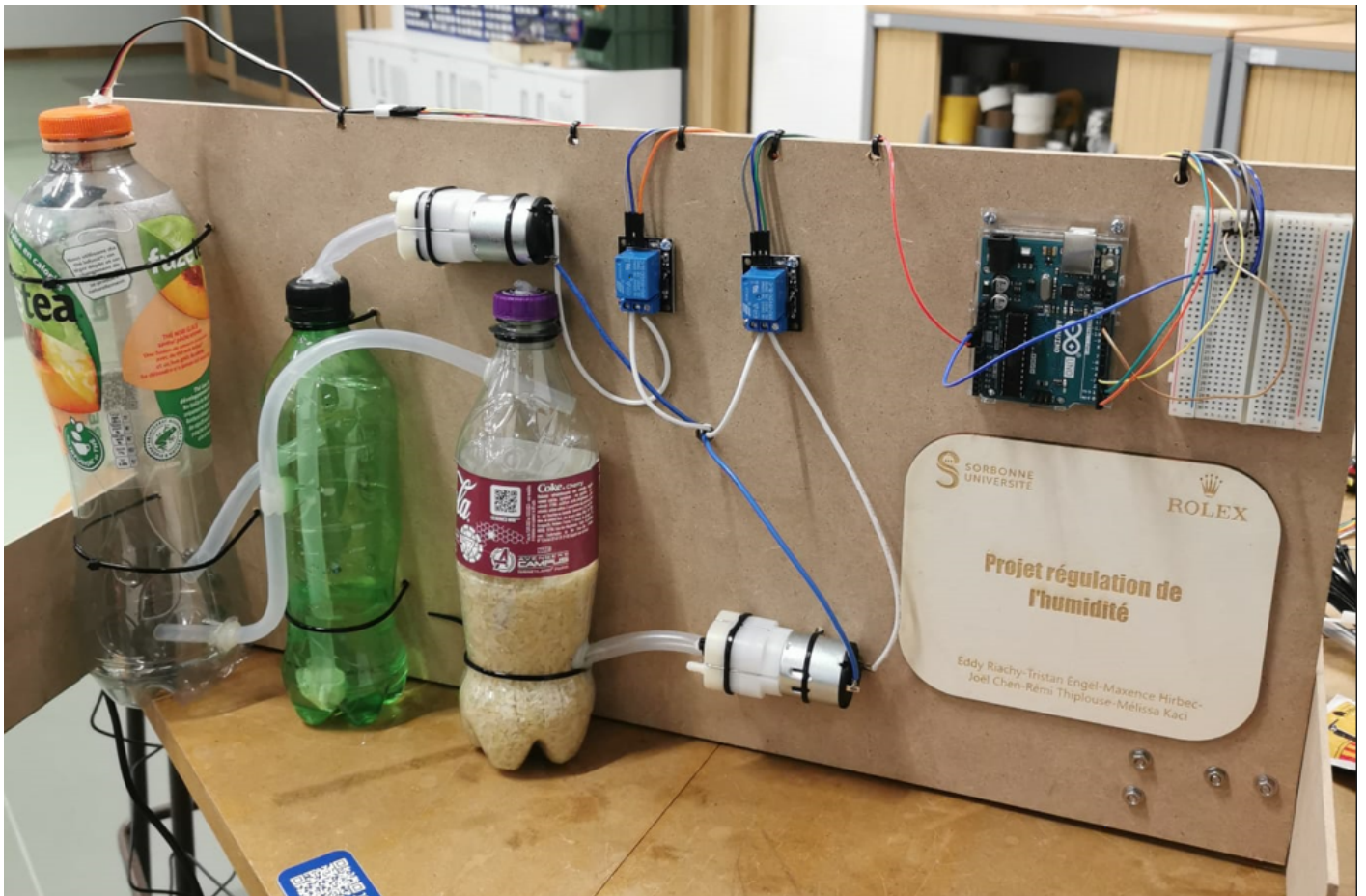


- Réalisation de visuel avec l'imprimante laser (26/01/2023)



Afin de réaliser un projet un peu plus esthétique nous avons réalisé un petit visuel à l'aide de l'imprimante laser sur une petite plaquette de bois.

- Projet final assemblé (26/01/2023)



On peut observer sur cet image notre projet final, on a ainsi nos 2 pompes contrôlées en mode on/off à partir de Arduino. La pompe du bas correspond à notre flux d'air sec qui passe par le riz et arrive dans la bouteille de fuze tea ou les flux seront mélangés, la pompe du haut contrôle notre flux de gaz humide qui est humidifié dans notre bouteille verte qui est notre barboteur improvisé relié à la bouteille de fuze tea. Ainsi notre capteur d'humidité se trouve en haut de la bouteille de fuze tea. Si le capteur d'humidité se trouve en haut de la bouteille c'est pour éviter qu'il soit trop proche des arrivées de flux d'air sec et humide afin de ne pas fausser les valeurs.

En conclusion le barboteur a été testé et fonctionne, pareil pour nos 2 pompes de flux d'air humide et d'air sec. Cependant nous avons un problème concernant l'écriture du code et plus spécifiquement la mise en commun du code pour le capteur d'humidité et le code concernant la régulation de la pompe.

---

Revision #12

Created 26 January 2023 14:29:39 by Thioulouse Remi

Updated 23 February 2023 12:36:06 by Engel Tristan