

SERRE AUTONOME ET AUTOMATISEE

JARDUINO



- 1 Processus de construction
- 2 Problèmes rencontrés
- 3 Présentation du code
- 4 Organisation



PROCESSUS DE CREATION

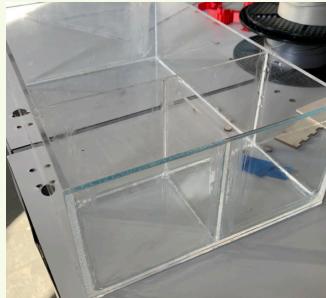
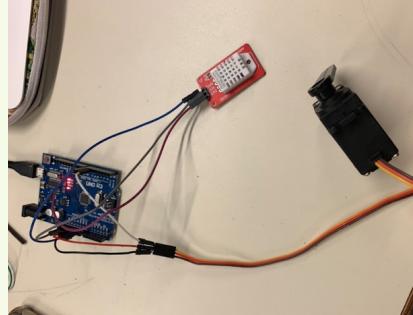
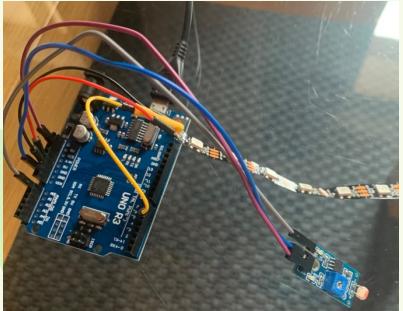
Matériel utilisé :

Capteur de luminosité - leds
Capteur d'humidité du sol - pompe à eau
Capteur d'humidité de l'air - servomoteur

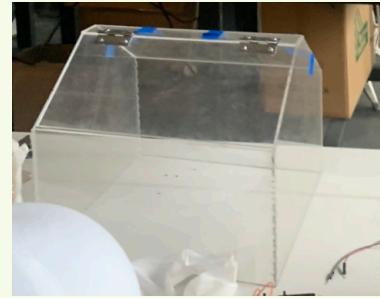
Construction de la serre :

Réalisée en plexiglass
Transparente et étanche

Mise en fonctionnement :
Etude de chaque capteur
Liaison capteur et composant



Partie basse de la serre

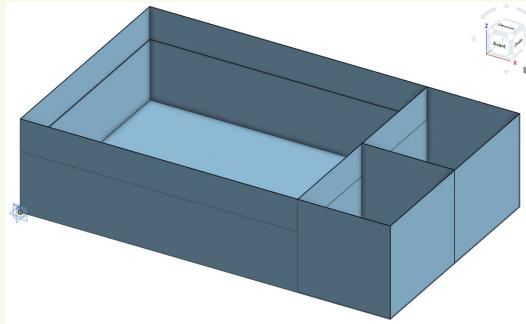


Partie haute de la serre

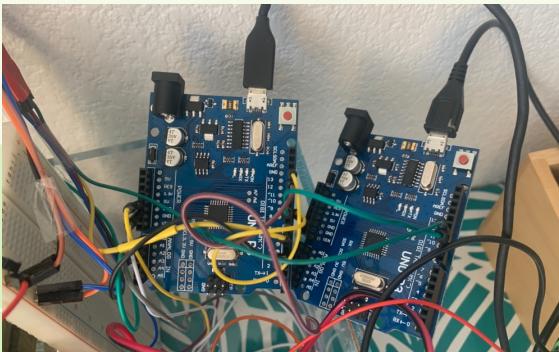


PROBLEMES RENCONTRES

Problèmes lors de la construction



Problèmes lors du rassemblement des composants et du code



Réalisation du bas de la serre sur Oneshape pour une impression 3D.

Résultat inutilisable : trop fin et matériel qui ne correspond pas à notre projet.

Solution : Nous décidons de construire notre serre en plexiglass.

Incompatibilité des leds et du servomoteur.
Recherches afin de trouver une solution.

Solution: utiliser une 2^{ème} carte Arduino et un autre code réservés au servomoteur.

PRESENTATION DU CODE

```
//initialisation pour le servomoteur et le capteur d'humidité air

#include <Servo.h> //bibliothèque pour le servomoteur
#include <DHT.h> //bibliothèque pour le DHT 22
#define brocheDeBranchementDHT 4
#define typeDeDHT DHT22

DHT dht(brocheDeBranchementDHT, typeDeDHT);
Servo monServomoteur;

// initialisation pour le capteur jour/nuit et les leds

#include <Adafruit_NeoPixel.h> //librairie pour les leds
int X = 54;
Adafruit_NeoPixel strip = Adafruit_NeoPixel(X, 6, NEO_GRB + NEO_KHZ800);

int Pin_Capteur_JourNuit = A0;
int Valeur_Lumiere;
int Pourcentage_Lumiere;
int nuit = 1000;
int jour = 0;
boolean couchersoleil = true;
boolean leversoleil = true;
int incr = 15;

// initialisation pour la pompe et le capteur hum sol

int Pin_Capteur_Humidite = A1;
int Valeur_Humidite;
int Pourcentage_Humidite;
int sec = 1023;
int humide = 200;
int Pin_Pompe = 13;

void setup() {
    // setup pour le capteur d'humidité air et servomoteur
    Serial.begin(9600);
    dht.begin();
    monServomoteur.attach(11);

    //setup pour le capteur jour/nuit et les leds
    strip.begin();
    strip.show();

    // setup pour la pompe et le capteur hum sol
    pinMode(Pin_Pompe, OUTPUT);
}
```

PRESENTATION DU CODE

```
void manageLuminosity()
{
    Valeur_Lumiere = analogRead(Pin_Capteur_JourNuit);
    Pourcentage_Lumiere = map(Valeur_Lumiere, nuit, jour, 0, 100);
    Serial.print("PourcentageJour = "); Serial.print(Pourcentage_Lumiere); Serial.println(" %")

    if (Pourcentage_Lumiere <= 40) {
        leversoleil = true;

        if (couchersoleil == true) {
            for (int i = 0; i < X; i++) {
                strip.setPixelColor(i, 244, 102, 27);
                delay(incr);
                strip.show();
            }

            for (int i = 0; i < X; i++) {
                strip.setPixelColor(i, 253, 108, 158);
                delay(incr);
                strip.show();
            }

            for (int i = 0; i < X; i++) {
                strip.setPixelColor(i, 253, 63, 146);
                delay(incr);
                strip.show();
            }

            for (int i = 0; i < X; i++) {
                strip.setPixelColor(i, 128, 0, 128);
                delay(incr);
                strip.show();
            }
        }
        couchersoleil = false;
    }
}
```

```
for (int i = 0; i < X; i++ ) {
    strip.setPixelColor(i, 128, 0, 128);
    strip.show();
    delay(incr);
}

}
else {
    couchersoleil = true;
    if (leversoleil == true) {
        for (int i = 0; i < X; i++ ) {
            strip.setPixelColor(i, 128, 0, 27);
            delay(incr);
            strip.show();
        }

        for (int i = 0; i < X; i++ ) {
            strip.setPixelColor(i, 253, 63, 146);
            delay(incr);
            strip.show();
        }

        for (int i = 0; i < X; i++ ) {
            strip.setPixelColor(i, 253, 108, 158);
            delay(incr);
            strip.show();
        }

        for (int i = 0; i < X; i++ ) {
            strip.setPixelColor(i, 244, 102, 27);
            delay(incr);
            strip.show();
        }

        leversoleil = false;
    }

    for (int i = 0; i < X; i++ ) {
        strip.setPixelColor(i, 255, 127, 0);
        strip.show();
        delay(incr);
    }
}
```

PRESENTATION DU CODE

```
void manageHumidity()
{
    digitalWrite(9,LOW);
    float PourcentageHumidite = dht.readHumidity();
    Serial.print("HumiditéAir = "); Serial.print(PourcentageHumidite); Serial.println(" %");
    Serial.println();
    if (PourcentageHumidite >= 90) {
        digitalWrite(9,HIGH);
    }
    else {
        monServomoteur.write(0);
        digitalWrite(9,LOW);
    }
}

#include <Servo.h>

Servo monServomoteur;

void setup() {
    monServomoteur.attach(11);
    pinMode(10,INPUT);
}

void loop() {
    if (digitalRead(10)==HIGH){
        monServomoteur.write(180);
    }
    else {
        monServomoteur.write(0);
    }
}
```

PRESENTATION DU CODE

```
void managePump()
{
    Valeur_Humidite = analogRead(Pin_Capteur_Humidite);
    Pourcentage_Humidite = map(Valeur_Humidite, sec, humide, 0, 100);
    Serial.print("PourcentageHumiditeSol :"); Serial.print(Pourcentage_Humidite); Serial.println(" %");
    if (Pourcentage_Humidite < 40) {
        digitalWrite(Pin_Pompe, HIGH);
        delay(4000);
        digitalWrite(Pin_Pompe, LOW);
    }
    else {
        digitalWrite(Pin_Pompe, LOW);
    }
    delay(60000);
}
```

```
void loop() {
    managePump();
    manageLuminosity();
    manageHumidity();
}
```

ORGANISATION

Evolution du planning

Semaines 1-2 Semaines 3-4 Semaines 5-6 Semaines 7-8

Etude des capteurs

Comprendre le fonctionnement

Codage

Programme pour chaque capteur

Construction partie basse

Découpe au laser, assemblage

Construction partie haute

Découpe au laser, assemblage

Assemblage

Branchements, placement matériel

Réécriture du code

Rassemblement et modification

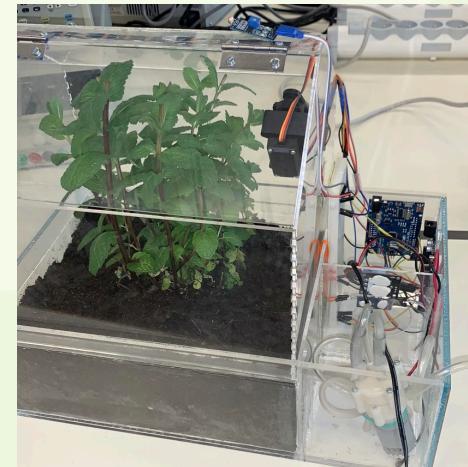
Plantation

Mise en fonctionnement de la serre

-respect des étapes prévues

-mauvaise prévision du temps nécessaire

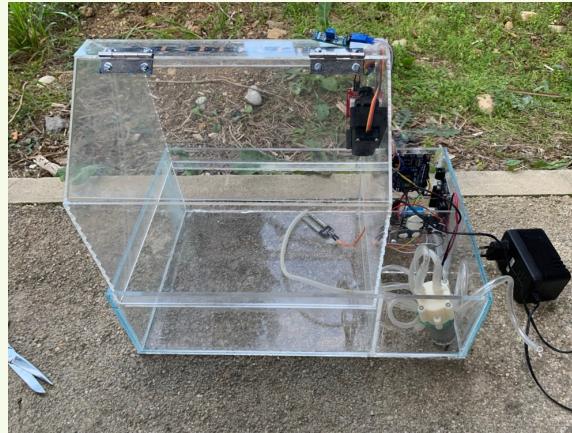
-répartition du travail



CONCLUSION

Ce que ce projet nous a apporté:

- Découvrir le travail d'un ingénieur et se confronter aux problèmes auxquels il fait face
- Travailler en équipe et surmonter les difficultés ensemble
- Savoir-faire et techniques en bricolage
- Apprentissage de nouveaux outils (Inkscape, Makercase...)
- Acquérir des connaissances dans le domaine Arduino



Perspective d'améliorer et d'ajouter de nouvelles fonctionnalités à notre serre et de voir son fonctionnement sur le long terme.