

Rapport de projet :

## *TP Solarius : Tracker solaire*



*Photographie d'un champ de tracker solaire*

## Sommaire :

### – Introduction :

- Contexte actuel de la production d'énergie en France
- Les objectifs du TP
- Définition et principe de fonctionnement du tracker solaire

### – Le montage du tracker solaire :

- Matériels et outils du Fablab ECAM à notre disposition
- Étapes de la construction.

### – Le système électronique :

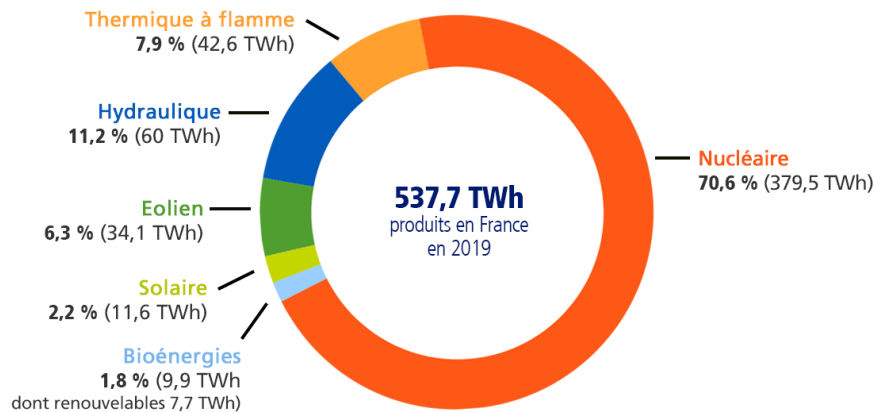
- L'Arduino
- Test de l'orientation du prototype
- Utilisation du capteur de luminosité
- Le code arduino

### – Conclusion

## Introduction :

### I – Contexte actuel de la production d'énergie en France :

La production d'énergie en France se décompose de la façon suivante :



La production française d'électricité en 2019

Source RTE - bilan électrique 2019

© EDF

Les énergies renouvelables comme l'hydraulique, l'éolien et le solaire ne représentent qu'environ 20% de l'énergie totale produite par an.  
Dans le cas de la production d'énergie solaire, la production est minoritaire.

Cependant les systèmes de productions d'énergie solaire se retrouvent assez fréquemment à petite échelle.

Par exemple :

- l'alimentation d'appareil nomade ;
- le système de rechargement de véhicule électrique ;
- l'alimentation d'habitation individuelle ...

Le plus souvent, les panneaux solaires sont sur un support fixe dans l'orientation varie notamment de la position géographique de l'installation.

### I – Les objectifs du TP :

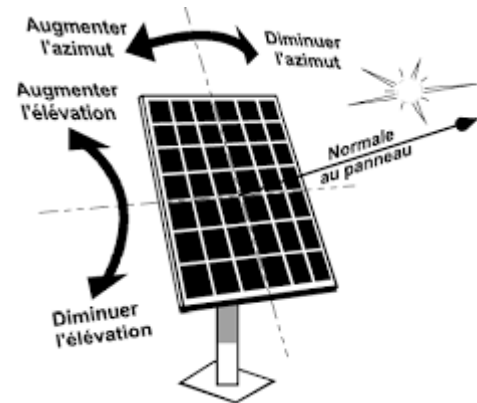
Pour le TP, les étapes suivantes doivent être :

- Le montage du prototype ;
- Le test de l'orientation du prototype ;
- Compréhension du fonctionnement et utilisation du capteur de luminosité ;
- Raccordement de la charge électrique sur le panneau solaire ;
- Mesure et enregistrement de la puissance de sortie.

### III – Définition et principe de fonctionnement du tracker solaire :

Un tracker solaire est une installation de production d'énergie utilisant le principe de l'héliostat. L'héliostat est le simple fait de suivre le trajet du soleil dans le but de recevoir un maximum de rayon solaire vers un point ou une surface.

Là où le panneau solaire est fixe, le tracker solaire, quand à lui, permet de pivoter le panneau solaire selon un ou deux axes dans le but de maximiser le rayonnement solaire, tout ceci pour augmenter la production d'énergie. Il est vrai que le soleil change constamment de position que ce soit dans la journée ou dans les différentes périodes de l'année. Il existe deux types de trackers, le mono-axiale et le bi-axiale. Suivre le soleil peut se faire sur deux axes, l'un d'est en ouest, c'est l'azimut et en hauteur selon les saisons et l'avancée des périodes. Le traqueur solaire se place en dessous du panneau ce qui va permettre de le déplacer suivant les deux axes. La puissance nécessaire au bon fonctionnement du tracker solaire est faible, de l'ordre de quelques watts.



## Le montage du tracker solaire :

#### - Matériels à disposition :

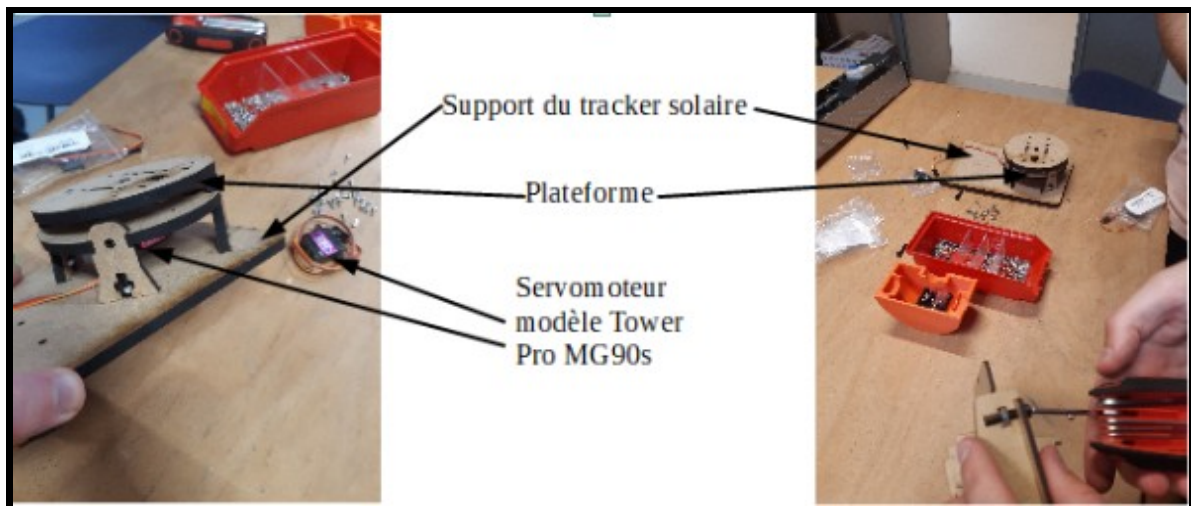
- 1 Microcontrôleur ADAFRUIT Circuit Playground Express (CPX) 3.3v
- 2 Servo Moteurs 3.3v
- 4 Capteurs de luminosités
- Circuit 3.3v, batteries
- maquette en bois issus du projet arduino réalisé par "Brown Dog Gadgets".

#### - Outils à disposition :

- Tous les outils du Fab Lab ECAM
- découpe laser, imprimante 3D.

Nous avons réalisé le montage du tracker solaire en moins de deux séances en suivant le prototype du professeur déjà construit. Plusieurs étapes lors de la construction du tracker solaire peuvent être mis en évidence pour expliquer comment nous avons procéder :

Première partie : construction de la partie support + plateforme pour faire pivoter le panneau solaire de gauche à droite.

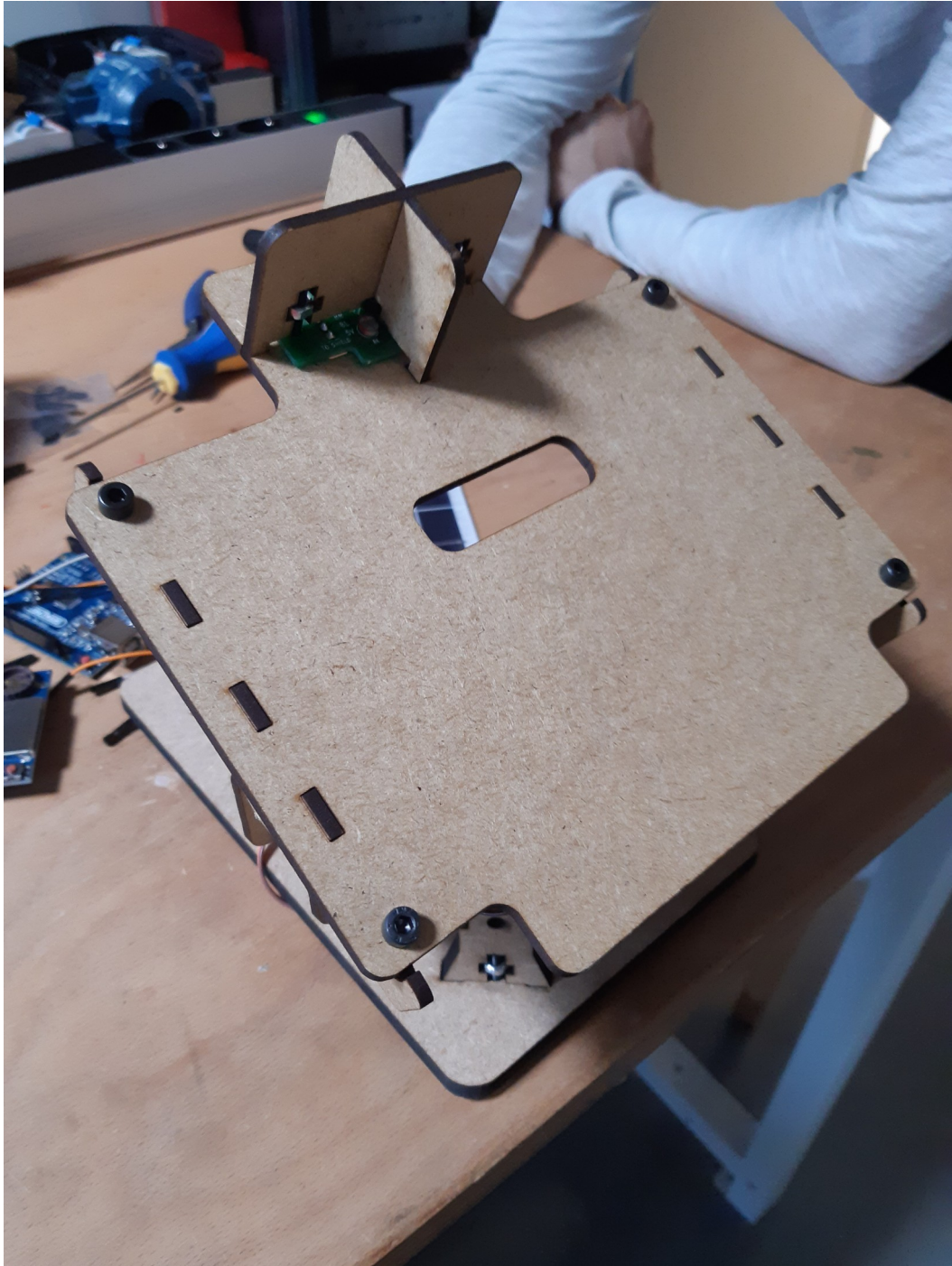


deuxième partie : brasures

troisième partie : assemblage de la partie supérieur du tracker solaire (+ axe verticale pour le faire changer de position)

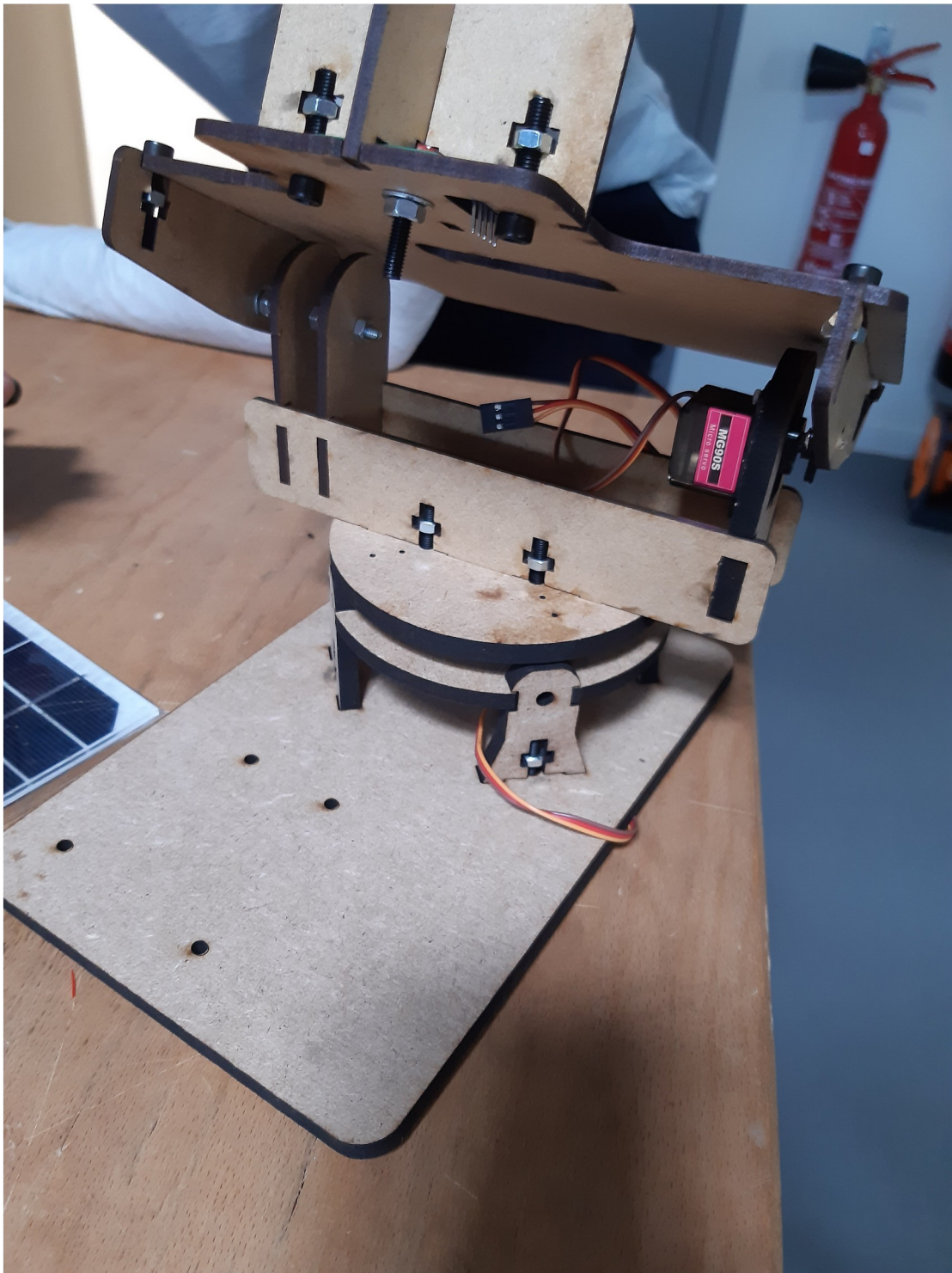
Ci-dessous, deux photos de notre montage du tracker solaire sans le panneau photovoltaïque :





*photo n°1 : vu de face du tracker solaire.*

Sur la plaque en bois observable au premier plan est fixé 4 capteurs de luminosité (sur la petite plaquette verte). Sur cette plaque en bois sera posé également le panneau photovoltaïque.



*photo n°2 : vue de derrière de racker solaire.*

L'arduino sera posé sur le support du tracker solaire où quatre trous ont été percés, visibles au premier plan de cette photo ci-dessus. Le montage une fois terminé, nous nous sommes rendu compte qu'un des deux servos ne fonctionnait pas. Nous avons donc perdu du temps pour refaire une partie du montage,



## Le système électronique :

### L'Arduino :

Une partie du travail de programmation a été réalisé lors de nos séances de TP d'informatique. Il était initialement prévu que le prototype fonctionne dans un environnement "Circuit playground" de chez Adafruit. En conséquence, l'alimentation de tous les appareils, comme les servos et le microcontrôleur, doivent donc se faire en 3.3 V.

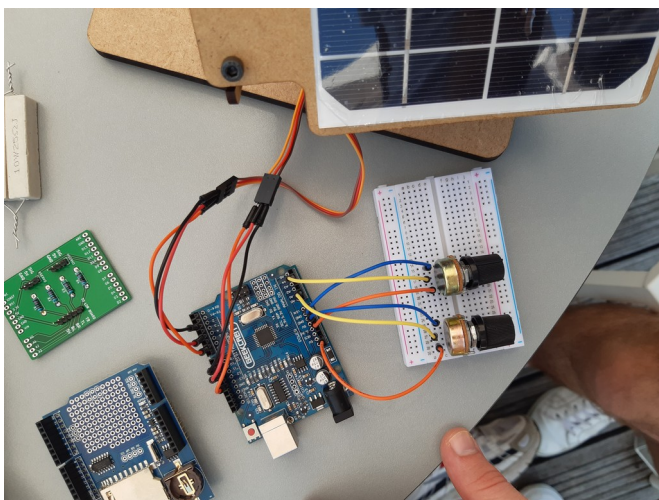
Cependant notre cahier des charges a évolué : suite à un problème d'approvisionnement, la tension d'alimentation doit impérativement passer de 3.3 V à 5 V. Ce qui implique que nous devons passer sur un environnement Arduino et donc de ne pas utiliser les codes pythons que nous avons réalisé en informatique pour faire fonctionner le tracker solaire.

La carte Arduino est la petite (5,33 x 6,85 cm) carte électronique bleu sur la photo ci-dessous. Elle est équipée d'un micro-contrôleur qui permet, à partir d'événements détectés par des capteurs, de programmer et commander des actionneurs. La carte Arduino est donc une interface programmable. Elle nous permet de donner l'information aux deux servomoteurs de tourner d'un certain angle en fonction des informations des capteurs de luminosités que l'Arduino reçoit.

### Test de l'orientation du prototype :

Une première utilisation de l'Arduino était de tester le bon fonctionnement des deux servomoteurs, sans pour le moment utilisé les capteurs de luminosités. Pour cela, nous avons utilisé deux potentiomètres positionnés sur une breadboard pour contrôler nos servomoteurs.

Ci-dessous est présentée une photo du câblage réalisé, mis en parallèle avec le schéma correspondant à celui-ci.



Photographie de notre câblage.

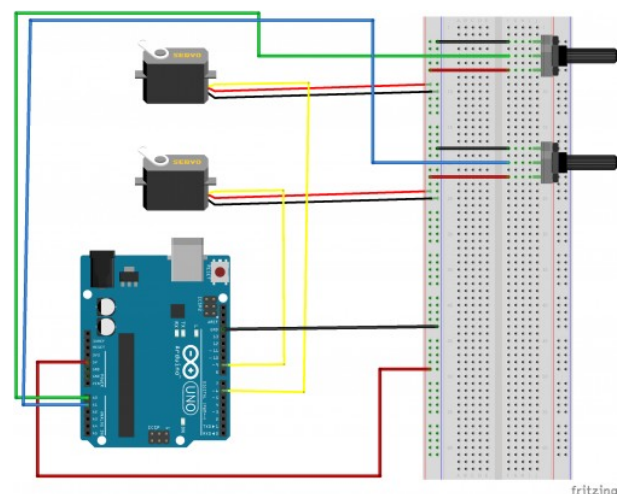
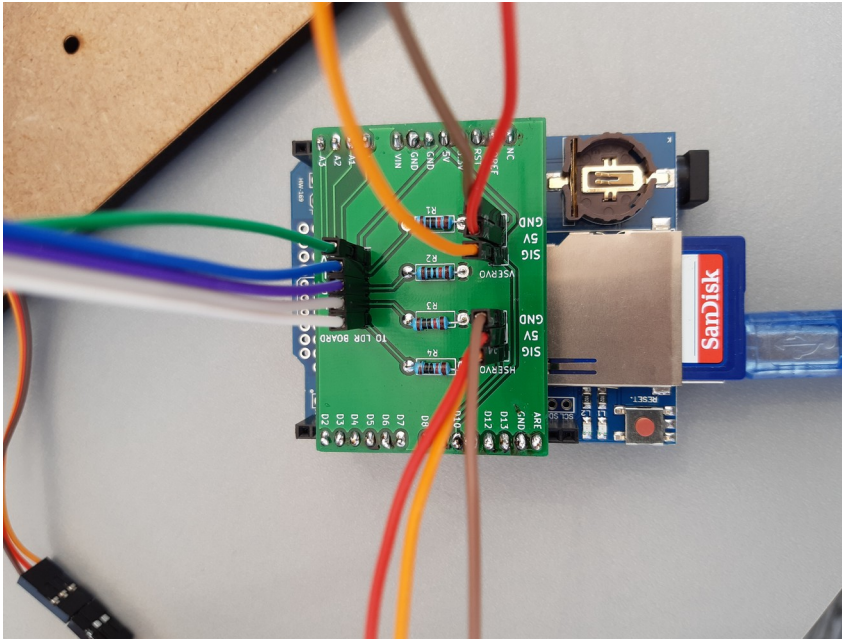
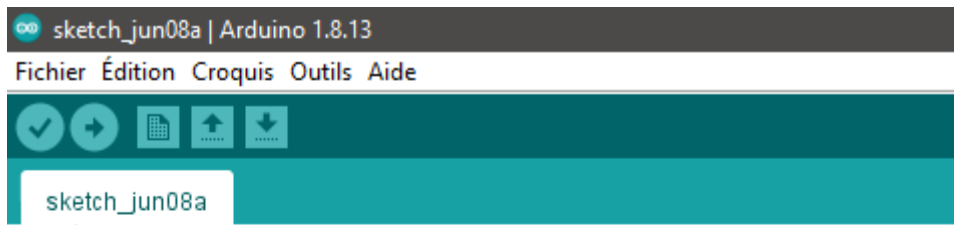


Schéma du circuit contenant deux servomoteurs, deux potentiomètres et l'Arduino.



**Utilisation du capteur de luminosité :**

Cette carte verte permet simplement de connecter le capteur de luminosité à l'arduino qui se trouve en dessous. La plaquette bleu qui surmonte l'arduino contient une carte SD qui permet d'enregistrer les intensités selon les horaires de la journée.

**Le code Arduino :**

```
1 #include <Servo.h> // include Servo library
2
3 // 180 horizontal MAX
4 Servo horizontal; // horizontal servo
5 int servoh = 90; // 90; // stand horizontal servo
6
7 int servohLimitHigh = 180;
8 int servohLimitLow = 65;
9
10 // 65 degrees MAX
11 Servo vertical; // vertical servo
12 int servov = 90; // 90; // stand vertical servo
13
14 int servovLimitHigh = 120;
15 int servovLimitLow = 15;
16
17
18 // LDR pin connections
19 // name = analogpin;
20 int ldrlt = 2; //LDR top left - BOTTOM LEFT <--- BDG
21 int ldrrt = 3; //LDR top right - BOTTOM RIGHT
22 int ldrlld = 0; //LDR down left - TOP LEFT
23 int ldrrd = 1; //ldr down right - TOP RIGHT
24
25 void setup()
26 {
27   Serial.begin(9600);
28   // servo connections
29   // name.attacht(pin);
```

```

30 horizontal.attach(9);
31 vertical.attach(10);
32 horizontal.write(180);
33 vertical.write(45);
34 delay(3000);
35 }
36
37 void loop()
38 {
39   int lt = analogRead(ldr1t); // top left
40   int rt = analogRead(ldr1r); // top right
41   int ld = analogRead(ldr1d); // down left
42   int rd = analogRead(ldr1r); // down right
43
44   // int dtime = analogRead(4)/20; // read potentiometers
45   // int tol = analogRead(5)/4;
46   int dtime = 10;
47   int tol = 50;
48
49   int avt = (lt + rt) / 2; // average value top
50   int avd = (ld + rd) / 2; // average value down
51   int avl = (lt + ld) / 2; // average value left
52   int avr = (rt + rd) / 2; // average value right
53
54   int dvert = avt - avd; // check the diffirence of up and down
55   int dhoriz = avl - avr; // check the diffirence og left and right

```

---

```

57
58 Serial.print(avt);
59 Serial.print(" ");
60 Serial.print(avd);
61 Serial.print(" ");
62 Serial.print(avl);
63 Serial.print(" ");
64 Serial.print(avr);
65 Serial.print(" ");
66 Serial.print(dtime);
67 Serial.print(" ");
68 Serial.print(tol);
69 Serial.println(" ");
70
71
72 if (-1*tol > dvert || dvert > tol) // check if the diffirence is in the tolerance else change vertical angle
73 {
74   if (avt > avd)
75   {
76     servov = ++servov;
77     if (servov > servovLimitHigh)
78     {
79       servov = servovLimitHigh;
80     }
81   }
82   else if (avt < avd)
83   {
84     servov = --servov;
85     if (servov < servovLimitLow)

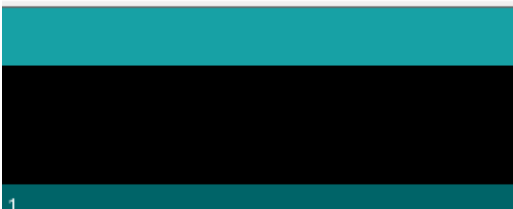
```

```

85     if (servov < servovLimitLow)
86     {
87         servov = servovLimitLow;
88     }
89 }
90 vertical.write(servov);
91 }
92
93 if (-1*tol > dhoriz || dhoriz > tol) // check if the difference is in the tolerance else change horizontal angle
94 {
95     if (avl > avr)
96     {
97         servoh = --servoh;
98         if (servoh < servohLimitLow)
99         {
100             servoh = servohLimitLow;
101         }
102     }
103     else if (avl < avr)
104     {
105         servoh = ++servoh;
106         if (servoh > servohLimitHigh)
107         {
108             servoh = servohLimitHigh;
109         }
110     }
111     else if (avl = avr)
112     {
113         // nothing
114     }
115     horizontal.write(servoh);
116 }
117 delay(dtime);
118
119 }

```

*capture d'écran du code arduino qui permet de donner l'information aux servomoteurs de tourner en fonction du capteur de luminosité*



## Conclusion :

Pour conclure, au cours du projet nous avons été confronté à de nombreuses difficultés qui commencent par la conception sans modèle plan. La deuxième difficulté fut de trouver un code adéquat et fonctionnel avec un bon câblage. Le tracker devait suivre correctement le positionnement du soleil. Par la suite la mise en place du datalogger n'a pas été une réussite. Bien que nous ayons eu des difficultés, des plans d'amélioration des performances du tracker solaire ont été imaginé comme par exemple utiliser un roulement a bille ou encore mettre en place un code qui met en veille tout le système dès le coucher du soleil et qui remet en veille au lever du soleil.

Par la fabrication de ce tracker solaire, nous avons exploré l'efficacité d'une installation disposant d'un système de suivi du soleil par rapport à une installation fixe. Enfin, par la même occasion, l'objectif de ce TP était de nous apprendre à connecter des éléments simples entre eux autour d'une carte Arduino. Il s'agissait pour le cas du tracker solaire de connecter un capteur de luminosité avec les deux servomoteurs par l'intermédiaire de la carte arduino. Enfin, une carte SD connectée à celui-ci permet d'enregistrer les données et nous pourrions mesurer et enregistrer la puissance de sortie du système.