

Projet Solarius

SOMMAIRE

I)Introduction.....	p2
II)Conception et réalisation du panneau solaire..	p4
IV)Bilan expérience personnelle.....	p11
IV)Conclusion.....	p12

I) Introduction

A) Les différents moyens de production d'énergie en France

Toutes les sources d'électricité proviennent de la transformation d'énergies primaires, directement présentes dans la nature (pétrole, bois, gaz naturel, etc.), en énergies secondaires produites par l'homme (comme la transformation de la chaleur en électricité dans les centrales thermiques). Une fois produite, l'énergie secondaire doit être transportée vers son lieu de consommation. L'électricité que reçoit le consommateur est appelée énergie finale.

Il existe 4 grands types d'énergie : mécanique, photovoltaïque, thermoélectrique et gravitationnelle. Certaines techniques de production sont anciennes de plus d'un siècle, comme les centrales au charbon, d'autres récentes, à l'instar de la géothermie pour de la biomasse. Certaines sont encore en cours de développement et pourraient bouleverser le paysage de la production d'électricité, comme les fours solaires, les usines marémotrices ou plus généralement les méthodes de gazéification de la biomasse. L'électricité produite provient pour 69 % du nucléaire ; pour 19 % des sources d'énergie renouvelables : production hydroélectrique : 11 %, éolien : 6 % et énergie solaire : 2 % ; et pour 11 % des centrales thermiques fossiles.

B) Les panneaux solaires

Un panneau solaire est un dispositif convertissant une partie du rayonnement solaire en énergie thermique ou électrique, grâce à des capteurs solaires thermiques ou photovoltaïques respectivement

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

C) Objectifs

Dans le cadre de ce TP, il est question d'explorer l'efficacité d'une installation disposant d'un système de suivi du soleil par rapport à une installation fixe. Les objectifs sont les suivants

- 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.

D) Cahier des charges

Le projet se décompose en deux scénarios :

- Le prototype devait fonctionner dans un environnement "Circuit playground" de chez Adafruit. Ainsi, l'alimentation de tous les appareils, comme les servomoteurs et le microcontrôleur, doit donc se faire en 3.3 V. Par suite d'un problème approvisionnement, la tension d'alimentation doit impérativement passer de 3.3 V et 5 V.

Il a donc été inévitable de passer sur un environnement Arduino ce qui implique que la partie programmation informatique se fasse majoritairement en C++ plutôt qu'en python.

- Modifier le prototype afin de contrôler manuellement l'orientation du panneau solaire.

- Intégrer le capteur de luminosité au fonctionnement du prototype pour ajuster la position du plateau (inclinaison et rotation), afin de suivre le soleil, de façon que le panneau soit toujours en face du soleil.

- Le panneau solaire doit être équipé d'une charge électrique ainsi que d'un système de mesure de puissance.

-Écrire les scripts qui correspondent à chacun des scénarios

-Enregistrement des données avec un dattaloger muni d'une carte SD.

E) Matériel nécessaire

- 2 Cartes Arduino Uno
- Capteur de lumières (photorésistances)
- Dattalogger
- Carte SD
- Câbles Arduino
- 2 Potentiomètres
- 2 Résistances de 25 Ω
- 1 cellule photovoltaïques
- Pièces détachées du prototype en découpe laser
- Quincailleries adéquat (vis, boulons)

II) Conception du panneau solaire

Dans un premier temps, il a été mis à disposition toutes les pièces détachées du prototype ainsi que tout l'équipement électronique nécessaire. Il a donc fallu monter le prototype sans s'occuper de la partie conception. Pour ce faire nous avons à disposition toutes les pièces détachées ainsi que tous les composants électroniques et enfin un prototype déjà assemblé.

Dès la première séance le prototype était monté et nous avons pu passer au premier scénario. Nous avons simplement collé la cellule à l'aide de patte adhésive afin de pouvoir la retirer à notre guise sans l'abimer. Nous avons eu comme idée de créer un support en découpe laser pour pouvoir le soutenir mais à cause du délai nous n'avons pas eu le temps puisqu'on a privilégié le fait d'avoir un prototype fonctionnel plutôt que design.

Premier scénario

Dans un second temps il a été question de le modifier (câblage et code) afin de le contrôler manuellement. Il a donc fallu prendre un potentiomètre pour chaque servomoteur du prototype, un pour le servomoteur s'occupant de l'inclinaison du panneau solaire et un autre pour la rotation, de se munir d'une breadboard pour les câblages et du circuit imprimé pour brancher les capteurs et les servomoteurs. Nous n'étions pas obligés de brancher les capteurs dès cette étape mais cela nous a fait gagner du temps dans le montage pour la partie suivante.

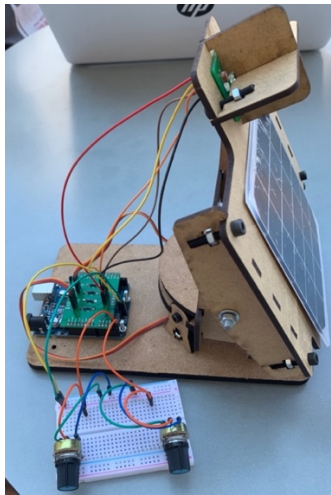


Photo du câblage pour le contrôle manuel des servomoteurs

Code Arduino :

```
#include <Servo.h>

Servo myservo;  // create servo object to control a servo
Servo myservo2;

int potpin = 4;  // analog pin used to connect the potentiometer
int val;         // variable to read the value from the analog pin
int potpin2 = 5;
int valu;
void setup() {
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
  myservo2.attach(10);
}

void loop() {
  val = analogRead(potpin);          // reads the value of the potentiometer (value between 0 and 1023)
  val = map(val, 0, 1023, 0, 180);    // scale it to use it with the servo (value between 0 and 180)
  myservo.write(val);                // sets the servo position according to the scaled value
  delay(15);                         // waits for the servo to get there
  valu = analogRead(potpin2);
  valu = map(valu, 0, 1023, 0, 180);
  myservo2.write(valu);
  delay(15);
}
```

Second scénario

On a intégré les capteurs de luminosité dans le code et dans le câblage afin de permettre au prototype de bouger de telle sorte à ce que la cellule photovoltaïque soit toujours face au soleil.

La partie câblage a été extrêmement simple pour cette partie. Nous avons simplement branché les servomoteurs et les capteurs au circuit imprimé en enlevant toute la partie câblage du control manuel il nous reste simplement la carte Arduino avec le circuit imprimé sur lequel on a branché les servomoteurs et les capteurs.

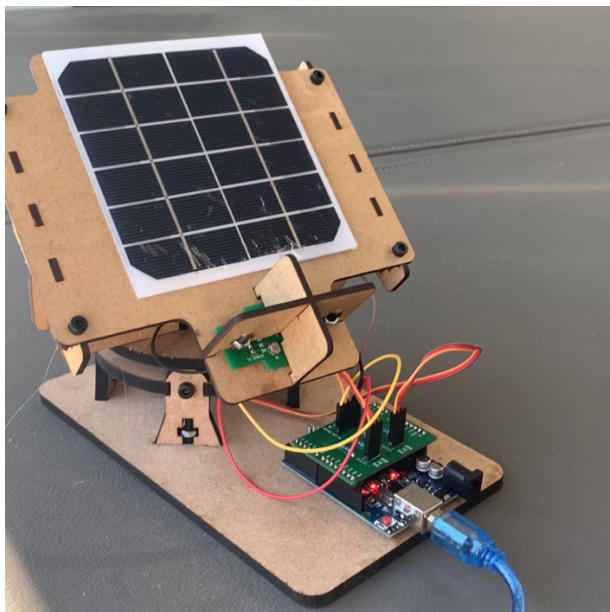


Photo du câblage

MALERBA FINIDORI

Script Arduino :

```
#!/usr/bin/C++
//  -*- coding: utf-8 -*-
// Date: Sunday 6 June 2021 09h02:01 CEST
// Author: Fabio Malerba
// Last updated by: Fabio Malerba
// Last updated time: Sunday 6 June 2021 10:20:20 CEST
// Description: Suivre le soleil

/*
  Ce programme permet en faisant la moyenne des capteurs
  d'adapter la position des servos et donc de la cellule
  photovoltaïque pour suivre le soleil
*/

#include <Servo.h>

Servo servoh;
Servo servov;

int horizontal = 0;
int vertical = 0;

int top_left;
int top_right;
int bottom_left;
int bottom_right;

int moyenne_1;
int moyenne_2;
int moyenne_3;
int moyenne_4;

int difference_horizontale;
int difference_veticale;
const int limite = 30 ;

void setup()
{
  servoh.attach(10);
  servov.attach(9);

  Serial.begin(9600);
}

void loop() {

  top_left = analogRead(A2); //lecture de la donnée des capteurs
  top_right = analogRead(A3);
  bottom_left = analogRead(A0);
  bottom_right = analogRead(A1);
  vertical = servov.read();
  horizontal = servoh.read();

  Serial.println(horizontal);

  moyenne_2 = ( bottom_left + bottom_right )/2; //moyenne des capteurs du bas
  moyenne_1 = (top_left + top_right)/2; //moyenne des capteurs du haut
  moyenne_3 = ( top_left + bottom_left) / 2; //moyenne des capteurs de gauche
  moyenne_4 = ( top_right + bottom_right) / 2; //moyenne des capteurs de droite

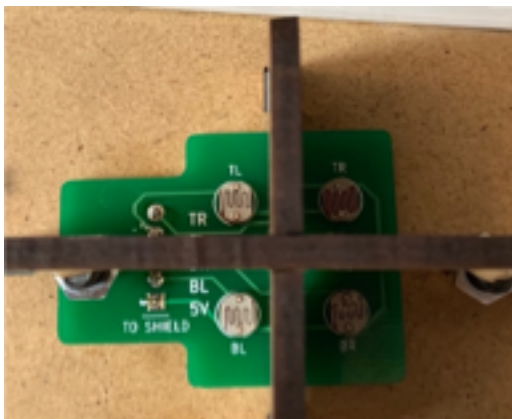
  difference_horizontale= abs(moyenne_2 - moyenne_1); //On regarde l'écart
  difference_veticale = abs(moyenne_3 - moyenne_4);

  if ((moyenne_1 > moyenne_2) && (difference_horizontale > limite))
  {
    if (vertical > 0)
    {
      vertical = vertical - 1;
      servov.write(vertical);
      delay(5);
    }
  }
}
```

MALERBA FINIDORI

```
}  
  
if((moyenne_2 > moyenne_1) && (difference_horizontale > limite))  
{  
  if (vertical < 180)  
  {  
    vertical = vertical + 1;  
    servov.write(vertical);  
    delay(5);  
  }  
}  
  
if ((moyenne_4 < moyenne_3) && (difference_vetiale > limite))  
{  
  if (horizontal > 0)  
  {  
    horizontal = horizontal - 1;  
    servoh.write(horizontal);  
    delay(5);  
  }  
}  
  
if((moyenne_3 < moyenne_4) && (difference_vetiale > limite))  
{  
  if (horizontal < 180)  
  {  
    horizontal = horizontal + 1;  
    servoh.write(horizontal);  
    delay(5);  
  }  
}  
delay(10);  
}
```

Comment fonctionne le capteur ?



Le capteur est composé de 4 photorésistances.

La photorésistance est un composant utilisé en électronique qui a la particularité d'avoir une résistivité variable en fonction de la lumière ambiante. Plus il y a de lumière plus la résistivité diminue et inversement, avec beaucoup de lumière sa résistivité augmente.

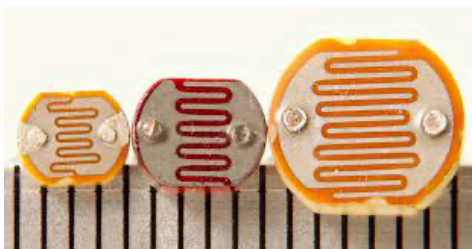


Photo de Photorésistances

Une photorésistance est composée d'un semi-conducteur à haute résistivité d'un moteur et d'une résistance. Si la lumière incidente est de fréquence suffisamment élevée (donc d'une longueur d'onde inférieure à la longueur d'onde seuil), elle transporte une énergie importante.

De plus la partie sensible du capteur est une piste de sulfure de cadmium en forme de serpent : l'énergie lumineuse reçue déclenche une augmentation de porteurs de charges libres dans ce matériau, de sorte que sa résistance électrique évolue.

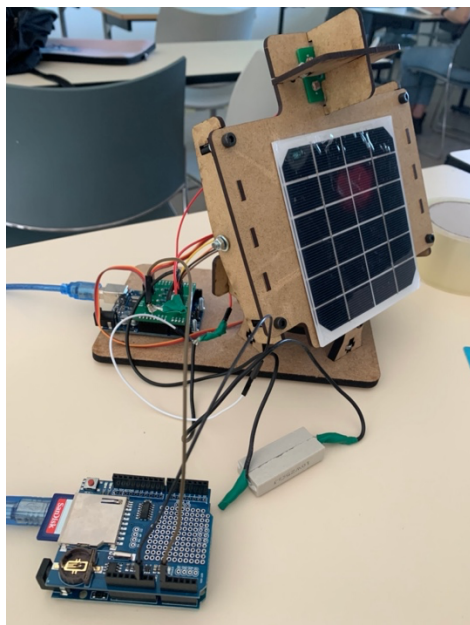
Charge électrique et proposition de câblage

Le panneau solaire doit être équipé d'une charge électrique ainsi que d'un système de mesure de puissance.

Cette charge électrique est modélisée par une association de deux résistances de 25Ω placées en parallèle. Ainsi on a une résistance équivalente de 12.5Ω

En revanche le capteur de puissance ne fait pas parti du matériel qui nous était mis à disposition mais cela reste réalisable en modifiant le câblage et en créant un code Arduino

De plus sachant que le panneau sort trop de tension (7V alors que l'Arduino ne peut qu'avoir 5V). Ces deux résistances en parallèles que l'on a brasées et insérées entre l'Arduino et le panneau. Nous avons également eu besoin d'une autre carte Arduino parce que toutes les entrées analogiques étaient occupées sur la premier carte, il était donc impossible de brancher les résistances.



```
int volt;  
float resistance = 12.5;  
float puissance;  
  
void setup()  
{  
  Serial.begin(9600);  
}  
  
void loop() {  
  volt = analogRead(A0);  
  volt = map(volt,0,1023,0,5);  
  puissance = (volt*volt)/resistance;  
  Serial.println(puissance);  
}
```

Photo du câblage ainsi que du code

Dans la nouvelle carte Arduino nous avons insérer un programme permettant de calculer la puissance électrique fournie par le panneau grâce à la relation $P = (U^2/R)$.

Enregistrement des données

Pour pallier le cahier des charges il a fallu enregistrer la mesure de la puissance électrique fournie par le panneau.

Afin de se rendre compte de la différence de performance entre la solution fixe et orientable, il est indispensable d'enregistrer la mesure de la puissance électrique trouvée dans la partie précédente.

Le montage est exactement le même que celui pour la mesure de puissance dans un souci d'efficacité. En ce qui concerne cette partie nous voulions récupérer les données du datalogger dans une carte SD via un fichier CSV afin d'avoir les valeurs sous forme de tableau et pour les ouvrir facilement afin de les traiter de la façon la plus rapide et plus simple possible.

Code pour la récupération de données via le catalogue :

```
#include "SD.h"
#include "SPI.h"
//
//the hardware CS pin (10 on most Arduino boards,
// 53 on the Mega) must be left as an output or the SD
// library functions will not work.
const int CSpin = 10;
String dataString = ""; // holds the data to be written to the SD card
float sensorReading1 = 0.00; // value read from your first sensor
float sensorReading2 = 0.00; // value read from your second sensor
float sensorReading3 = 0.00; // value read from your third sensor
File sensorData;
//
//
void setup()
{
  // Open serial communications
  Serial.begin(9600);
  Serial.print("Initializing SD card...");
  pinMode(CSpin, OUTPUT);
  //
  // see if the card is present and can be initialized:
  if (!SD.begin(CSpin)) {
    Serial.println("Card failed, or not present");
    // don't do anything more:
    return;
  }
  Serial.println("card initialized.");
}
//
void loop(){
  // build the data string
  dataString = String(sensorReading1) + "," + String(sensorReading2) + "," + Str
  saveData(); // save to SD card
  delay(60000); // delay before next write to SD Card, adjust as required
}

//
void saveData(){
  if(SD.exists("data.csv")){ // check the card is still there
    // now append new data file
    sensorData = SD.open("data.csv", FILE_WRITE);
    if (sensorData){
      sensorData.println(dataString);
      sensorData.close(); // close the file
    }
  }
  else{
    Serial.println("Error writing to file !");
  }
}
```

Test, problèmes et solutions apportées

Pour les phases de test, les câblages avaient pour chaque partie été préalablement fait de notre côté grâce à des schémas puis réalisés en vrai en classe. Grace à cela nous avons pu suivre la démarche de fabrication dans les délais. Les phases de test étaient réalisées pendant les séances. Pour le contrôle manuel cela a été très rapide notre prototype a directement été fonctionnel et les vidéos déposées le prouve. Il en est de même pour le suivre du soleil avec capteurs mais le programme était plus fastidieux nous avons dû faire des retouches dessus pendant les phases d'essais.

Pour les dernières parties nous avons eu plus de mal. Pour la partie affichage de puissance, c'était trop binaire, en utilisant le programme et le moniteurs série on remarquait quand exposant le prototype la puissance admettait une valeur. Et que lorsqu'on le cachait, elle en admettait un autre, il n'avait pas de dégression c'était direct et puis apparaissait des 0 de partout. Nous n'avons pas trouvé de solution à ce problème.

IV Bilan expérience personnel

En ce qui concerne la méthodologie de test nous avons procéder de la manière suivante :
Les programmes étaient réfléchis et fait à l'avance en dehors des heures de Travaux Pratiques afin d'utiliser nos heures au Fablab uniquement pour des tests.

Pour le montage du prototype toutes les pièces détachées étaient à disposition. Nous avons un prototype terminer comme exemples il a donc été plutôt simple de le monter.

Concernant le scénario permettant de gérer le prototype manuellement cela n'a pas été la source la plus importante de nos problèmes durant ce projet. Le programme était plutôt basique tout comme les branchements. En revanche pour ce qui est de la partie où il fallait prendre en compte les capteurs. En effet ces capteurs photorésistances étaient nouveaux pour nous et il a fallu comprendre leur fonctionnement et leur utilité pour avoir un programme qui marche.

Réaliser un programme fonctionnel en intégrant les capteurs a été vraiment fastidieux. Nous ne connaissions rien à propos du maniement du code pour le capteur. Il a fallu effectuer un bon nombre de recherches afin de comprendre et d'en déduire le programme adapté à notre prototype.

Pour la partie câblage et mesure de puissance il n'a eu aucun problème. Cela a juste prit un peu de temps pour bien souder les résistances au panneau. Le programme qui en découle est plutôt facile puisqu'il s'agit d'une simple formule de puissance à retranscrire en programme Arduino. Enfin la partie la plus fastidieuse est celle de la récolte de donnée avec le datalogger. C'était quelque chose de tout nouveau et pour aboutir au bon programme avec une démarche cohérente étant donné le délai que nous avions était presque infaisable. Nous n'avons pas trouvé de solution à ce problème malgré les différentes routes que nous avons empruntée pour le régler. Même si l'on a bien dégorgé le problème, le délai très court est la cause la plus importante de notre échec.

IV) Conclusion

Dans l'ensemble, le projet s'est plutôt bien déroulé, à part la partie enregistrement de donnée nous n'avons pas rencontré de difficultés insurmontables. La plus grosse difficulté était dans le sens ou la plupart des choses étaient nouvelle (datalogger, capteur de lumière) il a donc fallu comprendre le fonctionnement de chacun de ces objets pour ensuite créer le code adéquat avec la solution que nous avons trouvé pour compléter le cahier des charges le tout dans un délais réduit.