

## BIBLIOGRAPHIE PROJET - PEIP2

*Année scolaire 2020-2021*

### ***THE MOOD LAMP***

**Étudiants : Romane KWIATKOWSKI, Julie PERRET**

**Encadrants : Pascal MASSON**

École Polytechnique Universitaire de Nice Sophia-Antipolis,  
1645 route des Lucioles, Parc de Sophia Antipolis, 06410 BIOT



# SOMMAIRE

I.	Introduction.....	5
II.	État de l'art : .....	6
III.	Le moteur .....	8
A.	Les objectifs et attentes.....	8
B.	Les différentes options possibles .....	8
1.	Le moteur à courant continu avec le module L293D.....	8
2.	Le servomoteur.....	9
3.	Le servomoteur à rotation continue (RC) .....	10
C.	Conclusion.....	10
IV.	Les lumières.....	11
A.	Objectifs et attentes .....	11
B.	Les différentes options .....	11
1.	Les Neopixels .....	11
2.	Le module LED RVB.....	14
C.	Conclusion : .....	14
V.	Les pétales de notre lampe .....	15
A.	Objectifs et attentes .....	15
B.	Les différentes options .....	15
1.	Le plexiglass .....	15
2.	L'impression 3D .....	15
C.	Conclusion .....	16
VI.	Le principe du mécanisme d'ouverture/ fermeture .....	16
A.	Objectifs et attentes .....	16
B.	Les différentes options .....	16
1.	Les fils .....	16
2.	La tige rétractable.....	17
C.	Conclusion .....	17
VII.	La connexion Bluetooth.....	17
A.	Objectifs et attentes .....	17
B.	Les différentes options .....	18
1.	Le module RF 433 mHz .....	18
2.	Le module HC-06 .....	18
3.	Un dérivé, le module HM-10 .....	20
C.	Conclusion : .....	20
VIII.	Conclusion générale .....	21



## I. Introduction

Pour notre projet d'Arduino, nous souhaitons réaliser une lampe d'ambiance, que nous avons nommée The Mood Lamp.

Notre lampe sera en forme de fleur (tulipe), les pétales pourront s'ouvrir et se fermer (à l'allumage et la fermeture de la lampe).

Notre lampe sera créatrice d'ambiance, elle pourra changer de couleur en fonction d'un modèle prédéfini ou d'un modèle défini par l'utilisateur.

Notre lampe sera décorative et apaisante ou relaxante selon l'ambiance choisie. De plus, l'utilisateur pourra choisir ou créer ses ambiances grâce à une application sur son smartphone.

Pour choisir les meilleurs éléments à utiliser pour concevoir notre Mood Lamp, nous allons nous intéresser au moteur, à la lumière, aux pétales, au mécanisme d'ouverture / de fermeture et à la connexion à un smartphone.

## II. État de l'art :

Les objets relaxants basés sur un effet de lumière sont abondants sur le marché. Il y en a de toutes sortes et de toutes formes. En voici quelques exemples :



photo II.1

« Fleur Lotus Relaxante »,  
du site Sept-Chakras



photo II.2

La lampe « Arbre de Rose »,  
du site Sourire Zen



photo II.3

« Lampe Ikigai »,  
⊕

De nombreux projets Arduino sont créés dans le monde chaque jour. Nous avons trouvé une multitude de projets ressemblants au notre. Nous nous sommes donc inspirés de ces projets afin de réaliser le nôtre. En voici quelques exemples :

### Un projet avec du plexiglass :

[https://www.youtube.com/watch?v=pQwgZwrXfhc&ab\\_channel=RobertRobert](https://www.youtube.com/watch?v=pQwgZwrXfhc&ab_channel=RobertRobert)

Ce projet ne ressemble pas à une fleur, pourtant, le mécanisme peut être intéressant. On y comprend comment fonctionnent les néopixels et comment faire varier les couleurs et les intensités lumineuses.



photo II.4

### **The Fairy Lantern**

<https://www.instructables.com/Fairy-Lantern/>

Cette lanterne inspirée d'une des fleurs les plus fragiles du monde, le lys globe, est très similaire au projet que nous souhaitons réaliser. Cette lanterne nécessite de l'impression 3D afin de réaliser la structure principale. Elle possède une LED RVB afin de lui donner toutes les couleurs possibles.



photo II.5

### **La Ever Blooming Mechanical Tulip :**

<https://www.instructables.com/Ever-Blooming-Mechanical-Tulip/>

Parmi tous les projets que nous avons pu trouver, celui-là est de loin le plus proche de notre idée de départ.

Les pétales sont assez transparents pour y laisser passer la lumière. De plus, la lumière utilisée permet de recréer des ambiances. Cette fleur possède un mécanisme ressemblant à celui que nous voulions utiliser. De plus, les LED présentes le long des pétales donnent un très beau rendu. C'est de ce projet dont nous allons le plus nous inspirer pour créer notre Mood Lamp.

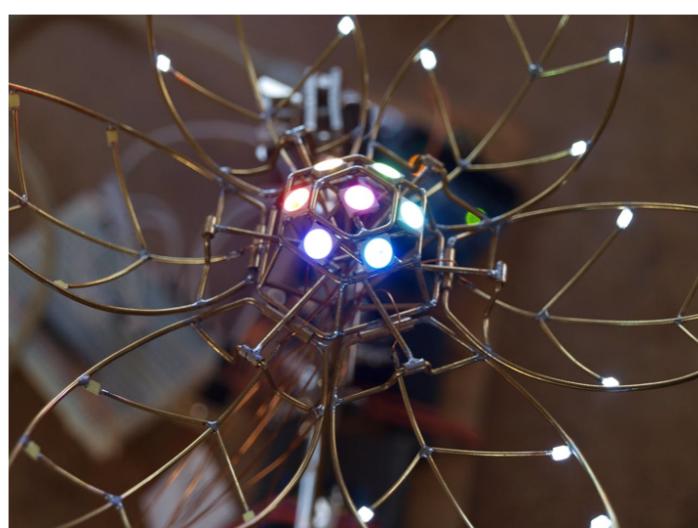


photo II.6

### III. Le moteur

#### A. Les objectifs et attentes

Notre fleur d'ambiance peut s'ouvrir et se fermer à notre demande, il faut donc qu'elle soit motorisée. Le moteur, avec un module si cela est nécessaire, devra être capable de :

- Tourner dans les deux sens (pour ouvrir puis fermer la fleur).
- S'arrêter dans une certaine position (pour éviter le risque de casse).
- Tourner à une vitesse donnée (pour que l'ouverture et la fermeture soient délicates).

Nous allons voir quelles options s'offrent à nous.

#### B. Les différentes options possibles

##### 1. Le moteur à courant continu avec le module L293D

Source : <https://electrotoile.eu/arduino-moteur-DC-shield.php#004>

Pour contrôler le sens de rotation, nous pourrions utiliser un pont en H (moteur + 4 interrupteurs) mais cela présente deux inconvénients :

- Montage encombrant avec les 4 relais
- Variation de la vitesse impossible

Il existe d'autres méthodes pour faire cela, dont le composant L293D (lien Amazon <https://amzn.to/2NenDqH> ; lien dataSheet <https://www.ti.com/lit/ds/symlink/l293.pdf>). Ce composant est équipé d'un pont en H (pour inverser le sens de rotation du moteur) et permet aussi de recevoir des informations pour varier la vitesse (via une borne qui est reliée à un montage PWM).

*Câblage L293D, moteur, pile 9V (avec adaptateur) et Arduino*

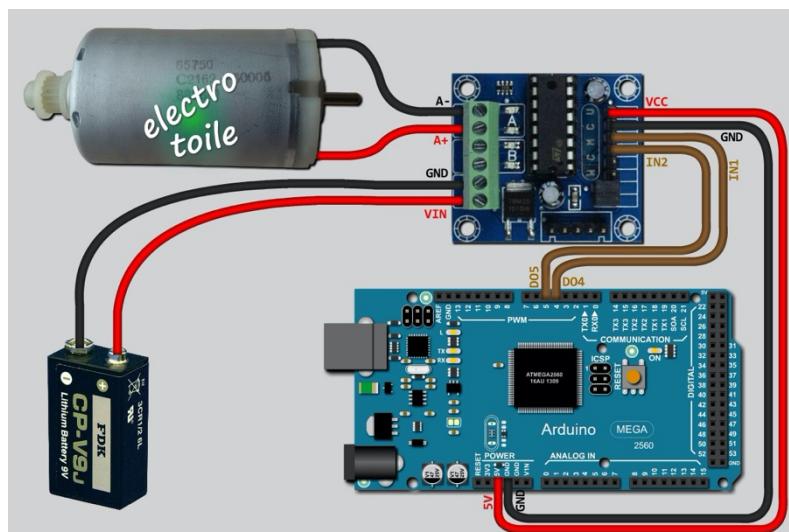


photo III.1

Pour l'utilisation de ce module, il n'y a pas de librairie spéciale à importer. Il faut seulement déclarer les deux IO (une qui représentera le sens horaire du moteur et l'autre le sens antihoraire) comme sorties digitales. Ensuite, le contrôle du sens se fait par *digitalWrite* en affectant LOW ou HIGH.

L'utilisation du PWM (pour contrôler la vitesse du moteur et donc des pétales) ayant été étudié en classe, nous n'allons pas expliquer son fonctionnement en détail.

## 2. Le servomoteur

Le servomoteur est un petit moteur permettant de répondre à toutes nos attentes.

Sources :

[https://www.youtube.com/watch?v=4w3\\_zbaPdcs](https://www.youtube.com/watch?v=4w3_zbaPdcs)

<http://users.polytech.unice.fr/~pmasson/Enseignement/Elements%20de%20robotique%20avec%20arduino%20-%20Moteurs%20-%20Projection%20-%20MASSON.pdf>

Le servomoteur permet d'arrêter le moteur dans une position fixe. C'est un moteur à courant continu.

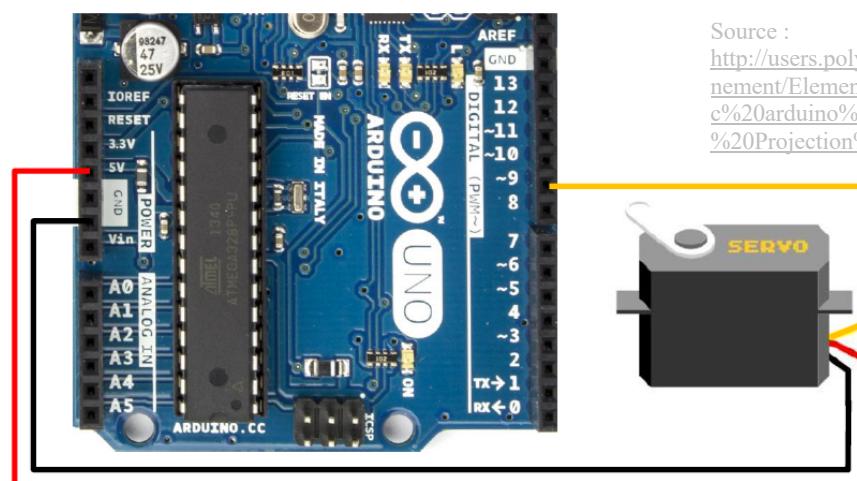
Il possède un réducteur qui permet d'augmenter son couple (c'est-à-dire augmenter sa force). L'angle de positionnement du servomoteur va de 0° à 180° (respectivement de 1ms à 2ms si on exprime les angles en microsecondes). Nous verrons plus loin les fonctions qu'il faut utiliser. Le servomoteur qui correspond le mieux à notre projet est le *micro-servo 9g* car il est petit et assez puissant pour ce que nous voulons faire.



photo III.2

Le branchement sera réalisé grâce aux trois fils du servo :

- Le marron pour la masse (GND de l'Arduino)
- Le rouge pour 5V (Vcc de l'Arduino)
- Le jaune pour la commande (peut être reliée à une sortie PWM (9 par exemple))



Source :  
<http://users.polytech.unice.fr/~pmasson/Enseignement/Elements%20de%20robotique%20avec%20arduino%20-%20Moteurs%20-%20Projection%20-%20MASSON.pdf>

schéma III.3

## Contrôle de l'angle du servomoteur grâce à un montage PWM :

Il est possible de contrôler l'angle du servo en envoyant une impulsion plus ou moins longue grâce au PWM. L'impulsion à fournir doit être comprise entre 1ms et 2ms. Suite à cette impulsion, le servomoteur se positionnera dans la position donnée et s'arrêtera jusqu'à la prochaine impulsion.

## Contrôle de l'angle du servomoteur grâce à la librairie “Servo” :

Cette librairie permet de ne pas utiliser de montage PWM. Elle fonctionne de la manière suivante :

- Il faut d'abord importer la librairie servo :

```
#include <Servo.h>
```

- Déclarer quelle IO a été choisie (9 par exemple) :

```
servo.attach(9);
```

ou

```
servo.attach(9,min,max); //avec min et max les angles en microsecondes
```

- Pour attribuer un angle au servo :

```
servo.write(90); //en degré
```

```
servo.writeMicroseconds(..); //en microsecondes
```

### 3. Le servomoteur à rotation continue (RC)

Le servomoteur à rotation continue est une version dérivée du servomoteur classique. L'angle qu'on lui donne ne correspond plus à un angle de rotation mais à une vitesse de rotation. Entre 0° et 89° le servo rc tourne dans le sens antihoraire de plus en plus lentement, à 90° il est à l'arrêt et entre 91° et 180° il tourne dans le sens horaire de plus en plus vite.

[http://www.planete-sciences.org/iledefrance/images/psidf/pdf/Fiche\\_F4\\_Commander\\_des\\_servomoteurs\\_classiques\\_et\\_a\\_rotation\\_continue.pdf](http://www.planete-sciences.org/iledefrance/images/psidf/pdf/Fiche_F4_Commander_des_servomoteurs_classiques_et_a_rotation_continue.pdf)

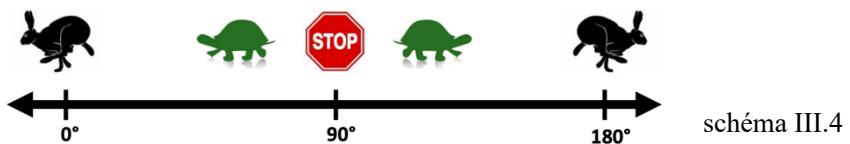


schéma III.4

Il existe aussi une librairie, qui est : <Servo.h>. Elle comprend les mêmes fonctions que pour le servomoteur classique.

### C. Conclusion

A la différence du module L293D et du servomoteur à rotation continue (qui tourne sans s'arrêter), le servomoteur classique est ce qui correspond le mieux à notre projet car il répond à toutes nos attentes (contrôle de son angle et arrêt quand l'angle est atteint).

Nous pensons utiliser la librairie Servo car elle est plus simple d'utilisation et ne comprend pas de câblages supplémentaires.

## IV. Les lumières

### A. Objectifs et attentes

Notre fleur devra changer de couleur en fonction de l'ambiance choisie par l'utilisateur. Nous voulons que chaque LED puisse changer de couleur et d'intensité. C'est pour cela que nous ne pouvons pas choisir les LED vues en cours car elles ne peuvent changer que d'intensité. Leur couleur est prédéfinie.

Nous avons donc choisi de nous intéresser, dans un premier temps, aux Neopixels. Puis, dans un second temps, nous verrons une alternative aux néopixels : le module LED RVB.

### B. Les différentes options

#### 1. Les Neopixels

##### Qu'est-ce qu'un neopixel ?

Le neopixel est un terme inventé chez Adafruit pour les brins RVB et RGBW de LED adressables à un seul fil, peuvent être configurés de différentes manières.

Il y a des limites au nombre de ces LED qui peuvent être contrôlées par un seul microprocesseur. Les deux contraintes majeures sont la mémoire disponible et la puissance. Les néopixels se composent de 3 ou 4 LED et d'une puce pilote dédiée. Ils consomment jusqu'à 60 milliampères de courant lorsqu'ils sont entièrement éclairés.

##### Principe de fonctionnement d'un ruban (ou anneau) de neopixel:

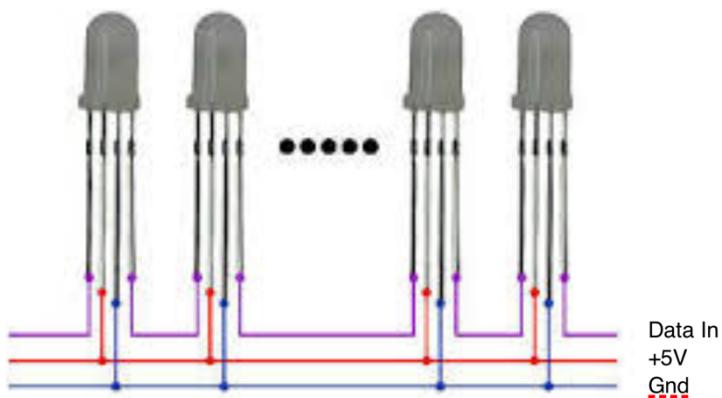


schéma IV.1

Le fil violet (Data In) passe dans chacune des LEDs. Il contient deux informations : le numéro du neopixel et un numéro correspondant à une couleur. Si le premier numéro ne correspond pas à celui du neopixel dans laquelle passe l'information, rien ne se passe et l'information continue de circuler dans la bande.

Si, au contraire, les deux numéros sont identiques, le neopixel concerné s'allumera dans la couleur donnée par le second numéro. C'est pour cela que seul un fil est nécessaire. Il contient toutes les informations pour chacun des neopixels.

## Plusieurs types de Neopixels existent :

Leur forme peut varier :

- en bande
- en anneau

Pour notre projet, il est préférable d'utiliser les Neopixels en anneau car il sera placé à la base de notre fleur.

Il existe plusieurs tailles pour ces anneaux. Il en existe pour 7, 8, 12, 16, 24, 35, 45 neopixels, ou encore bien d'autres tailles. Cependant, nos pétales vont être placées autour des neopixels afin que la lumière s'y reflète. Il faut donc que notre anneau soit assez petit pour que chaque pétale ait la place. Si nous prenons l'anneau de 45 neopixels par exemple, sachant que nous voulons 6 ou 7 pétales, l'anneau va être beaucoup trop grand. Le diamètre de cet anneau est de 12,0 cm.

Le meilleur choix à faire ici est de prendre un anneau de 7 neopixels. En effet, son diamètre de 2,3 cm va nous permettre de placer nos 7 pétales correctement.

## Alimentation de l'anneau :

Un neopixel est composé de 3 LED (une rouge, une bleue et une verte), afin de pouvoir recréer n'importe quelle couleur.

Chaque LED a besoin de 20 mA pour fonctionner. Chaque neopixel aura donc besoin de 60 mA pour s'allumer. Nous possédons 7 neopixels. Il nous faudra donc 0,42 A pour faire fonctionner notre anneau.

## Comment brancher cet anneau :

Un anneau de neopixel est composé de 4 PIN :

- un PIN +5V
- un PIN GND
- un PIN DI
- un PIN DO

Le PIN +5V est le PIN qui sert à alimenter l'anneau. C'est ici que l'on va brancher le 5V de notre carte.

Le PIN GND est la masse.

Le PIN DI signifie Data In. C'est ici que les commandes pour allumer les neopixels seront données à l'anneau.

Le PIN DO signifie Data Out. Ce PIN permet de relier des anneaux entre eux. Nous n'utiliserons qu'un seul anneau, ce PIN nous est donc inutile.

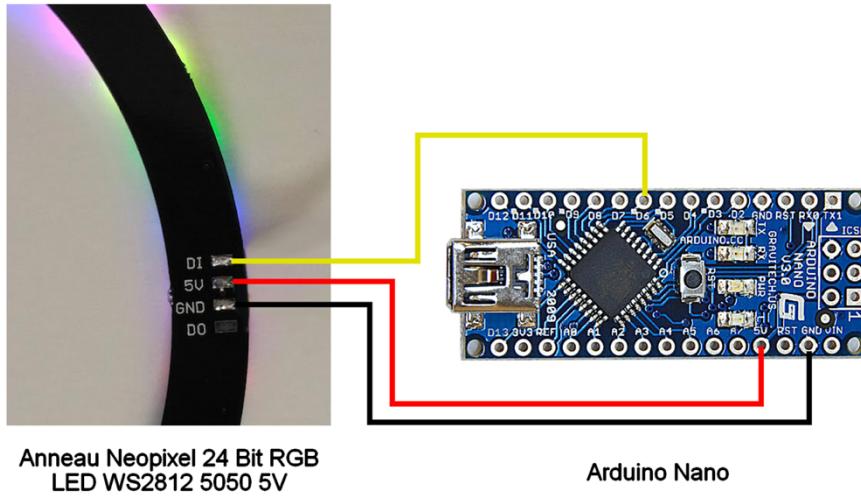


schéma IV.2

Source :  
<https://www.magicmanu.com/arduino/2018/02/horloge-a-led-assassins-creed/>

Sur le schéma IV.2, on peut voir comment peut être branché l'anneau. Ici, c'est un anneau d'une taille différente de celle que nous avons choisie mais les branchements sont les mêmes.

#### Quelques fonctions utiles de la bibliothèque <Adafruit\_Neopixel.h> :

Les Neopixels peuvent changer de couleur et d'intensité à volonté, ce qui est un point important pour notre lampe. Pour changer les couleurs, il nous faudra utiliser le module <Adafruit\_Neopixel.h>. C'est une bibliothèque regroupant les fonctions permettant de faire fonctionner les Neopixels.

Adafruit\_Neopixel ring(paramètre 1, paramètre 2, paramètre 3) : permet de configurer l'anneau. Ici « ring » indique de nous initialisons un anneau de Neopixels et pas une bande (pour une bande, on met « string » à la place de « ring »).

Paramètre 1 : nombre de neopixels que possède notre anneau.

Paramètre 2 : PIN où est branché le DI.

Paramètre 3 : notre type de pixel.

ring.begin() : permet d'initialiser l'anneau.

ring.setBrightness(X) : permet d'initialiser l'anneau à la luminosité X, X allant de 0 à 255.

ring.setPixelColor(W,X,Y,Z) : permet d'initialiser les led du neopixel W+1 : X pour la led rouge, Y pour la LED verte et Z pour la LED Z (X,Y,Z compris entre 0 et 255).

ring.fill(couleur,début,fin) : permet d'appliquer la couleur choisie sur les neopixels début+1 jusqu'à fin+1.

ring.show() : affiche les modifications.

## 2. Le module LED RVB

Il est également possible d'utiliser le module LED RVB pour colorer notre fleur de magnifiques couleurs. Ces modules sont constitués de 3 LEDs, regroupées sous une coque en plastique semi-transparente. Chaque LED est adressable individuellement par une broche différente.

4 broches différentes sont présentes sur ce module : la broche VCC, et les broches G, B, R (respectivement pour le vert, le bleu et le rouge).

### Principe de fonctionnement :

Ce module fonctionne avec la carte Arduino UNO que nous utilisons lors des travaux pratiques. Le branchement est simple et est décrit grâce au schéma IV.3 :

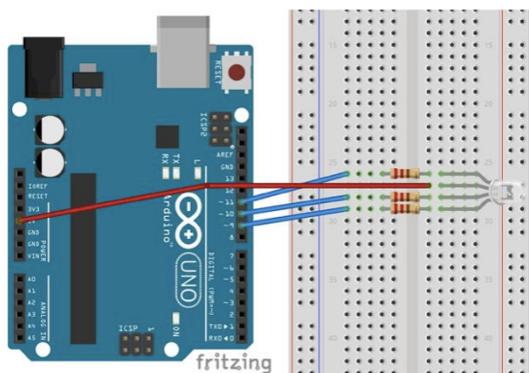


schéma IV.3

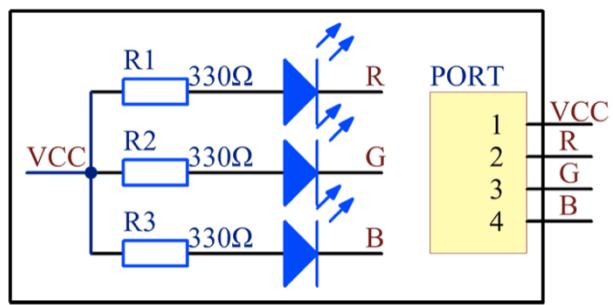


schéma IV.4

Chaque fils ayant pour but de relier la carte Arduino aux différentes LED composant le module, doit passer par une résistance de 330 Ohm. Cela est nécessaire si on ne veut pas griller les LED à l'intérieur du module. Ces fils sont reliés aux sorties de la carte compatibles avec le PWM. Ici, il est très important de savoir utiliser le PWM. En effet, chaque LED va être commandée à l'aide de ce type de signal. Ce signal va commander l'intensité de chacune des couleurs. L'association de ces trois LED à intensité variable va nous permettre de réaliser toutes les couleurs de l'arc-en-ciel.

## C. Conclusion :

L'anneau de néopixels a de nombreux avantages :

- il est très petit
- il possède 7 néopixels, ce qui pourra nous permettre de créer des effets visuels
- il ne se commande qu'à l'aide d'un seul fil, le branchement est donc peu encombrant.

Le module LED RVB ne possède pas tous ces avantages. C'est pour cela que nous allons de préférence nous tourner vers l'anneau de néopixels pour réaliser notre fleur.

## V. Les pétales de notre lampe

### A. Objectifs et attentes

Nous voulons créer une lampe d'ambiance. Pour cela, nous voulons utiliser des lumières permettant de nous plonger dans une certaine atmosphère. Pour que la lumière se perçoive correctement, nous voulons utiliser des pétales assez transparents. Ainsi, la lumière pourra s'y refléter.

Il nous faudra trouver le matériau :

- permettant de refléter au mieux la lumière.
- assez flou pour cacher les imperfections et donner un résultat un peu « fumé » aux pétales.
- assez épais afin d'obtenir une fleur solide.

### B. Les différentes options

#### 1. Le plexiglass

Nous avons d'abord pensé à utiliser du plexiglass. Nous comptons plus particulièrement choisir le plexiglass flou afin que les petits détails au centre de notre lampe ne soient pas visibles. Nous comptons choisir un plexiglass d'environ 3mm, ce qui nous semble assez épais pour ne pas avoir une fleur trop fragile.

Il faudra découper des morceaux de plexiglass de la taille de nos pétales puis, à l'aide d'une chauffeuse thermique, nous allons leur donner la forme souhaitée. Afin de pouvoir avoir un rendu homogène, il nous faudra sûrement créer un moule.

#### 2. L'impression 3D

Nous envisageons également l'impression 3D afin de fabriquer nos pétales. Avec cette technique, tous les pétales seraient totalement identiques.

L'impression 3D nous permet d'obtenir plusieurs couleurs de pétales (nous expérimenterons le blanc et le transparent). D'après le projet de la Ever Blooming Mechanical Tulip, cela nous permettra d'avoir un rendu semi-transparent, ce qui est parfait pour notre fleur.

De plus, nous pourrons y inclure une partie qui pourra nous servir à l'ouverture et à la fermeture de la fleur.

L'impression 3D nécessite l'apprentissage d'un de ces logiciels :

- Freecad
- Inkscape
- OpenSCAD

Tous ces logiciels sont équivalents. Nous avons décidé de choisir Freecad car il est utilisable sur Mac et que de nombreux tutoriels sur internet vont nous permettre d'apprendre les bases facilement.

### C. Conclusion

Le plexiglass n'est pas la meilleure idée selon nous. En effet, il est vendu rigide ce qui nous obligerait à le travailler avec de la chaleur. Cette technique ne nous garantit pas un résultat parfait. L'impression 3D est, pour l'instant, l'option qui nous convient le mieux.

## VI. Le principe du mécanisme d'ouverture/ fermeture

### A. Objectifs et attentes

Notre idée est de faire une fleur capable de s'ouvrir et de se fermer. C'est pour cela qu'un mécanisme muni d'un moteur est nécessaire. Il existe plusieurs possibilités de mécanisme d'ouverture/ fermeture.

Trouver le mécanisme d'ouverture/fermeture :

- le moins visible
- le plus efficace

### B. Les différentes options

#### 1. Les fils

Notre première idée était de placer des fils sur chaque pétales et de les embobiner autour d'un axe central à l'aide d'un moteur, afin de modifier leur distance du centre. Cela aurait permis aux pétales de changer d'inclinaison. Pour cela, il nous faudrait placer les fils assez bas au niveau des pétales afin de ne pas les rendre trop visibles. Un perçage sur chaque pétales est alors nécessaire afin d'y faire passer le fil. Une tige centrale permettrait de relier tous les fils entre eux, en un seul fil plus épais, relié au moteur.

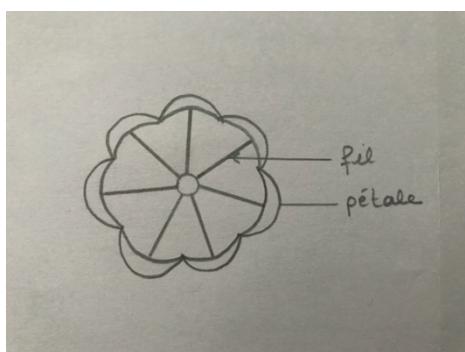


schéma VI.1

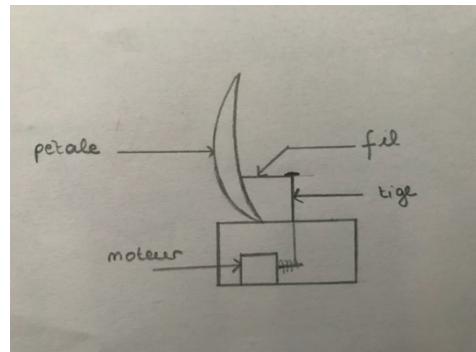


schéma VI.2

## 2. La tige rétractable

L'astuce utilisée dans la Ever Blooming Mechanical Tulip est très intéressante. Elle consiste à changer la hauteur de la tige. La tige appuierait sur les pétales et leur permettrait de s'ouvrir ou de se fermer. C'est pour cela qu'inclure une partie sur laquelle des petites tiges peuvent appuyer au niveau des pétales est intéressant.

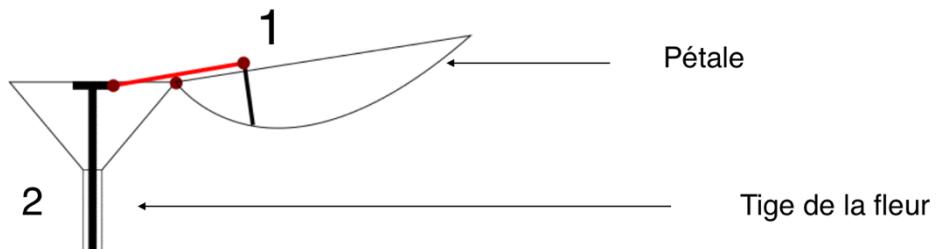


schéma VI.3

Sur le schéma du site de la Ever Blooming Mechanical Tulip (schéma I.1), nous pouvons mieux comprendre le mécanisme :

- 1 : Partie accrochée au pétale (troué) qui permet à la tige intermédiaire (rouge) de faire bouger le pétale.
- 2 : Tige de la fleur qui peut être montée et descendue à l'aide d'un moteur, caché dans le socle de notre lampe. Cela va permettre de tirer ou pousser la tige intermédiaire et ainsi, ouvrir ou fermer notre fleur.

## C. Conclusion

Le mécanisme des fils n'est pas le meilleur pour de nombreuses raisons :

- les fils peuvent s'abîmer au cours du temps
- les fils peuvent s'emmêler et casser la fleur
- cela nous oblige à percer les pétales afin d'y placer les fils, ou de les coller directement dessus. Cela risque d'altérer le rendu final de notre projet.

C'est pour cela que la technique de la tige rétractable est celle que nous envisagerons en premier.

## VII. La connexion Bluetooth

### A. Objectifs et attentes

L'un des objectifs de notre projet est de permettre à l'utilisateur de choisir son ambiance. Pour cela, nous souhaitons mettre à disposition une application sur le téléphone qui permet de changer la couleur de la fleur. Nous avons donc décidé d'utiliser une connexion Bluetooth pour connecter le téléphone avec la fleur.

## B. Les différentes options

### 1. Le module RF 433 mHz

Pour cela, nous nous sommes tout d'abord renseignées sur le RF 433mHz qui est un module utilisant les radiofréquences.

Recepteur

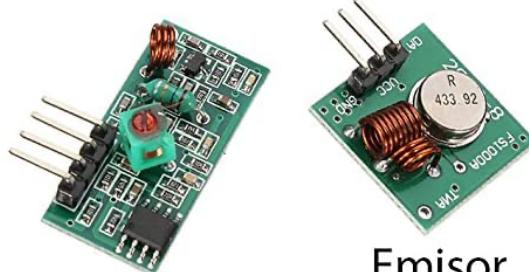


schéma VII.1

Emisor

<https://images-na.ssl-images-amazon.com/images/I/51cyl4FESnL.AC.jpg>

Pour utiliser la radiofréquence, il nous faut deux modules : un émetteur et un récepteur. Or, nous voulons utiliser notre smartphone afin d'envoyer des informations au récepteur. L'émetteur nous serait donc inutile. Les smartphones ne peuvent initialement pas communiquer en radiofréquence. Il nous faudrait trouver un autre moyen de permettre à notre téléphone de communiquer avec notre fleur. Toutes nos recherches nous ont menées vers un module beaucoup plus simple d'utilisation, qui utilise le Bluetooth : le HC-06.

Le Bluetooth un mode de communication sans fil visant à connecter deux appareils entre eux. La portée de cette connexion est très courte. Le Bluetooth fonctionne sur un schéma de maître/esclave, c'est-à-dire qu'un appareil « maître » contrôle un autre appareil « esclave ».

### 2. Le module HC-06

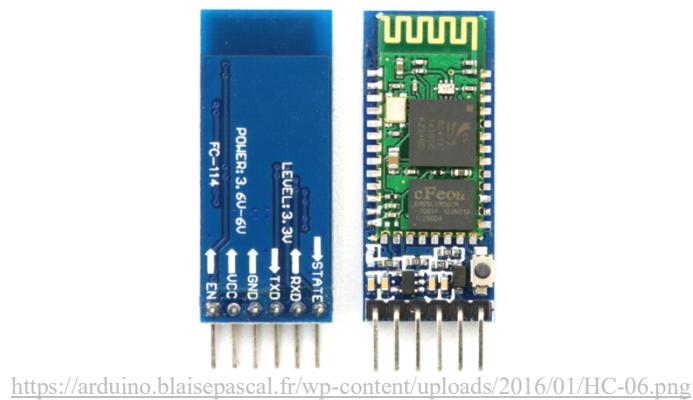


schéma VII.2

<https://arduino.blaisepascal.fr/wp-content/uploads/2016/01/HC-06.png>

Le module HC-06 va nous permettre de contrôler notre carte Arduino grâce à une application. Il possède 6 branches. 4 d'entre-elles vont nous être utiles :

- Vcc : entre 3,6V et 6V
- Gnd : la masse
- RX : pour la réception des données
- TX : pour la transmission des données

Ce module est un module esclave (il est contrôlé par un autre module Bluetooth), contrairement au module HC-05, qui peut contrôler d'autres modules lorsqu'il est sur le mode maître (ce qui ne nous intéresse pas ici).

### Quelques commandes pour la configuration du module :

- AT : permet de vérifier si la connexion avec le module a été effectuée.
- AT+PINxxx : permet de modifier le mot de passe de connexion au module (le mot de passe par défaut est 1234).
- AT+NAMExxx : permet de changer le nom du module.
- AT+BAUDx : permet de changer la vitesse avec laquelle le module communique avec la carte Arduino.

### Branchements du module HC-05 :

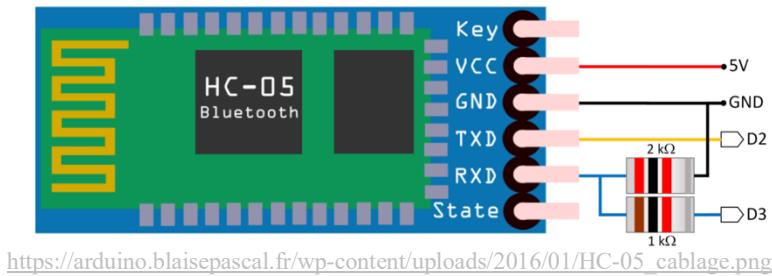


schéma VII.3

[https://arduino.blaisepascal.fr/wp-content/uploads/2016/01/HC-05\\_cablage.png](https://arduino.blaisepascal.fr/wp-content/uploads/2016/01/HC-05_cablage.png)

Voici un schéma (schéma VII.3) de branchement du module HC-05. Le schéma est le même pour le module HC-06. On peut y voir un diviseur de tension entre la branche RX et la carte Arduino. Ce diviseur permet de descendre la tension à 3,3V à l'entrée de la branche RX. Ce type de branchement est très conseillé afin de ne pas abîmer les composants.

Pour communiquer des informations de la carte Arduino au module, on utilise la librairie « SoftwareSerial ». Pour définir les I/O RX et TX on utilise le code suivant :

SoftwareSerial XXX(RX, TX) où XXX est le nom donné à la communication série entre la carte et le module.

### Les fonctions de cette librairie sont :

- begin() : permet de définir la vitesse de communication.
- read() : permet de lire un caractère de la broche de réception.
- print() : permet d'afficher les données émises par le module HC-06.
- write() : permet d'envoyer des données, octet par octet.
- available() : permet de tester si des données sont disponibles dans le buffer.

### Connexion entre le smartphone avec le module : l'application

Pour changer l'ambiance de notre fleur, nous allons passer par l'intermédiaire d'une application smartphone. Pour cela, nous pouvons utiliser plusieurs logiciels disponibles sur Android comme MIT App Inventor ou bien Bluetooth Electronics.

### Le fonctionnement de la communication :

Le smartphone envoie des informations au module grâce à l'application. Les données envoyées sont stockées dans un buffer que le module lit constamment (pour cela on utilise un code avec la fonction available() placée dans un if : si une information est reçue, on effectue des instructions).

### 3. Un dérivé, le module HM-10

C'est un module produit par Texas Instrument Bluetooth Low Energy. Ce module est très similaire au HC-06. Il possède cependant une vitesse de transmission d'informations beaucoup plus grande que le HC-06. La vitesse du HC-06 est de 2.1 Mbps (Mégabits par seconde) contre 24 Mbps pour le HM-10. De plus, le module HM-10 consomme pas beaucoup d'énergie et a une portée de 100m.

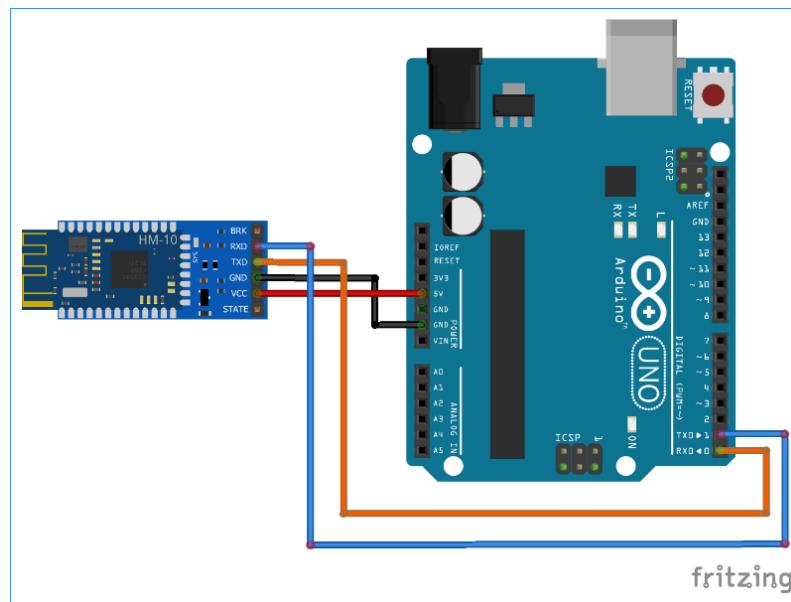


schéma VII.4

[https://circuitdigest.com/sites/default/files/circuitdiagram\\_mic/Circuit-Diagram-for-HM-10-BLE-Module-with-Arduino-to-Control-an-LED-using-Android-App\\_0.png](https://circuitdigest.com/sites/default/files/circuitdiagram_mic/Circuit-Diagram-for-HM-10-BLE-Module-with-Arduino-to-Control-an-LED-using-Android-App_0.png)

Il possède les mêmes PIN que les modules HC-05 et HC-06 : les PINs : RX, TX, Vcc et GND. Ce module est un peu plus cher que les précédents.

D'après nos recherches, ce module peut être contrôlé avec la librairie « SoftwareSerial » soit la même librairie que pour les modules HC-05 et HC-06.

### C. Conclusion :

Le module HC-06 est un module assez simple d'utilisation qui, combiné à une application, va permettre à l'utilisateur de gérer lui-même les ambiances de la Mood Lamp. Le module HM-10 est un module très utilisé et plus performant que le HC-06. Cependant, son coût est plus élevé et les avantages de ce module sont secondaires pour notre projet.

## VIII. Conclusion générale

Pour réaliser notre Mood Lamp, nous comptons utiliser les éléments suivants :

- Un servomoteur classique pour actionner notre lampe.
- Un anneau de neopixels de 7 LED afin de pouvoir allumer notre lampe en créant différentes ambiances.
- Une imprimante 3D pour imprimer nos pétales.
- Un module HC-06 et l'application MIT App Inventor (sur Android).

Voici le résultat que nous souhaitons obtenir (sans la feuille décorative) :



<https://floower.io/we-bloom-all-around-the-world/>

photo VIII.1