

Rapport de Mini-projet ARDUINO

Établissement : EMSI Casablanca

Site : Maarif

Classe : 2AP

Groupe : G31

Titre :

SUNNY UN SYSTEME SUIVEUR DE LUMIERE



Réalisé par :

**- Ikram MOUSLIH
- Sohaib JALIL**

Encadré par:

- Mme. CHAKIR Asmae

Année universitaire: 2020/2021

Dédicace

A nos chers parents, qui nous ont aidé et soutenu à réussir ce projet que Dieu les gardes en bonne et parfaite santé.

Nos sœurs et frères qui nous éblouissent par leurs soutient et leurs encouragements.

Tous nos chers (es) amis (es), à tous ceux que nous aimons et à toutes les personnes qui nous ont prodigués des encouragements pour la réalisation de ce projet.

Remerciement

Avant de commencer la présentation de ce travail, nous profitons l'occasion pour remercier toutes les personnes qui ont contribués de près ou de loin à la réalisation de ce projet.

Au terme de ce travail, nous tenons à exprimer notre profonde gratitude et nos sincères remerciements à notre encadrante Mme. CHAKIR Asmae pour tout le temps qu'elle nous a consacré, sa directive précieuse, et pour la qualité de son suivi durant toute la période de travail sur ce projet.

Nos profonds remerciements vont aussi à toute l'École Marocain des Sciences de l'Ingénieur Maarif et à toutes les personnes qui ont accepté de répondre à nos questions durant ce semestre et qui nous ont aidés par leurs paroles, leurs écrits, leurs conseils et leurs critiques qui ont guidé nos réflexions.

Enfin, nous adressons nos plus sincères remerciements à nos familles, tous nos proches et amis qui nous ont accompagnés, soutenus, aidés et encouragé tout au long de la réalisation de ce projet.

Table des matières

Dédicace.....	2
<i>Remerciement</i>	3
Liste des figures	6
Introduction générale	7
Chapitre I : Contexte et étude générale du projet	9
1. Problématique du projet :	9
2. Le système de poursuite solaire :	10
3. Les applications :	10
4. Les types de suiveurs de lumière :	11
4.1. Suiveurs de lumière passifs :	11
4.2. Suiveurs de lumière actifs :	12
5. Les Avantages et Les inconvénients des suiveurs de lumière :	13
5.1. Les Avantages :	13
5.2. Les inconvénients :	13
6. Conclusion :	13
Chapitre II : Etude et solution matérielle	14
Partie A : Description des composants du système.....	15
1. Introduction sur Arduino :	15
2. Servo moteur :	17
3. Photorésistance LDR :	18
4. LABDEC - Platine d'expérimentation :	20
5. Resistance :	20
6. Interrupteur à glissière – Slide switch :	21
7. LED - Light-emitting diode :	23
8. Bipeur – Buzzer piézo-électrique :	24
Partie B : Réalisation du suiveur de lumière.....	25
1. Fonctionnement de système :	25
2. Schéma du Circuit Complet :	25
3. Câblage du système étape par étape :	26
4. Circuit Complet sur Tinkercad :	30

Chapitre III : Programmation et solution logicielle	31
1. Logiciel Arduino :	31
2. Arduino Setup :	32
3. Les fonctions Arduino utilisées :	33
4. Code complet :	34
Conclusion :	39
Chapitre IV : Bilan de projet et simulation.....	40
Conclusion générale	41
Bibliographie.....	42

Liste des figures

Figure 1 : Modèle d'un héliostat dans ses premières applications	10
Figure 2 : Types des suiveurs solaires	11
Figure 3 : Suiveur solaire mono-axe	12
Figure 4 : Suiveur solaire double axe.....	12
Figure 5 : Carte Arduino UNO	15
Figure 6 : Servo moteur.....	17
Figure 7 : Structure d'un Servo moteur	17
Figure 8 : Branchement d'un Servo Moteur	18
Figure 9 : LDR et son symbole.....	19
Figure 10 : Branchement du LDR.....	19
Figure 11 : LABDEC.....	20
Figure 12 : Resistance et son symbole	20
Figure 13 : Interrupteur à glissière	21
Figure 14 : Symbole d'Interrupteur à glissière	22
Figure 15 : Branchement d'un interrupteur à glissière.....	23
Figure 16 : LEDs	23
Figure 17 : LED et son symbole.....	23
Figure 18 : Branchement d'une LED	23
Figure 19 : Buzzer.....	24
Figure 20 : Branchement d'un Buzzer	24
Figure 21 : Schéma du circuit final	25
Figure 22 : Circuit final sur Tinkercad.....	30
Figure 23 : Présentation de l'interface initiale du logiciel.....	31
Figure 24 : Photo du projet final.....	40

Introduction générale

La plupart d'énergie consommée actuellement vient de l'utilisation des combustibles fossiles comme le pétrole, le charbon, le gaz naturel ou l'énergie nucléaire. Ces ressources deviennent de plus en plus rares, pendant que les demandes énergétiques du monde s'élèvent toujours. Il est estimé que les réserves mondiales seront épuisées vers 2030 si la consommation n'est pas modifiée, et au maximum vers 2100 si des efforts sont produits sur la production et la consommation. Etant donné que cette forme d'énergie couvre une grosse partie de la production énergétique actuelle, il s'avère nécessaire de trouver une autre solution pour prendre le relais, la contrainte imposée et d'utiliser une source d'énergie économique et peu polluante car la protection de l'environnement est devenue un point important. À ce sujet, les énergies renouvelables, comme l'énergie solaire (photovoltaïque ou thermique), éolienne ou hydraulique, ... apparaissent comme des énergies inépuisables et facilement exploitables.

Les systèmes d'énergie solaire tirent une énergie propre et pure du soleil. Une grande partie de leurs fonctionnements se représente dans leur rotation qui se fait manuellement, ce qui exige que quelqu'un ajuste physiquement les panneaux à différents moments de la journée pour suivre le soleil. Ce n'est pas toujours pratique, car ça nécessite une personne pour surveiller en permanence le soleil et changer la position du système de panneaux solaires en conséquence. Et cela affecte la quantité de lumière atteignant le système. L'installation des panneaux solaires peut être un gros investissement, c'est pourquoi il devrait y avoir un moyen d'en profiter au maximum.

Afin de collecter le maximum d'énergie, on utilise des dispositifs appelés des systèmes de poursuite solaires ou suiveurs de lumière ou « trackers solaires ».

Dans ce cadre nous nous sommes chargés de faire le suivi d'un projet qui concerne un mini panneau solaire équipés d'un mécanisme suiveur de la lumière qui peut collecter plus de lumière, en d'autres termes, générer plus d'énergie pendant des périodes plus longues.

Le rapport du projet comportera les chapitres suivants :

Le premier chapitre sera consacré à la présentation de la problématique de notre sujet et un contexte général sur les systèmes de poursuite solaire, ensuite nous avons présenté leurs applications et les différents types des suiveurs en citant leurs avantages et leurs inconvénients.

Le deuxième chapitre, sera dédié à une étude approfondie de tous matériels et mécaniques appropriés à la mise en œuvre du projet tout en citant leurs fonctionnalités et leurs utilités.

Le troisième chapitre sera consacré à la réalisation de la partie code et programmation tout en expliquant le fonctionnement de cette dernière.

On termine avec une conclusion regroupant les différents aspects étudiés.

Chapitre I : Contexte et étude générale du projet

Le bon déroulement d'un projet nécessite la bonne compréhension de son environnement du travail. Vu que l'objectif principale de ce rapport se base sur la mise en œuvre d'un système suiveur de lumière qui peut être utiliser dans plusieurs domaines. Dans notre cas, ce système est complémentaire aux panneaux solaires qui doivent être dirigés vers le soleil toute la journée.

Ce chapitre se portera tout d'abord sur une brève description sur la problématique et les contraintes de ce travail, ensuite un aperçu sur les fonctionnalités et aussi les applications d'un tel système.

1. Problématique du projet :

Le rendement maximal des panneaux est atteint lorsque les panneaux sont perpendiculaires aux rayons du soleil et les angles de réception de la lumière au niveau des panneaux dans lesquels l'intensité du rayonnement solaire reçu du soleil sur la terre est affectée de manière significative par le mouvement journalier du soleil ainsi que le mouvement saisonnier de la terre. Tout simplement, les panneaux solaires doivent faire face au soleil aussi directement que possible pour un bon rendement.

Au cours des dernières années, des systèmes solaires fixes ou manuelles étaient utilisés, mais ces derniers ont des désavantages tels que :

- ✓ Un rendement minimal.
- ✓ Perte des ressources.
- ✓ Trop de travail et d'efforts.

Devant ces considérations, l'objectif principal de notre projet est d'optimiser le rendement de puissance des panneaux en s'assurant qu'ils sont inclinés vers le soleil à tout moment pendant la journée en utilisant les suiveurs de lumière.

Un suiveur de lumière est un système qui utilise le principe de l'héliostat. C'est une structure portante qui permet d'orienter des capteurs constamment vers la lumière. Ceci permet d'augmenter la production de l'énergie d'une manière substantielle. Suivre la lumière, ou plus précisément, le soleil d'une manière rigoureuse peut se faire normalement sur deux axes (en azimut et en altitude). Toutefois, il existe des suiveurs à un seul axe (en azimut).

2. Le système de poursuite solaire :

Les premières utilisations des suiveurs de lumière remontent vers les années 1750 où le principe repose sur l'utilisation de l'héliostat. La rotation du miroir de l'héliostat est faite par un mécanisme d'horlogerie. La plus ancienne mention du système se trouve dans des publications qui datent de 1742 du physicien Willem Jacobs Gravesande.

Un champ d'héliostats repère le mouvement du soleil en convergeant les rayons lumineux vers un même point, concentrant ainsi la lumière incidente en employant des miroirs de réflexion, durant toute la journée. Ce principe se rencontre dans diverses installations utilisant l'énergie solaire, surtout les centrales solaires à concentration appelées champs d'héliostats.

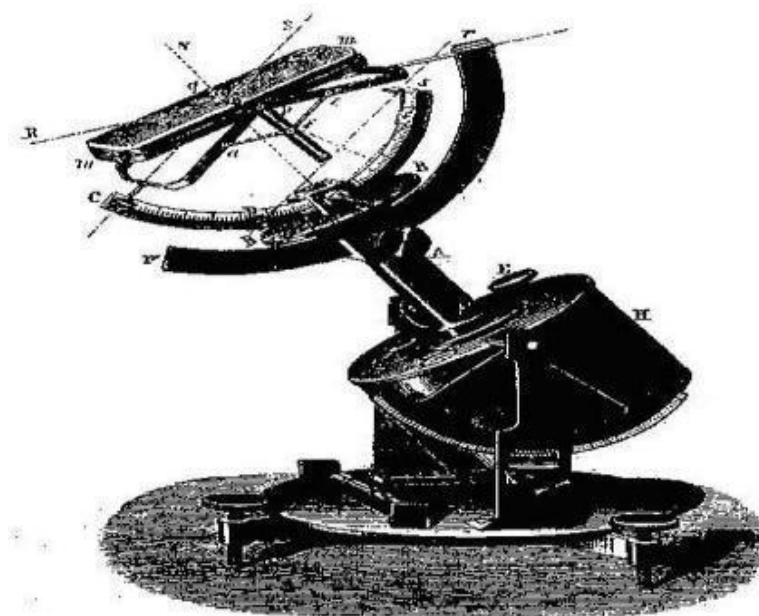


Figure 1 : Modèle d'un héliostat dans ses premières applications

3. Les applications :

Ce travail présente la conception et l'exécution d'un algorithme de commande pour une structure mécanique à prix réduit qui peut soutenir les dispositifs photovoltaïques et qui agit en tant que traqueur du soleil.

Nous pouvons citer 2 utilisations principaux pour les suiveurs de lumière :

- ✓ Dans les télescopes héliographiques utilisés pour apercevoir des objets célestes ponctuels difficilement perceptibles ou invisibles à l'œil nu.
- ✓ Dans l'installation de production d'énergie solaire en motorisant les panneaux solaires pour les orienter afin d'en augmenter la productivité.

4. Les types de suiveurs de lumière :

On distingue principalement deux grandes familles de suiveurs de lumière : les passifs et les actifs qui comportent les suiveurs mono-axe et double axe.

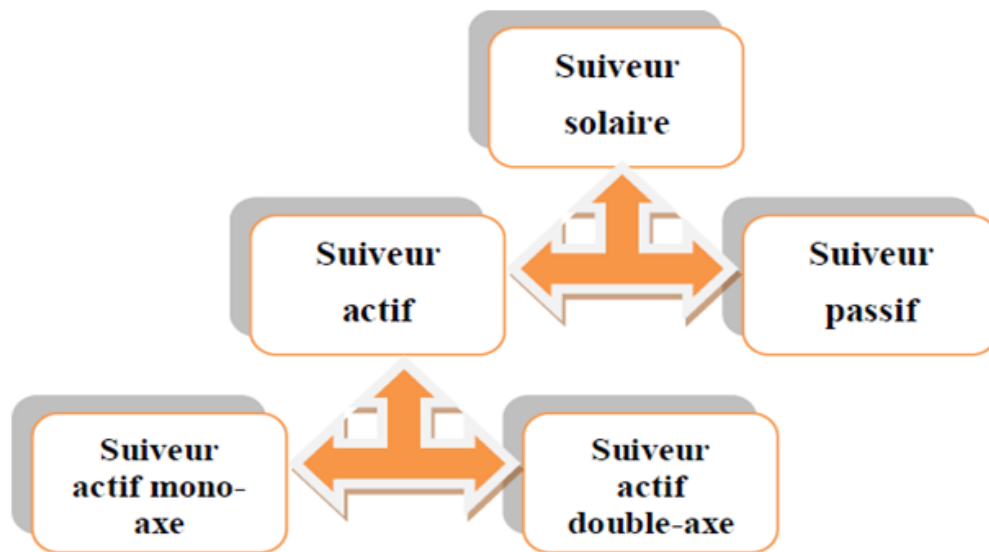
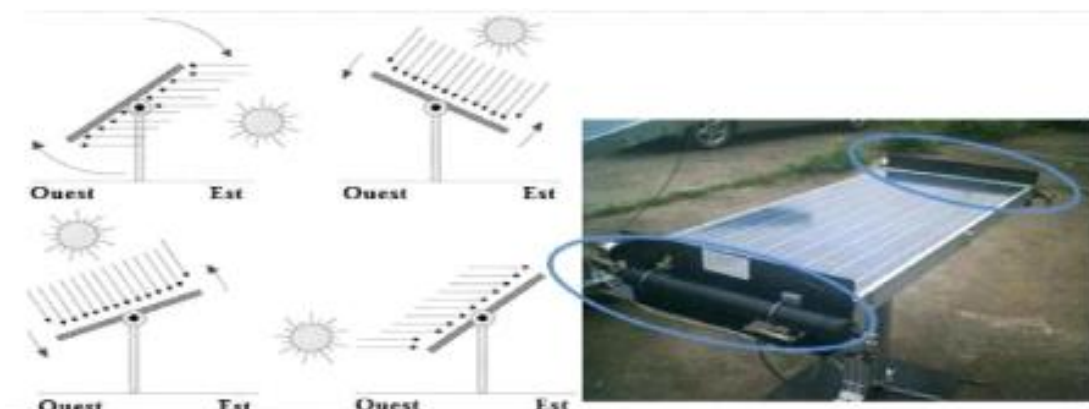


Figure 2 : Types des suiveurs solaires

4.1. Suiveurs de lumière passifs :

Un suiveur de lumière passif est caractérisé dans son architecture par deux tubes en cuivre montés sur les côtés Est et Ouest du capteur de lumière. Le tube de matière cuivre est rempli de fluides chimiques qui se vaporisent à basse température. Et donc, lorsque l'exposition au rayonnement solaire augmente, la température d'un côté du panneau augmente et le composé dans le tube en cuivre se vaporise. La partie gazeuse du composé occupe un plus grand volume interne, et sa partie liquide est décalée vers le côté ombragé. Cette opération de transfert de masse ajuste l'équilibre du capteur de lumière en le faisant tourner vers la source des rayons lumineux. Ce type de suiveur ne nécessite aucune énergie pour le repositionnement du panneau.



4.2. Suiveurs de lumière actifs :

Les suiveurs de lumière ou solaire actifs utilisent le principe de la détection de lumière, suivant la trajectoire solaire en cherchant à optimiser au maximum l'angle d'incidence du rayonnement solaire sur leur surface. Il y existe deux types: les suiveurs mono-axe et double axe. L'avantage de ces derniers par rapports aux suiveurs passifs c'est qu'ils représentent une précision de suivi et ne nécessitent aucune intervention manuelle pour les ajuster.

4.2.1. Suiveur de lumière mono-axe :

Le suiveur de lumière tourne autour d'un axe simple. Le trajet suivi par cet axe est souvent en azimut, c'est-à-dire d'Est en Ouest au fil de la journée. L'angle d'inclinaison du panneau est fixe, il est placé selon l'angle meilleur à recevoir le maximum de rayons du soleil durant toute l'année. Il est placé selon la latitude du lieu où est disposé le capteur. Ainsi, l'angle d'inclinaison du panneau reste constant mais son azimut varie au cours de la journée. Le système de poursuite mono-axial est la solution la plus simple et la plus utilisée.

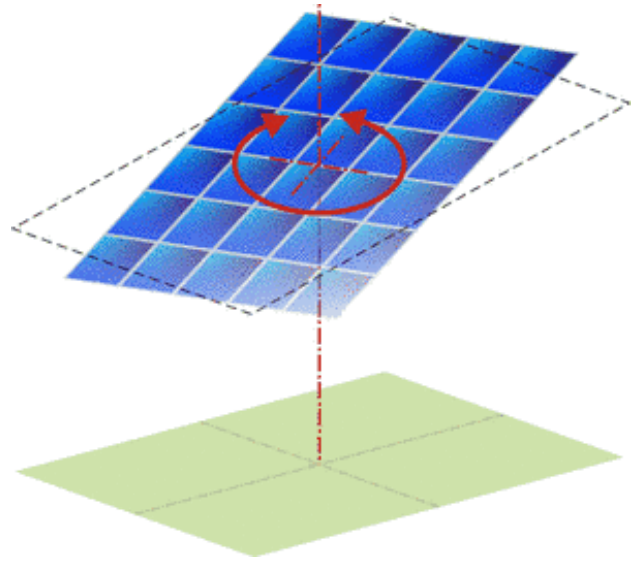


Figure 3 : Suiveur solaire mono-axe

4.2.2. Suiveur de lumière double axe :

Contrairement au précédent, ce suiveur de lumière possède deux axes ce qui lui permet de suivre les mouvements en azimut et en inclinaison au cours de la journée. Ainsi, le capteur reste constamment orienté vers le soleil. Ce type de suiveur est très imposant et fournit de meilleures performances par rapport au suiveur mono-axe, mais ce dernier consomme plus d'énergie.

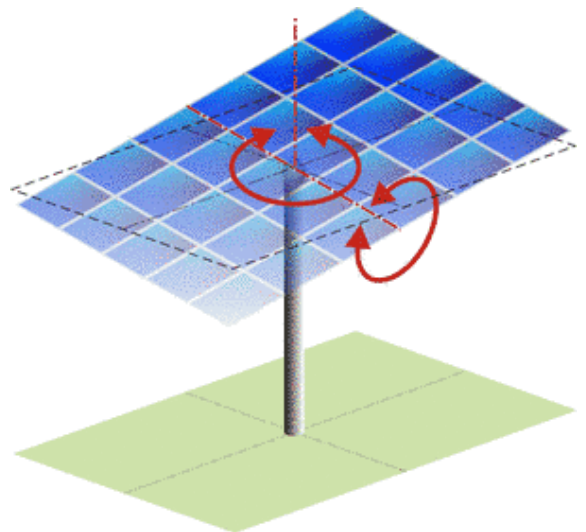


Figure 4 : Suiveur solaire double axe

5. Les Avantages et Les inconvénients des suiveurs de lumière :

5.1. Les Avantages :

- ✓ Les systèmes de suivi de lumière sont utilisés pour orienter les capteurs continuellement vers le soleil et peuvent aider à minimiser l'investissement en augmentant le rendement d'énergie.
- ✓ Pour le photovoltaïque, le rendement des panneaux est augmenté de 30 à 40%.
- ✓ La surface du parc solaire est réduite tout en gardant le même potentiel.
- ✓ Le temps de retour sur investissement est réduit.
- ✓ Exploitation optimale du rayonnement solaire même en présence des nuages.
- ✓ Gain intéressant, notamment aux heures de la journée les moins ensoleillées.

5.2. Les inconvénients :

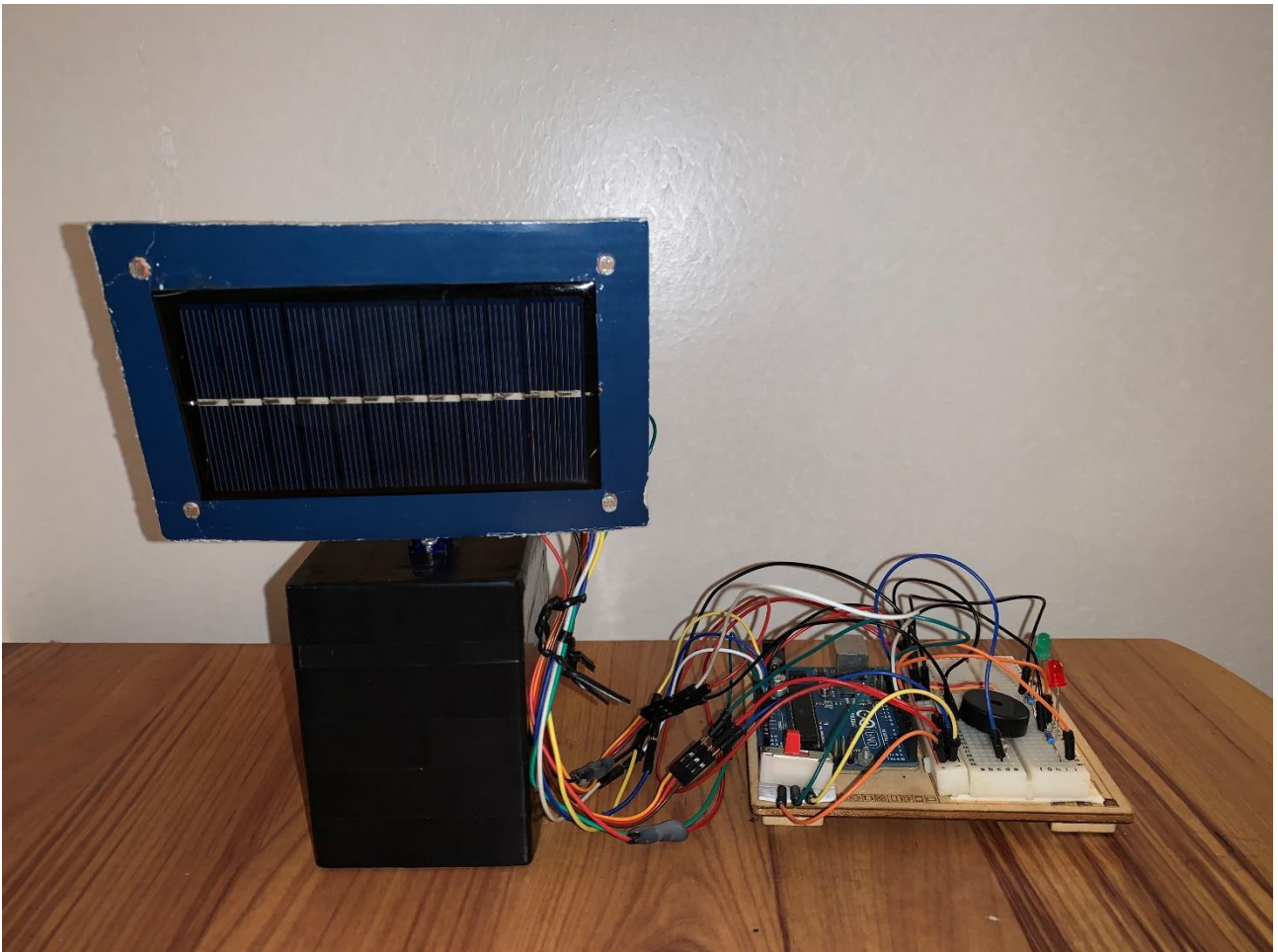
- ✓ Le coût d'investissement initial (surtout pour les trackers à 2 axes) est élevé.
- ✓ La maintenance des trackers, car les systèmes mécaniques atteignent rarement une durée de vie de 20 ans.
- ✓ Pour des raisons de sécurité, les panneaux solaires doivent être parallèles au sol lors de vents forts.

6. Conclusion :

Dans ce chapitre nous avons démontré la raison derrière l'invention de ces trackers et la manière à définir la meilleure façon avec laquelle sera effectué le suivi automatique. Nous avons aussi présenté un état de l'art sur les méthodes de suivi. Un intérêt particulier a été porté à la technique de suivi active reposant sur le principe du repérage via des capteurs de lumière. C'est cette technique qu'on a adoptée et qui va être étudiée en vue d'une réalisation matérielle. Dans le chapitre qui suit, nous allons parler sur les dispositifs utilisés afin de réaliser notre suiveur.

Chapitre II : Etude et solution matérielle

Dans cette partie nous présentons la procédure de réalisation de notre suiveur de lumière à base de la carte Arduino. Le dispositif électronique comprend deux servomoteurs, quatre photorésistances LDR, et un mini panneau solaire. Le système d'orientation est basé sur la programmation en logiciel Arduino du déplacement du panneau solaire qui est installé sur deux barres métalliques équipés d'un servo moteur chacune. Le premier moteur contrôle la rotation horizontale alors que le deuxième contrôle la rotation verticale. Le dispositif tourne selon le niveau de lumière calculé par les photorésistances installées sur les 4 coins du mini panneau solaire d'où les rayons de la lumière soient perpendiculaires avec le panneau solaire.



Partie A : Description des composants du système

1. Introduction sur Arduino :

Arduino est une plate-forme de prototypage d'objets interactifs constituée d'une carte électronique et d'un environnement de programmation. Cet environnement matériel et logiciel permet à l'utilisateur de créer ses projets par l'expérimentation directe avec l'aide de nombreuses ressources disponibles en ligne. Arduino est un projet open source (les plans, les schémas, etc. sont accessibles et libres de droits) la grande communauté d'utilisateurs et de concepteurs permet à chacun de trouver les réponses à ces questions et apporte un travail énorme de documentation au projet. Les projets Arduino peuvent être autonomes, comme ils peuvent communiquer avec d'autres logiciels installés sur l'ordinateur tel que Flash, Processing ou MaxMPS, Matlab). Ces cartes sont faites à base d'une interface entrée/sortie simple et d'un environnement de développement proche du langage.

1.1. La constitution de la carte Arduino UNO :

L'Arduino UNO fournit un environnement de développement en s'appuyant sur des outils open source comme interface de programmation. L'injection du programme déjà converti par l'environnement sous forme d'un code « HEX » dans la mémoire du microcontrôleur se fait d'une façon très simple par la liaison USB. En outre, des bibliothèques de fonctions "clé en main" sont également fournies pour l'exploitation d'entrées-sorties. Cette carte est basée sur un microcontrôleur ATmega 328 et des composants complémentaires. La carte Arduino contient une mémoire morte de 1 kilo. Elle est dotée de 14 entrées/sorties digitales (dont 6 peuvent être utilisées en tant que sortie PWM), 6 entrées analogiques et un cristal a 16 MHz, une connexion USB et Possède un bouton de remise à zéro et une prise jack d'alimentation.

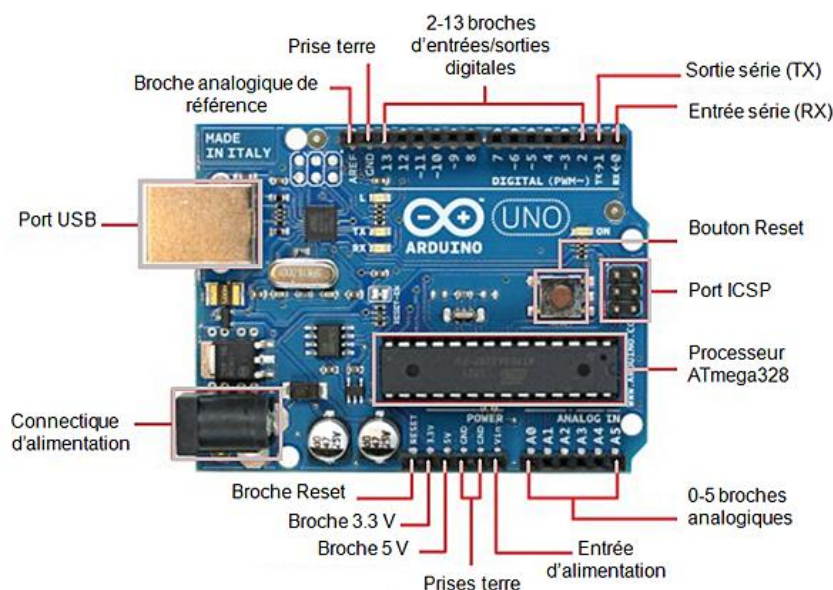


Figure 5 : Carte Arduino UNO

1.2. Pourquoi Arduino UNO :

Il y a de nombreuses cartes électroniques qui possèdent des plateformes basées sur des microcontrôleurs disponibles pour l'électronique programmée. Tous ces outils prennent en charge les détails compliqués de la programmation et les intègrent dans une présentation facile à utiliser. De la même façon, le système Arduino simplifie la façon de travailler avec les microcontrôleurs tout en offrant à personnes intéressées plusieurs avantages cités comme suit :

- ✓ **Multi plateforme** : le logiciel Arduino, écrit en JAVA, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux. La plupart des systèmes à microcontrôleurs sont limités à Windows.
- ✓ **Un environnement de programmation clair et simple** : l'environnement de programmation Arduino (le logiciel Arduino IDE) est facile à utiliser pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.
- ✓ **Logiciel Open Source et extensible** : le logiciel Arduino et le langage Arduino sont publiés sous licence open source, disponible pour être complété par des programmeurs expérimentés. Le logiciel de programmation des modules Arduino est une application JAVA multi plateformes (fonctionnant sur tout système d'exploitation), servant d'éditeur de code et de compilateur, et qui peut transférer le programme au travers de la liaison série (RS232, Bluetooth ou USB selon le module).
- ✓ **Matériel Open source et extensible** : les cartes Arduino sont basées sur les Microcontrôleurs ATMEL ATMEGA8, ATMEGA168, ATMEGA 328, les schémas des modules sont publiés sous une licence créative Commons, et les concepteurs des circuits expérimentés peuvent réaliser leur propre version des cartes Arduino, en les complétant et en les améliorant. Même les utilisateurs relativement inexpérimentés peuvent fabriquer la version sur plaque d'essai de la carte Arduino, dont le but est de comprendre comment elle fonctionne pour économiser le coût.
- ✓ **Le prix (réduits)** : les cartes Arduino sont relativement peu coûteuses comparativement aux autres plates-formes. La moins chère des versions du module Arduino peut être assemblée à la main, (les cartes Arduino préassemblées coûtent moins de 3500 Dinars).

2. Servo moteur :

Les servomoteurs sont des moteurs un peu spécifiques, qui peuvent tourner avec une liberté d'environ 180° et garder de manière relativement précise l'angle de rotation que l'on souhaite obtenir. On utilise des servomoteurs couramment en modélisme pour contrôler des systèmes mécaniques (gouverne d'avion, accélérateur de moteur thermique, etc.). Les servomoteurs sont aussi couramment utilisés en robotique pour faire des minirobots, des actionneurs ou des indicateurs rotatifs.



Figure 6 : Servo moteur

2.1. Le fonctionnement interne d'un servomoteur :

Un petit circuit électronique permet de contrôler un moteur à courant continu en fonction de la position d'un potentiomètre intégré au servomoteur. La sortie du moteur à courant continu est reliée mécaniquement à une série d'engrenages qui augmente la force (le couple) du servomoteur en réduisant la vitesse de rotation de celui-ci.

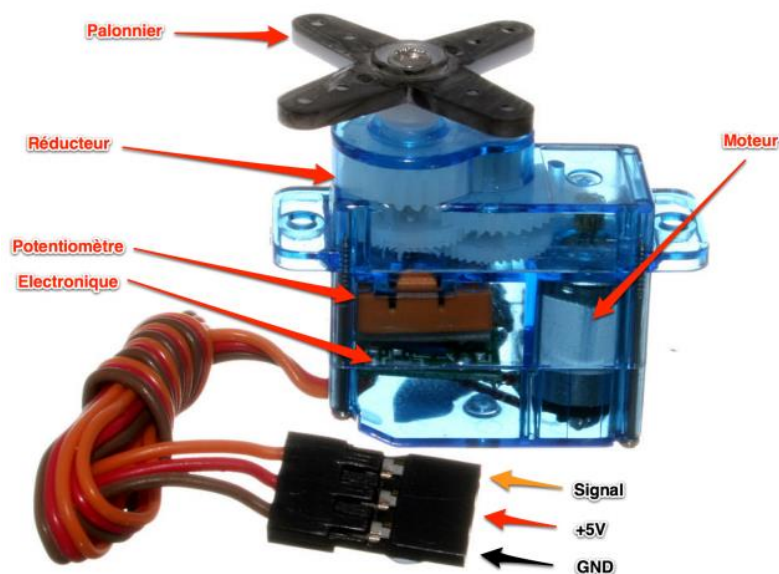


Figure 7 : Structure d'un Servo moteur

Quand le moteur tourne, les engrenages s'animent, le bras bouge et entraine avec lui le potentiomètre. Le circuit électronique ajuste continuellement la vitesse du moteur pour que le potentiomètre (et par extension le bras) reste toujours au même endroit. Il suffit de donner une consigne au servomoteur et le servomoteur fera de son mieux pour rester au plus près de cette consigne. Pour être précis, Cette consigne est transmise au moyen d'un signal (d'une impulsion) numérique. Pour que le servomoteur reste à une position donnée, il faut transmettre toutes les 20 millisecondes (soit à une fréquence de 50Hz) une impulsion d'une longueur comprise entre 1 et 2 millisecondes.

2.2. Branchement :

Le câblage est relativement simple : fil rouge du servomoteur sur la broche 5V de la carte Arduino, fil noir sur la broche GND et fil blanc (ou jaune en fonction des constructeurs) sur la broche D9 de la carte Arduino.

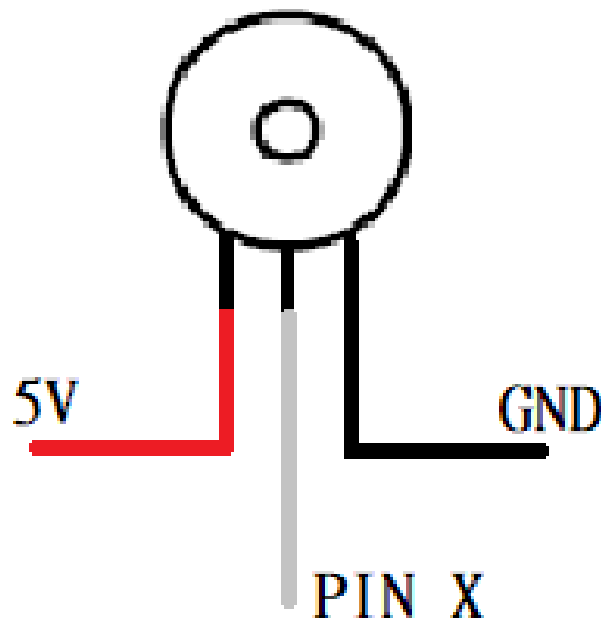


Figure 8 : Branchement d'un Servo Moteur

3. Photorésistance LDR :

3.1. Principe de fonctionnement :

Une photo résistance LDR (Light Dépendent Résistor ou résistance dépendant de la lumière) est un composant dont la valeur en Ohms dépend de la lumière à laquelle il est exposé. Les matériaux utilisés pour la construction sont généralement du Sulfure ou du Sélénure de Cadmium qui se comporte comme des semi-conducteurs. La principale utilisation de la photo résistance est la mesure de l'intensité de la lumière (Appareil photo, de comptage, système d'alarme ... etc). Elle est fortement concurrencée par la photodiode.

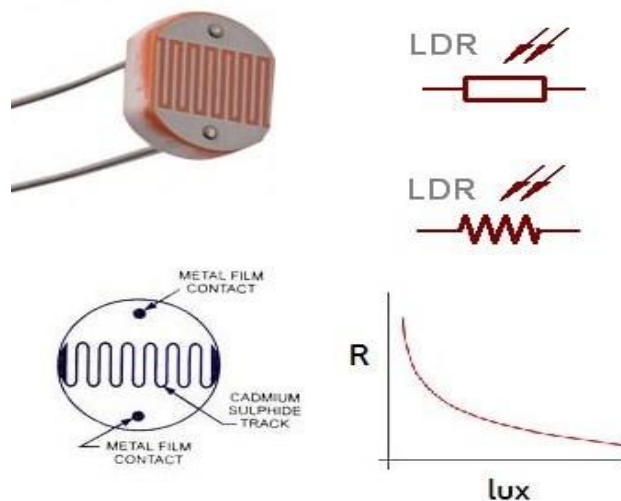


Figure 9 : LDR et son symbole

Un cristal à semi-conducteur à basse température contient peu d'électrons libres. La conductivité du cristal est très faible, proche de celle d'un isolant. Lorsque la température du cristal augmente. De plus en plus d'électrons qui étaient immobiles dans les liaisons covalentes s'échappent et peuvent participer à la conduction.

A température ambiante si le même cristal semi-conducteur est soumis à une radiation lumineuse, l'énergie apportée par les photons peut suffire à libérer certains électrons utilisés dans les liaisons covalentes entre atomes du cristal. Plus le flux lumineux sera intense, plus le nombre d'électrons libres pour assurer la conduction sera grand, ainsi la résistance de la LDR est inversement proportionnelle à la lumière reçue. La sensibilité dépend de la fréquence de la radiation lumineuse. Pour convertir cette variation de la lumière en une tension on utilise le circuit diviseur de tension.

3.2. Branchement :

Reliez une patte au +5V de l'Arduino puis reliez l'autre patte à une résistance 10K vers le GND de l'Arduino. Et enfin de la seconde patte vers le A0 de l'Arduino.

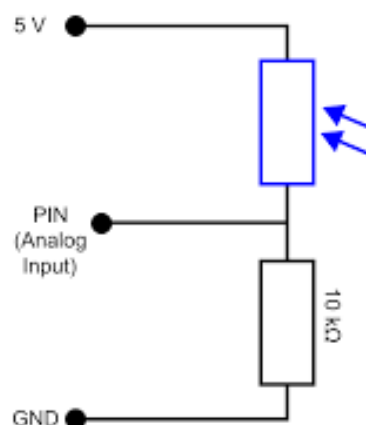


Figure 10 : Branchement du LDR

4. LABDEC - Platine d'expérimentation :

LABDEC est un dispositif qui permet de réaliser le prototype d'un circuit électronique et de le tester. L'avantage de ce système est d'être totalement réutilisable, car il ne nécessite pas de soudure. On peut de plus câbler sur une platine d'expérimentation une grande variété de composants afin de réaliser des circuits électroniques, du plus simple circuit jusqu'au microprocesseur.

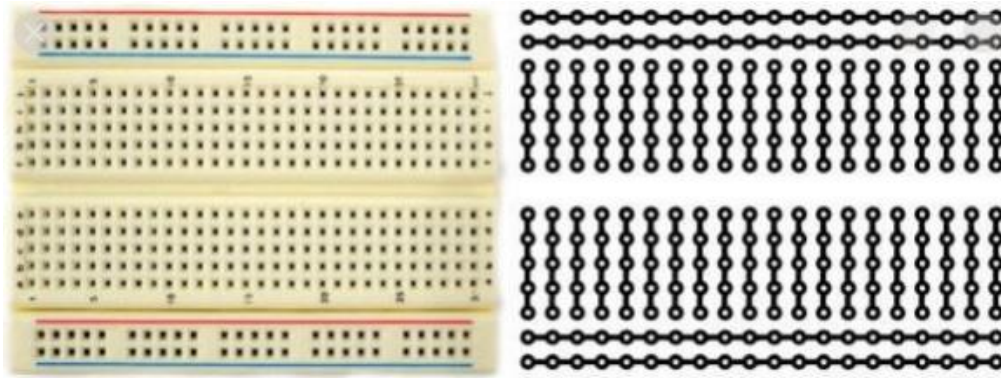


Figure 11 : LABDEC

5. Resistance :

Une résistance est un composant électronique ou électrique dont la principale caractéristique est d'opposer une plus ou moins grande résistance (mesurée en ohms) à la circulation du courant électrique. La résistance électronique est l'un des composants primordiaux dans le domaine de l'électricité.

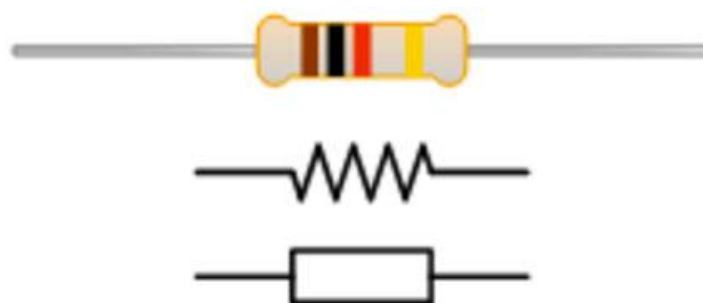


Figure 12 : Resistance et son symbole

- **Code couleur :** Pour connaître la valeur ohmique d'une résistance, il faut identifier les couleurs présentes sur la résistance et l'associer au code universel des couleurs. La norme internationale **CEI 60757**, intitulée *Code de désignation de couleurs (1983)*, définit un code de couleur qui est apposé sur les résistances, les condensateurs (et d'autres composants). Ce code définit la valeur des résistances, condensateurs, etc...

Table du code des couleurs des résistances :

	1° anneau gauche	2° anneau gauche	Dernier anneau gauche	Anneau droite
	1er chiffre	2e chiffre	Multiplicateur	Tolérance
Noir	0	0	1	
Marron	1	1	10	1%
Rouge	2	2	10^2	2%
Orange	3	3	10^3	
Jaune	4	4	10^4	
Vert	5	5	10^5	0.5%
Bleu	6	6	10^6	0.25%
Violet	7	7	10^7	0.1%
Gris	8	8	10^8	0.05%
Blanc	9	9	10^9	
or			0.1	5%
argent			0.01	10%

Exemple : Avec cette résistance électronique :

Les couleurs sont les suivantes : Jaune, Violet, Rouge, Or.

Et elles se traduisent par : "4", "7", " 10^2 ", et une précision de 5%.

Ainsi : $47 * 10^2 = 4700\Omega = 4,7k\Omega$.

On a donc une résistance de $4,7k\Omega$ de précision 5%.



6. Interrupteur à glissière – Slide switch :

Les interrupteurs à glissière permettent de contrôler le flux de courant dans un circuit. Ils utilisent généralement une glissière mécanique pour activer et désactiver un courant en passant entre les états ouvert et fermé. Ils sont adaptés pour contrôler le flux de courant dans les circuits plus petits. Les interrupteurs à glissière sont souvent utilisés comme des commutateurs de marche/arrêt primaires dans les petits appareils électriques alimentés par batterie.



Figure 13 : Interrupteur à glissière

6.1. Principe de fonctionnement :

Ils fonctionnent de manière très similaire à celle des interrupteurs à bouton-poussoir, mais leur forme tactile est plus évidente, ce qui permet à l'utilisateur de différencier avec plus de confiance les états de marche et arrêt.

La gamme RS d'interrupteurs à glissière accueille des tensions nominales entre 4 V c.c. et 400 V c.a., avec des résistances de contact de 10 milliohms à 10 ohms. Actions de commutation Les commutateurs à coulisse sont généralement utilisés pour les applications de marche et d'arrêt simples et le passage entre l'état normalement ouvert (NO) et l'état normalement fermé (NF), mais d'autres fonctions de commutation sont disponibles dans la gamme RS.

Ces applications incluent :

- ✓ Commutation permanente : un fonctionnement de commutation permanent garantit la position physique du commutateur à l'état actionné. Idéal pour les applications d'actionnement permanent telles que la mise en marche et en arrêt.
- ✓ Commutation non court-circuitante (BBM) : cette configuration garantit que le contact dans le commutateur est interrompu avant l'établissement d'un nouveau chemin de connexion, ce qui est généralement préférable dans les applications de puissance.
- ✓ Commutation court-circuitante (MBB) : cette configuration garantit l'établissement d'un nouveau chemin de connexion avant l'interruption de la connexion précédente. Cela signifie que le contact central d'un commutateur est connecté momentanément aux deux contacts, garantissant qu'aucune interruption de l'alimentation ne se produise.

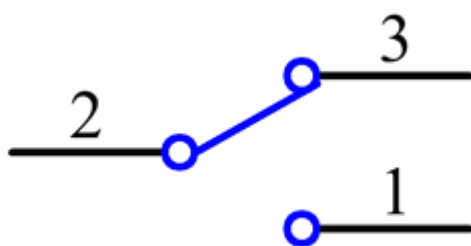


Figure 14 : Symbole d'Interrupteur à glissière

6.2. Branchement :

Reliez une patte au +5V de l'Arduino puis reliez l'autre patte au GND de l'Arduino. Et enfin de la seconde patte à une résistance vers un pin.

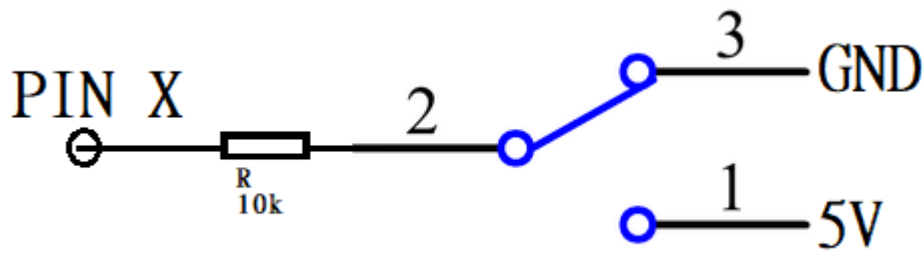


Figure 15 : Branchement d'un interrupteur à glissière

7. LED - Light-emitting diode :

7.1. Principe de fonctionnement :

Une diode électroluminescente est un dispositif opto-électronique capable d'émettre de la lumière lorsqu'il est parcouru par un courant électrique. Une diode électroluminescente ne laisse passer le courant électrique que dans un seul sens et produit un rayonnement monochromatique ou polychromatique non cohérent par conversion d'énergie électrique lorsqu'un courant la traverse. La longue patte est le positif (Anode) et celle courte est le négatif (Cathode).

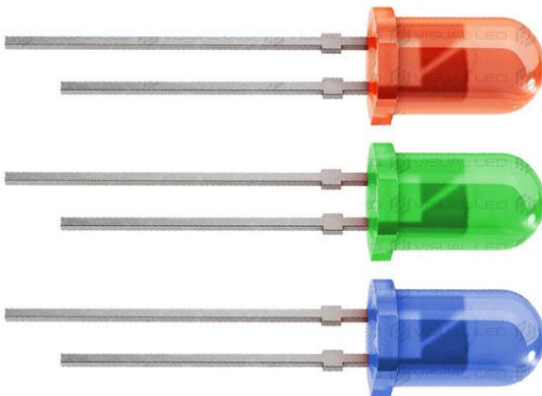


Figure 16 : LEDs

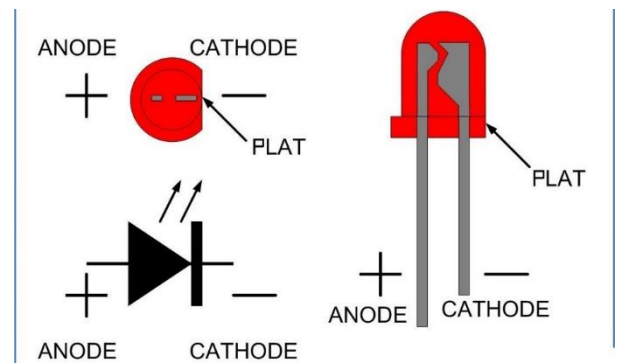


Figure 17 : LED et son symbole

7.2. Branchement :

On connecte le pin de la carte Arduino à la patte de la résistance.

Puis, on connecte la deuxième patte de la résistance à l'anode (borne +) de la LED.

Finalement on branche la cathode (borne -) de LED à la GND de l'Arduino.

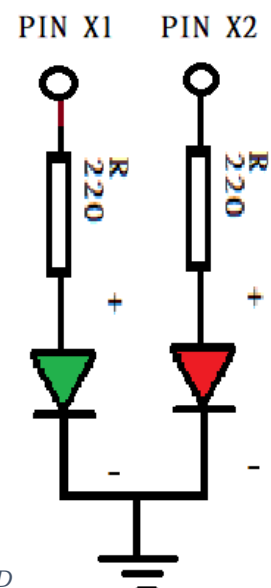


Figure 18 : Branchement d'une LED

8. Bipeur – Buzzer piézo-électrique :

Un buzzer ou bipeur est un élément électromécanique ou piézoélectrique qui produit un son caractéristique quand on lui applique une tension : le bip. Certains nécessitent une tension continue, d'autres nécessitent une tension alternative.



Figure 19 : Buzzer

8.1. Principe de fonctionnement :

Il s'agit simplement de l'assemblage dans un même boîtier, d'un transducteur piézo-électrique et d'une électronique de commande (générateur de signal rectangulaire). Le tout s'alimente alors avec une simple tension continue, généralement comprise entre 3V et 20V, et requiert un courant compris entre 10 mA et 30 mA (la consommation du buzzer dépend principalement de la tension utilisée).

8.2. Branchement :

Le buzzer fonctionne en numérique. Il faudra donc le relier à un pin sur Arduino. Le montage consiste à relier l'alimentation (5V et GND) et une sortie digitale du Arduino au buzzer. La résistance est facultative.

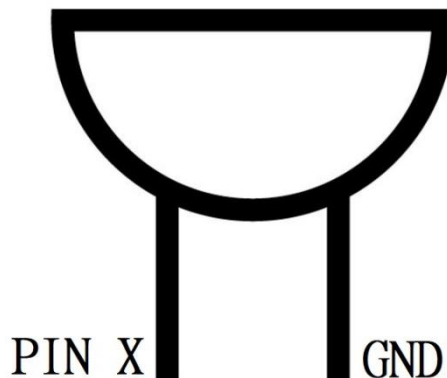


Figure 20 : Branchement d'un Buzzer

Partie B : Réalisation du suiveur de lumière

Après avoir décrit la partie matérielle de notre système, nous sommes arrivés à la partie la plus importante de notre rapport qui est la réalisation du système de poursuite lumière.

1. Fonctionnement de système :

Le principe consiste à remarquer que la lumière reçue par les photorésistances est identique dès que le capteur est orienté dans l'axe de la lumière. Il faut donc mesurer la lumière reçue par chaque photorésistance en mesurant la tension à leurs bornes. Cette mesure est réalisée grâce à la connexion des ports pin analogique de A0 à A3 de la carte Arduino. Le programme compare ensuite ces tensions en ordonnant aux servomoteurs de pivoter dans la direction des photorésistances qui reçoivent le plus de lumière.

2. Schéma du Circuit Complet :

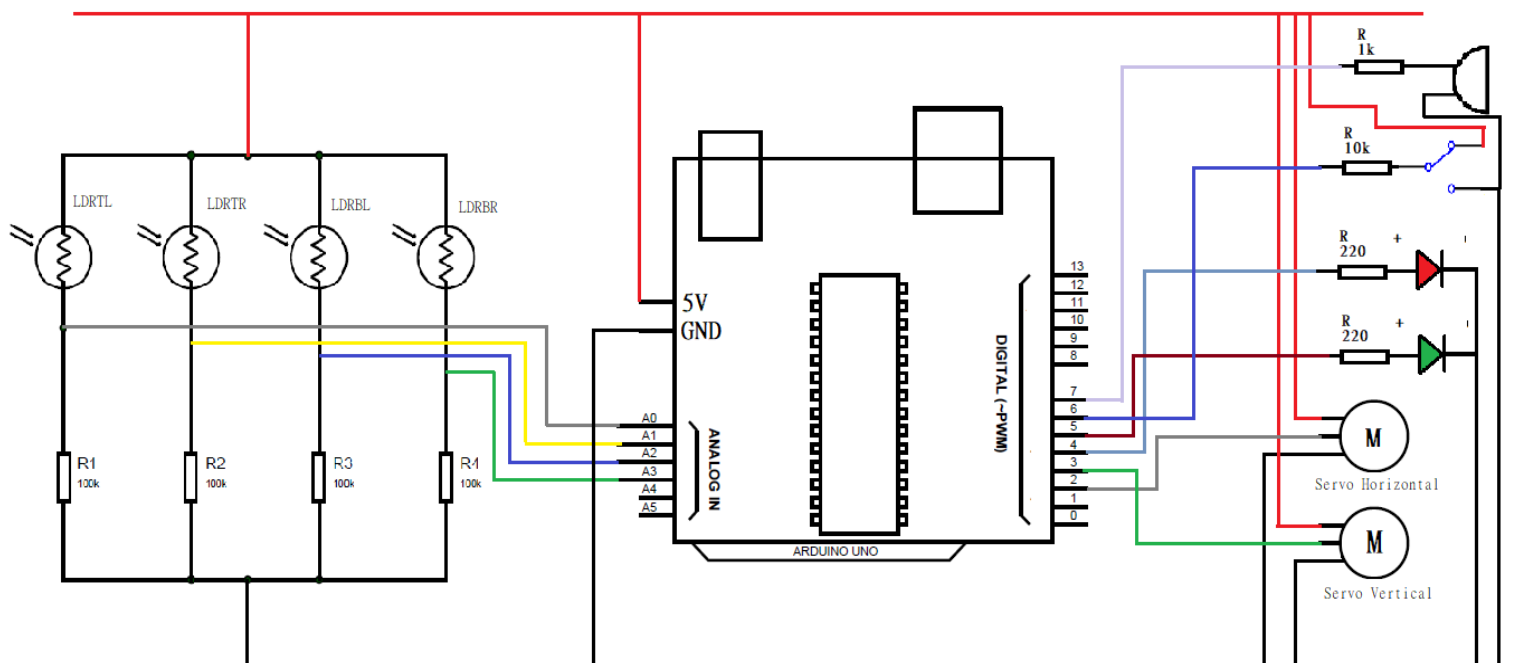


Figure 21 : Schéma du circuit final

3. Câblage du système étape par étape :

3.1. Etape 1 :

Attachement du panneau solaire et des capteurs LDR sur un morceau de carton.

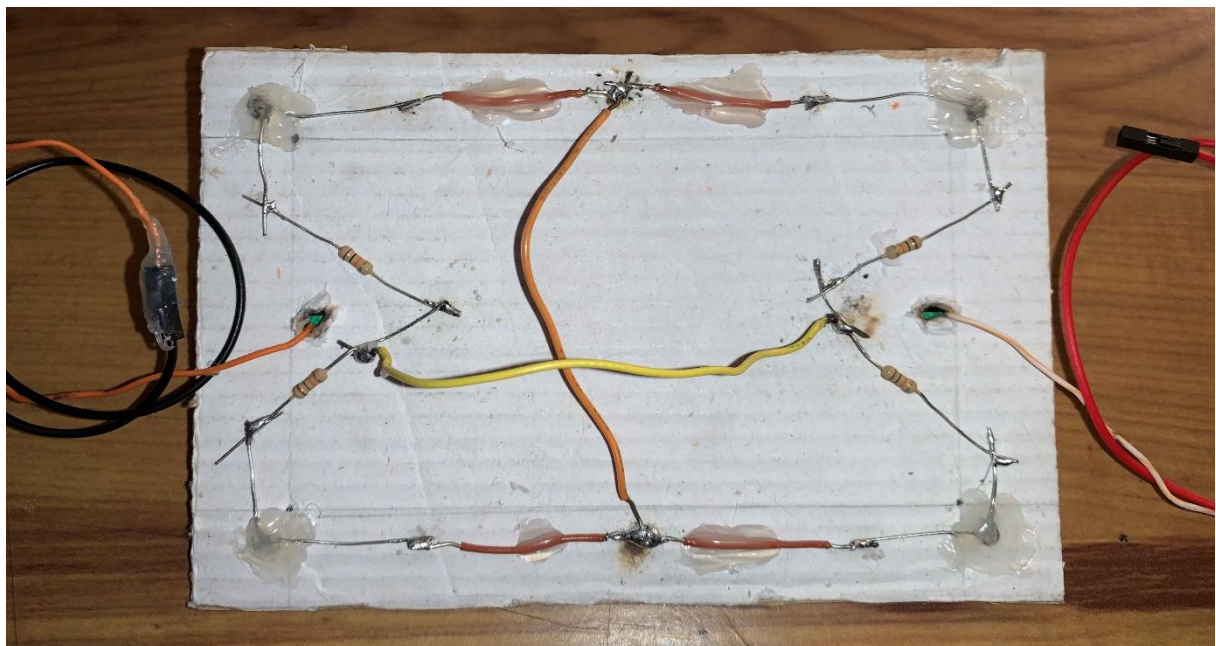
Les capteurs LDR doivent être sur les 4 côtés extrêmes du panneau.



3.2. Etape 2 :

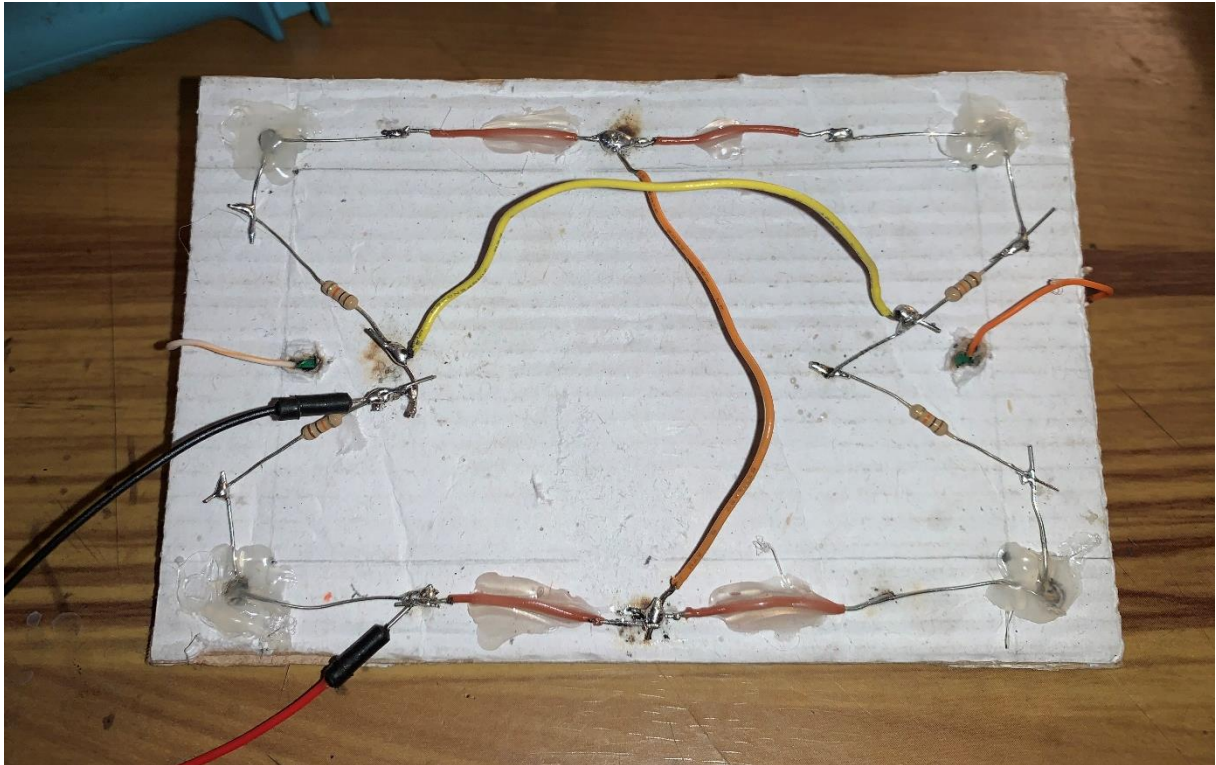
Réalisation du circuit suivant sur l'autre côté du carton.

Les fils rouge et noir appartiennent au panneau solaire.



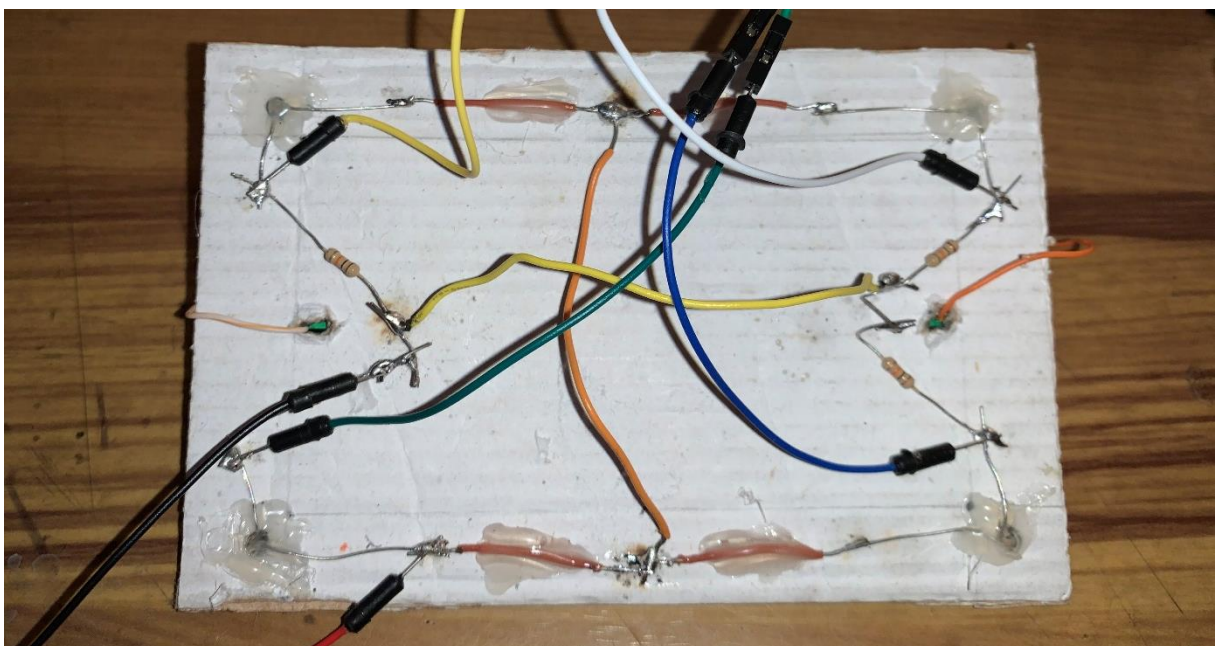
3.3. Etape 3 :

Attachement de deux fils qui vont être attacher au 5V et GND de l'Arduino. Le 5V doit être attacher entre les capteurs et le GND doit être attacher entre les résistances



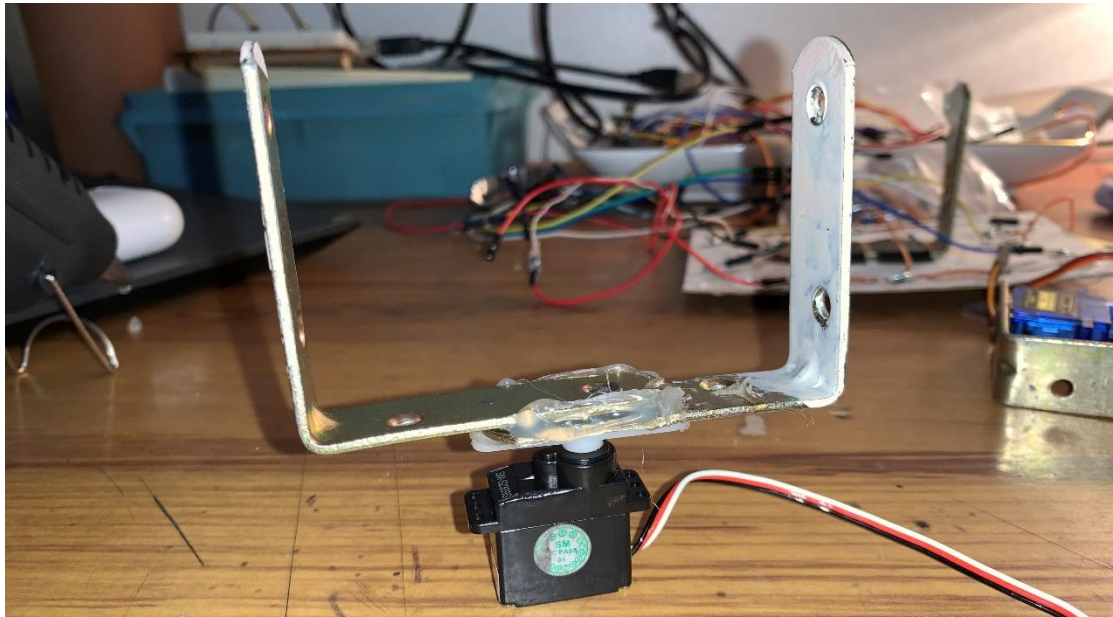
3.4. Etape 4 :

Attachement de quatre fils pour les LDR qui vont être attacher aux pins PWD de l'Arduino. Ces fils doivent être câbler entre le capteur et la résistance.



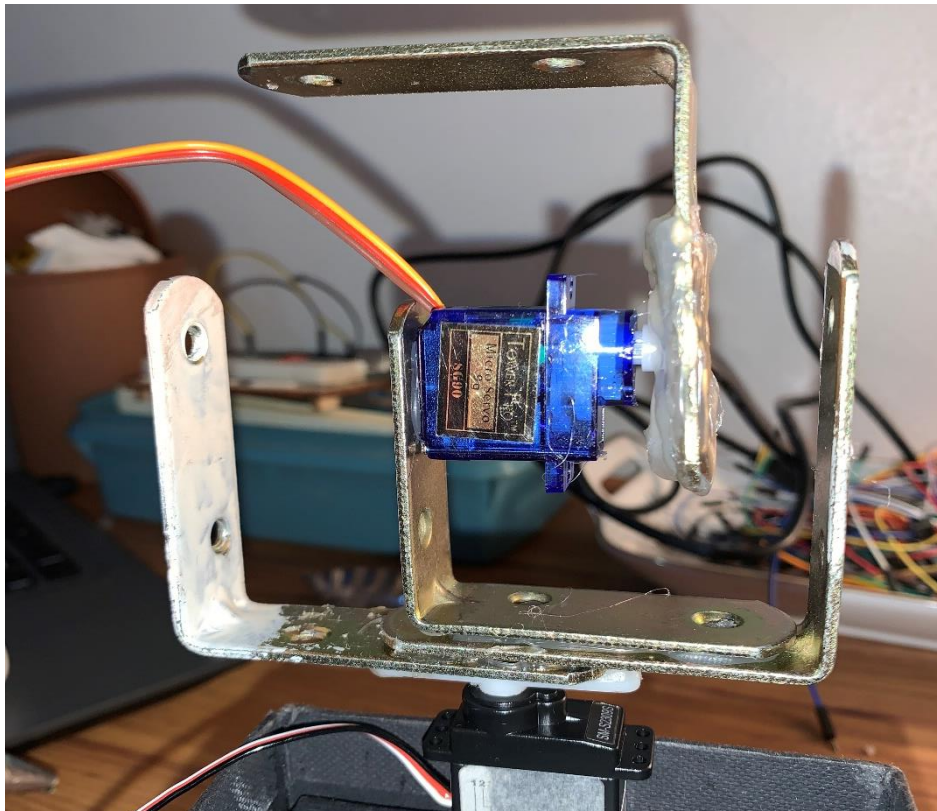
3.5. Etape 5 :

Attachement du premier Servo moteur à une barre de métal qui va porter le dispositif.
Ce Servo moteur est responsable de la rotation droite et gauche.



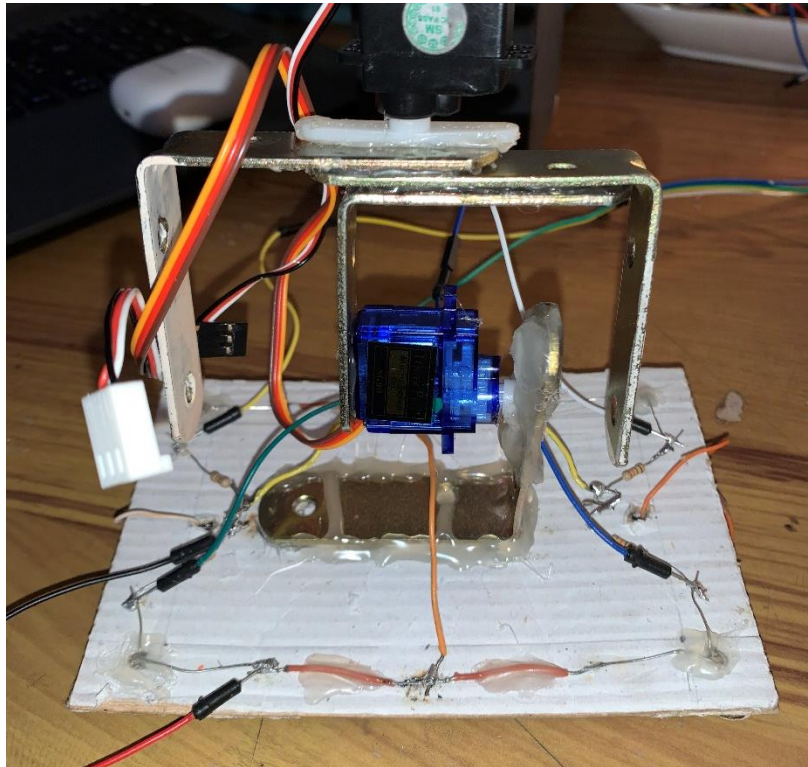
3.6. Etape 6 :

Attachement du deuxième Servo moteur à une barre de métal qui va porter le panneau et les capteurs. Et attachement du tout sur la première barre. Ce Servo moteur est responsable de la rotation en haut et en bas.



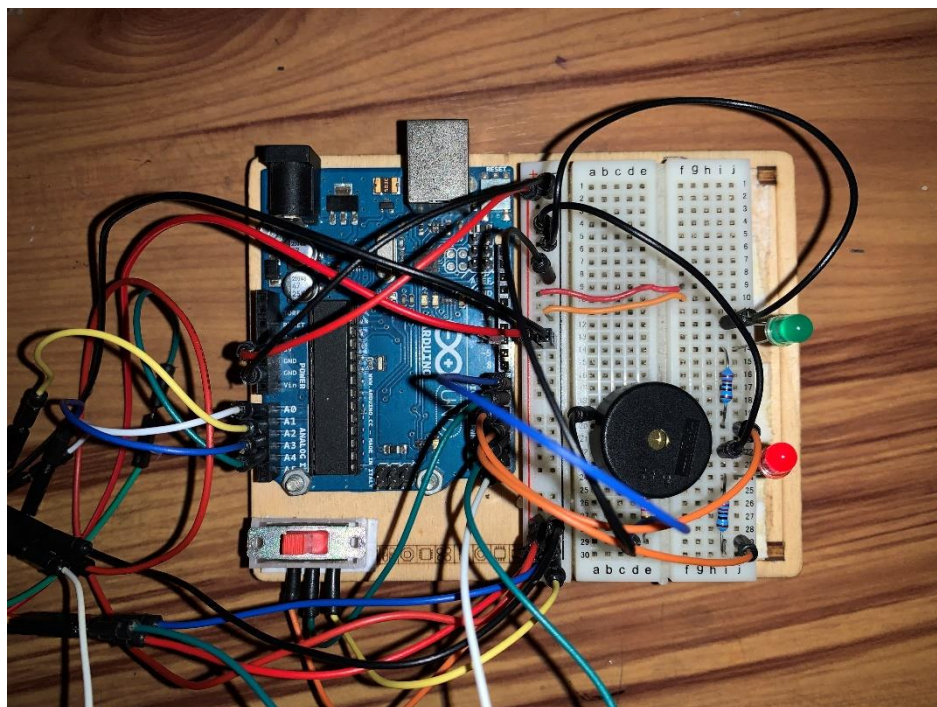
3.7. Etape 7 :

Attachement de la deuxième barre au carton.



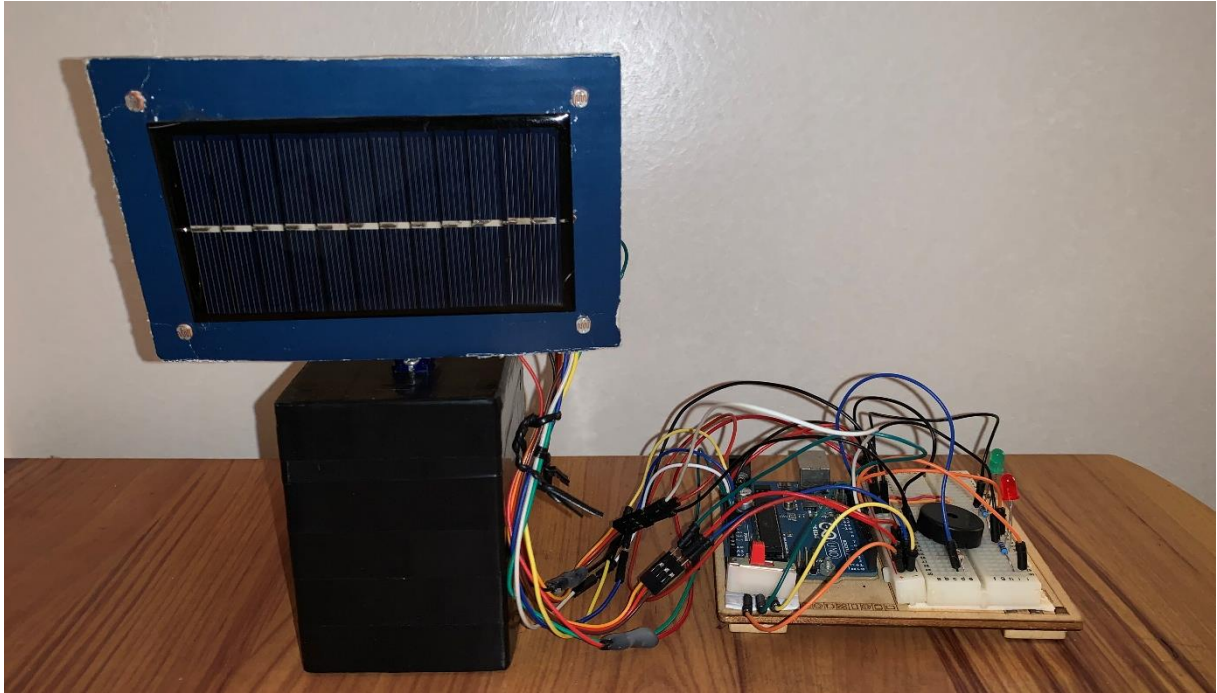
3.8. Etape 8 :

Câblage des LEDS, du Buzzer et du Switch sur LABDEC avec l'Arduino.



3.9. Etape 9 :

Attachement des fils de la première partie à l'Arduino pour figer les deux parties tout en installant le dispositif sur une base pour le supporter.



4. Circuit Complet sur Tinkercad :

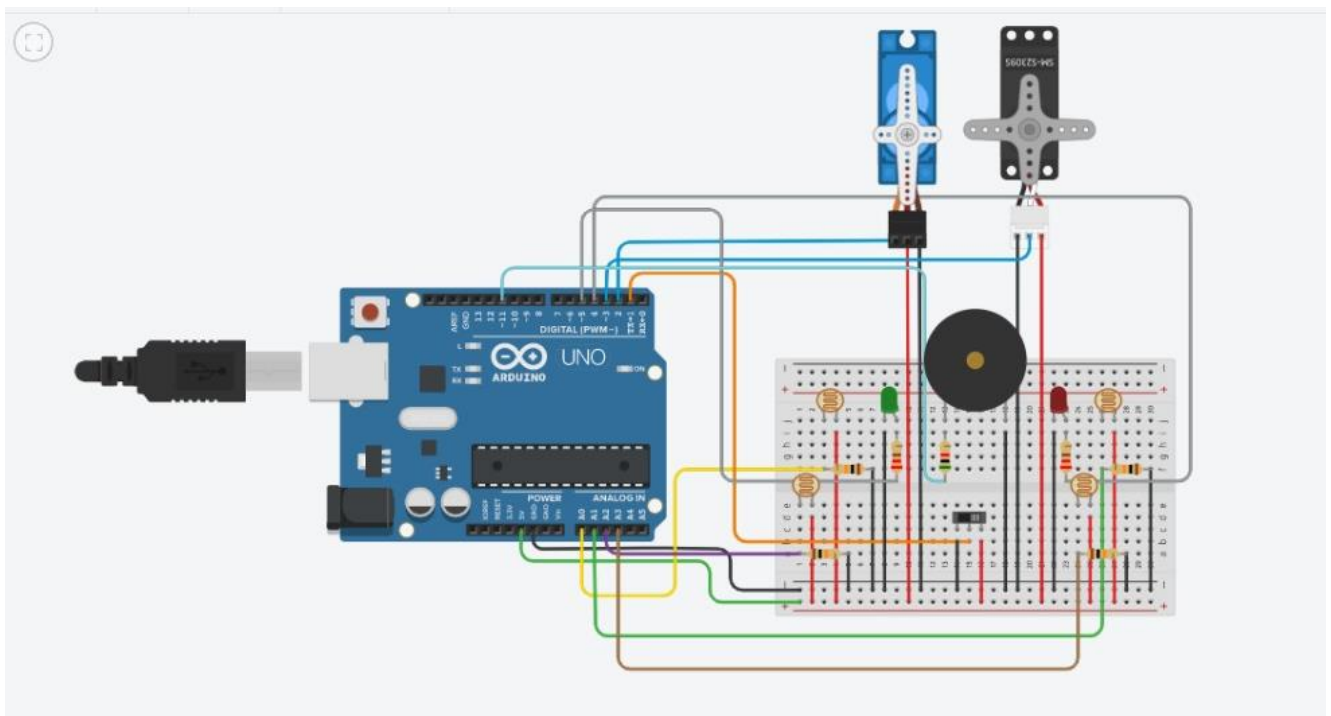


Figure 22 : Circuit final sur Tinkercad

Chapitre III : Programmation et solution logicielle

Après avoir décrit la partie matérielle de notre système et son câblage, nous sommes prêts à développer la partie programmation de système de poursuite lumière.

1. Logiciel Arduino :

L'open-source Arduino Software (IDE), il est facile d'écrire du code et de le transférer à la carte. Il fonctionne sur Windows, Mac OS X et Linux. L'environnement est écrit en Java et basé sur le traitement et d'autres logiciels open-source. Ce logiciel peut être utilisé avec toute carte Arduino.

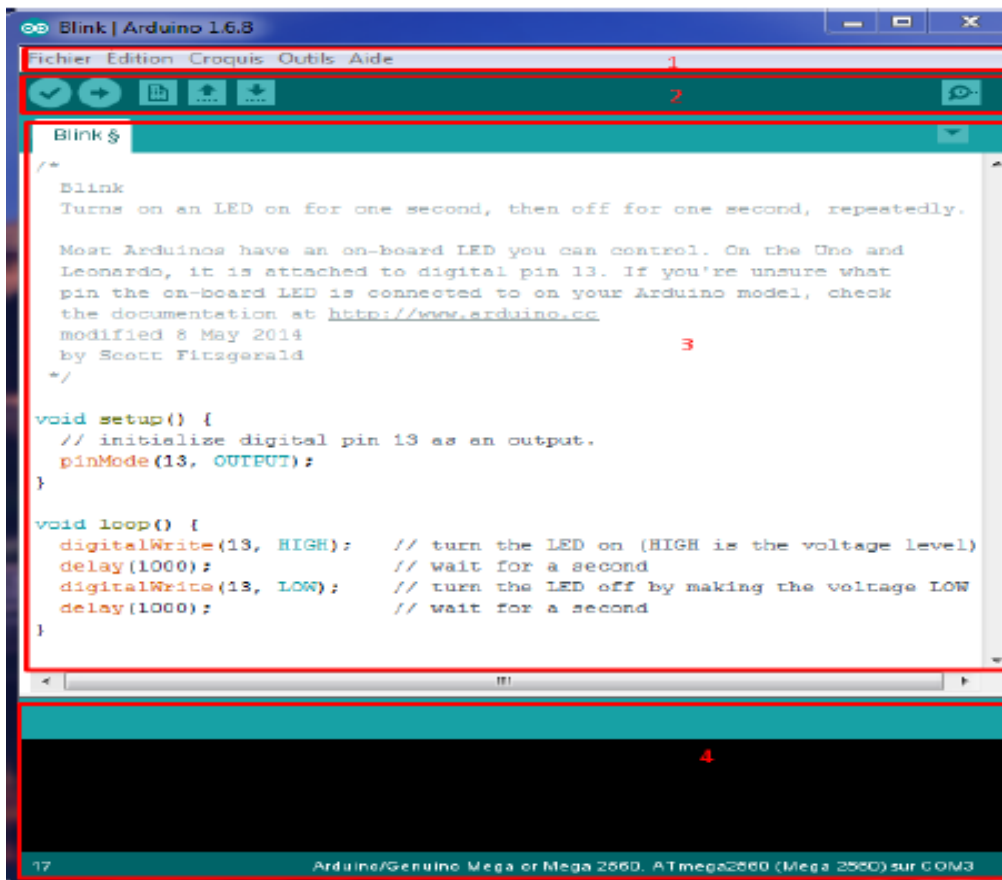


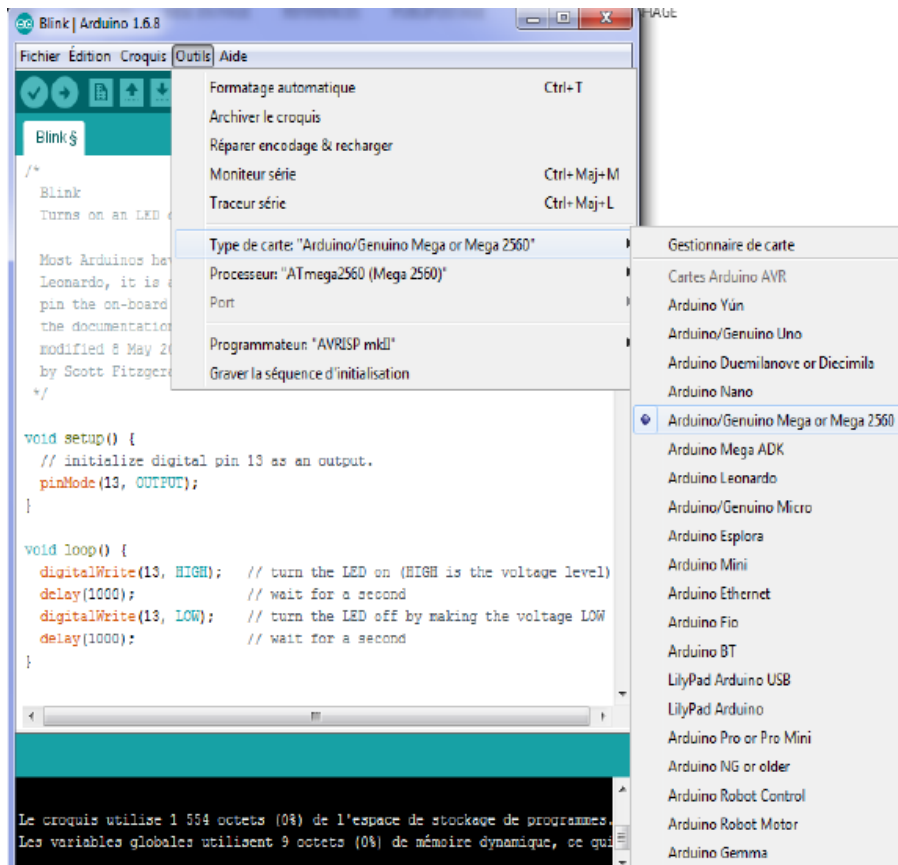
Figure 23 : Présentation de l'interface initiale du logiciel

- ✓ Le cadre numéro 1 : ce sont les options de configuration du logiciel.
- ✓ Le cadre numéro 2 : il contient les boutons qui vont nous servir lorsque l'on va programmer nos cartes.
- ✓ Le cadre numéro 3 : ce bloc va contenir le programme que nous allons créer.
- ✓ Le cadre numéro 4 : celui-ci est important, car il va nous aider à corriger les fautes dans notre programme. C'est le débogueur.

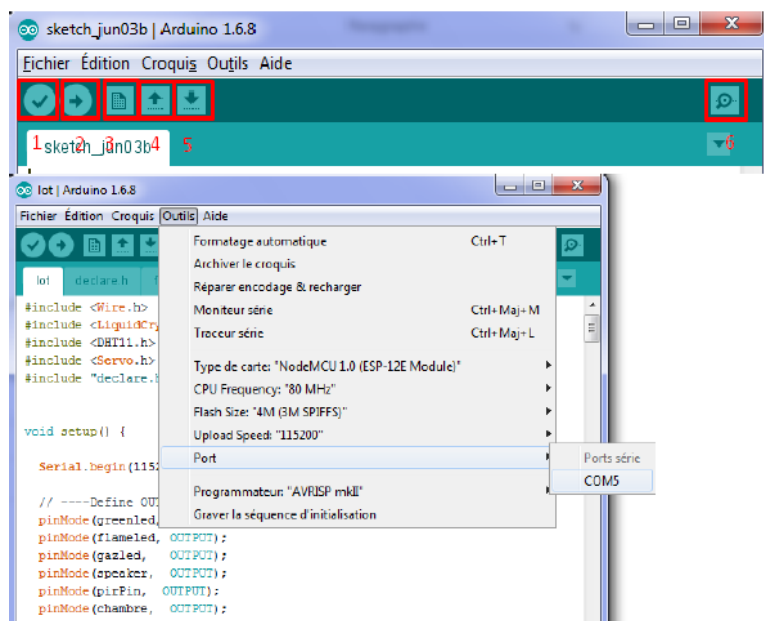
2. Arduino Setup :

Dans un premier temps on va Télécharger le logiciel Arduino gratuitement depuis le sitearduico.cc, ensuite lancer le programme.

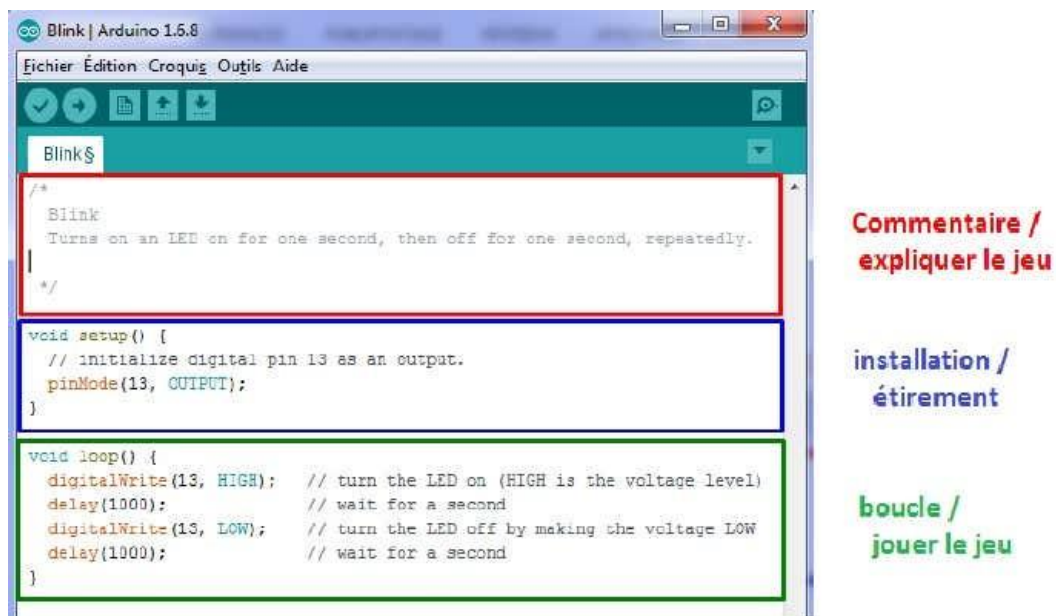
- On doit d'abord choisir le type de la carte. Dans notre cas, Arduino UNO.



- Après, on doit choisir le type do port.



- ✓ Bouton 1 : Ce bouton permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme.
 - ✓ Bouton 2 : Charge (téléverser) le programme dans la carte Arduino.
 - ✓ Bouton 3 : Crée un nouveau fichier.
 - ✓ Bouton 4 : Ouvre un fichier.
 - ✓ Bouton 5 : Enregistre le fichier.
 - ✓ Bouton 6 : Ouvre le moniteur série.
- Il existe 3 parties principales d'un code :



3. Les fonctions Arduino utilisées :

- `digitalWrite(pin,HIGH/LOW);`
- `digitalRead(pin);`
- `analogRead(pin) ;`
- `Serial.println() ;`
- `myservo.attach(pin);`
- `myservo.write([angle]) ;`
- `myservo.read() ;`
- `tone(pin, frequence,durée) ;`
- `noTone(pin) ;`
- `delay([durée en ms]) ;`

4. Code complet :

```
#include<Servo.h>

//predefinition des notes pour le buzzer
#define NOTE_B0 31
#define NOTE_C1 33
#define NOTE_CS1 35
#define NOTE_D1 37
#define NOTE_DS1 39
#define NOTE_E1 41
#define NOTE_F1 44
#define NOTE_FS1 46
#define NOTE_G1 49
#define NOTE_GS1 52
#define NOTE_A1 55
#define NOTE_AS1 58
#define NOTE_B1 62
#define NOTE_C2 65
#define NOTE_CS2 69
#define NOTE_D2 73
#define NOTE_DS2 78
#define NOTE_E2 82
#define NOTE_F2 87
#define NOTE_FS2 93
#define NOTE_G2 98
#define NOTE_GS2 104
#define NOTE_A2 110
#define NOTE_AS2 117
#define NOTE_B2 123
#define NOTE_C3 131
#define NOTE_CS3 139
#define NOTE_D3 147
#define NOTE_DS3 156
#define NOTE_E3 165
#define NOTE_F3 175
#define NOTE_FS3 185
#define NOTE_G3 196
#define NOTE_GS3 208
#define NOTE_A3 220
#define NOTE_AS3 233
#define NOTE_B3 247
#define NOTE_C4 262
#define NOTE_CS4 277
#define NOTE_D4 294
#define NOTE_DS4 311
#define NOTE_E4 330
#define NOTE_F4 349
#define NOTE_FS4 370
#define NOTE_G4 392
#define NOTE_GS4 415
#define NOTE_A4 440
#define NOTE_AS4 466
#define NOTE_B4 494
#define NOTE_C5 523
#define NOTE_CS5 554
#define NOTE_D5 587
#define NOTE_DS5 622
#define NOTE_E5 659
```

```

#define NOTE_F5 698
#define NOTE_FS5 740
#define NOTE_G5 784
#define NOTE_GS5 831
#define NOTE_A5 880
#define NOTE_AS5 932
#define NOTE_B5 988
#define NOTE_C6 1047
#define NOTE_CS6 1109
#define NOTE_D6 1175
#define NOTE_DS6 1245
#define NOTE_E6 1319
#define NOTE_F6 1397
#define NOTE_FS6 1480
#define NOTE_G6 1568
#define NOTE_GS6 1661
#define NOTE_A6 1760
#define NOTE_AS6 1865
#define NOTE_B6 1976
#define NOTE_C7 2093
#define NOTE_CS7 2217
#define NOTE_D7 2349
#define NOTE_DS7 2489
#define NOTE_E7 2637
#define NOTE_F7 2794
#define NOTE_FS7 2960
#define NOTE_G7 3136
#define NOTE_GS7 3322
#define NOTE_A7 3520
#define NOTE_AS7 3729
#define NOTE_B7 3951
#define NOTE_C8 4186
#define NOTE_CS8 4435
#define NOTE_D8 4699
#define NOTE_DS8 4978
#define REST 0

// création des servo moteurs
Servo myservoH;//responsable du rotation droite et gauche
Servo myservoV;//responsable du rotation en haut et en bas

int button;//variable pour l'etat du switch
//variables pour manipulation de buzzer
int start = 0;
int end = 0;

//les angles initiales des servo moteurs
int angleH = 80;
int angleV = 80;
//les pins des LDR
const int LDRTL = 0;
const int LDRTR = 1;
const int LDRBL = 2;
const int LDRBR = 3;

//les angles maximaux et minimal pour protection de dispositif
int angleV_max = 150;
int angleV_min = 30;
int angleH_max = 180;
int angleH_min = 0;

```

```

//des variables pour stocker les valeurs des LDRs
int val_LDRTL;
int val_LDRTR;
int val_LDRBL;
int val_LDRBR;

//des variables pour stocker la moyenne calculer des LDR
//pour avoir la direction meilleur pour faire face à la lumière
int moy_TOP;
int moy_BOTTOM;
int moy_LEFT;
int moy_RIGHT;

int melody_start[] = {
  //start up sound
  NOTE_E5,8, NOTE_E5,8, REST,8, NOTE_E5,8, REST,8, NOTE_C5,8, NOTE_E5,8, //1
  NOTE_G5,4, REST,4, NOTE_G4,8, REST,4,
};

int melody_end[] = {
  //game over sound
  NOTE_C5,-4, NOTE_G4,-4, NOTE_E4,4, //45
  NOTE_A4,-8, NOTE_B4,-8, NOTE_A4,-8, NOTE_GS4,-8, NOTE_AS4,-8, NOTE_GS4,-8,
  NOTE_G4,8, NOTE_D4,8, NOTE_E4,-2,
};

// sizeof gives the number of bytes, each int value is composed of two bytes (16 bits)
// there are two values per note (pitch and duration), so for each note there are four bytes
int notes_start = sizeof(melody_start) / sizeof(melody_start[0]) / 2;
int notes_end = sizeof(melody_end) / sizeof(melody_end[0]) / 2;

// change this to make the song slower or faster
int tempo = 200;

// change this to whichever pin you want to use
int buzzer = 7;

// this calculates the duration of a whole note in ms
int wholenote = (60000 * 4) / tempo;

int divider = 0, noteDuration = 0;

void setup()
{
  Serial.begin(9600); //debit de communication
  pinMode(6, INPUT); //Switch
  pinMode(4, OUTPUT); //LED rouge
  pinMode(5, OUTPUT); //LED vert
  pinMode(7, OUTPUT); //Buzzer
  myservoH.attach(2); //Servo moteur horizontal
  myservoV.attach(3); //Servo moteur vertical
  myservoH.write(angleH); //initialiser Servo moteur horizontal à l'angle 80
  myservoV.write(angleV); //initialiser Servo moteur vertical à l'angle 80
}

```

```

void loop()
{
    button = digitalRead(6);//lecture d'etat du switch

    if(button == 1)//si le switch est en etat on
    {
        //allumer LED vert et etteindre LED rouge
        digitalWrite(5,HIGH);
        digitalWrite(4,LOW);

        //partie du buzzer
        start++;
        end = 0;
        if(start == 1)
        {
            // iterate over the notes of the melody.
            // Remember, the array is twice the number of notes (notes + durations)
            for (int thisNote = 0; thisNote < notes_start * 2; thisNote = thisNote + 2)
            {
                // calculates the duration of each note
                divider = melody_start[thisNote + 1];
                if (divider > 0)
                {
                    // regular note, just proceed
                    noteDuration = (wholenote) / divider;
                }
                else if (divider < 0)
                {
                    // dotted notes are represented with negative durations!!
                    noteDuration = (wholenote) / abs(divider);
                    noteDuration *= 1.5; // increases the duration in half for dotted notes
                }
                // we only play the note for 90% of the duration, leaving 10% as a pause
                tone(buzzer, melody_start[thisNote], noteDuration * 0.9);
                // Wait for the specief duration before playing the next note.
                delay(noteDuration);
                // stop the waveform generation before the next note.
                noTone(buzzer);
            }
        }
        else
        {
            start = 2;
        }

        //lecture des angles des servo moteurs
        angleH = myservoH.read();
        angleV = myservoV.read();

        //lecture des valeurs envoyées par les LDRs
        val_LDRTL = analogRead(LDRTL);
        val_LDRTR = analogRead(LDRTR);
        val_LDRBL = analogRead(LDRBL);
        val_LDRBR = analogRead(LDRBR);

        //Affichage de ces valeurs sur le terminal
        Serial.println(val_LDRTL);
        Serial.println(val_LDRTR);
        Serial.println(val_LDRBL);
        Serial.println(val_LDRBR);
    }
}

```

```

//calcul des moyennes des capteurs LDR pour manipuler les servo moteurs
moy_TOP = (val_LDRTL + val_LDRTR) / 2;
moy_BOTTOM = (val_LDRBL + val_LDRBR) / 2;
moy_LEFT = (val_LDRTL + val_LDRBL) / 2;
moy_RIGHT = (val_LDRTR + val_LDRBR) / 2;

//manipulation des servo moteur en se basant sur les moyennes calculées
if(moy_TOP > moy_BOTTOM)
{
  myservoV.write(angleV + 1);
  if(angleV > angleV_max)
  {
    angleV = angleV_max;
  }
  delay(10);
}
else if(moy_TOP < moy_BOTTOM)
{
  myservoV.write(angleV - 1);
  if(angleV < angleV_min)
  {
    angleV = angleV_min;
  }
  delay(10);
}
else
{
  myservoV.write(angleV);
}

if(moy_LEFT > moy_RIGHT)
{
  myservoH.write(angleH - 1);
  if(angleH < angleH_min)
  {
    angleH = angleH_min;
  }

  delay(10);
}
else if(moy_LEFT < moy_RIGHT)
{
  myservoH.write(angleH + 1);
  if(angleH > angleH_max)
  {
    angleH = angleH_max;
  }
  delay(10);
}
else
{
  myservoH.write(angleH);
}
}

```

```

else if(button == 0)//si le switch est en etat off
{
    //partie LED
    digitalWrite(4,HIGH);//allumer LED rouge
    digitalWrite(5,LOW);//eteindre LED verte

    //partie du buzzer
    end++;
    start = 0;
    if(end == 1)
    {
        // iterate over the notes of the melody.
        // Remember, the array is twice the number of notes (notes + durations)
        for (int thisNote = 0; thisNote < notes_end * 2; thisNote = thisNote + 2)
        {
            // calculates the duration of each note
            divider = melody_end[thisNote + 1];
            if (divider > 0)
            {
                // regular note, just proceed
                noteDuration = (wholenote) / divider;
            }
            else if (divider < 0)
            {
                // dotted notes are represented with negative durations
                noteDuration = (wholenote) / abs(divider);
                noteDuration *= 1.5; // increases the duration in half for dotted notes
            }
            // we only play the note for 90% of the duration, leaving 10% as a pause
            tone(buzzer, melody_end[thisNote], noteDuration * 0.9);
            // Wait for the specief duration before playing the next note.
            delay(noteDuration);
            // stop the waveform generation before the next note.
            noTone(buzzer);
        }
    }
    else
    {
        end = 2;
    }
}
}

```

Conclusion :

La réalisation du suiveur de soleil nécessite un bon choix des composants électronique utilise pour ne pas compliquer le principe de command. Pour conclure, nous pouvons dire que nous avons réussi à réaliser le principal objectif de ce projet, soit monté et faire fonctionner un suiveur de lumière, s'adaptant aux trajectoires de lumière afin d'en récupérer le maximum possible.

Chapitre IV : Bilan de projet et simulation

L'idée de la réalisation de ce suiveur de lumière est d'utiliser des capteurs LDR pour déterminer la direction du maximum du lumière et déplacer le panneau vers cette direction en utilisant les Servo moteurs. Cela est fait par le calcul de rendement des photorésistances, et cette valeur est utilisée pour calculer la moyenne de la lumière entre les quatre côtés de panneau. Le but est d'avoir des moyennes égales ce qui signifie que la lumière est dispersée en égalité, et donc les rayons sont perpendiculaires avec le panneau. Après le calcul des moyennes, on compare ces valeurs et puis on déplace le panneau en utilisant les servos moteurs à gauche, à droite, en haut ou en bas d'une manière qui permet de réaliser le but cité avant qui se présente dans l'égalité des moyennes.

Le boot de dispositif est commandé par le Switch. Quand ce dernier est en état HIGH, la LED rouge s'éteint et la LED vert s'allume, ensuite une mélodie de Startup démarre depuis le Buzzer. Cette mélodie est construite par les fréquences des notes. Ensuite on a le cas du Switch en état LOW, la LED rouge s'allume et la LED vert s'éteint, ensuite une autre mélodie de Shutdown démarre depuis le Buzzer.



Figure 24 : Photo du projet final

Conclusion générale

Le travail présenté dans ce rapport porte sur la « Réalisation d'un Suiveur de lumière à Base d'une carte Arduino UNO » à l'aide des composants électroniques programmable gérés par un microcontrôleur (Arduino). Le but de ce travail réside dans la mise en œuvre d'une solution technique permettant de transformer un panneau photovoltaïque fixe en un panneau mobile afin d'améliorer son rendement.

Pour cela, le système que nous avons pu réaliser s'est avéré capable de suivre la position du soleil et fonctionnel aux environ de 100% ou avec quelques imperfections, malgré les problèmes rencontrés durant la réalisation.

Ce projet d'Arduino était pour nous l'occasions pour examiner les connaissances acquises au cours de notre cursus et de faire un travail multidisciplinaire, associant l'instrumentation, l'électronique, la mécanique, la programmation et même apprendre des nouvelles notions. En perspective, on souhaite d'améliorer notre instrument on lui ajoutant des fonctionnalités de protection et qu'il soit autonome.

Bibliographie

• Mémoires :

- Sukhraj Singh Cheema, « Simulation Studies on Dual Axis Solar Photovoltaic Panel Tracking System ».Thapar University Patiala 2012.
- Mr Arafa Kaddouri, « Conception et réalisation d'un suiveur solaire à deux axes ». Projet de fin d'étude, Université sidi Mohammed ben Abdallah, Fès, Maroc 2015.
- Mr Slama Fateh, « Modélisation d'un système multi générateurs photovoltaïques interconnectés au réseau électrique ». Projet de fin d'étude, Université Ferhat Abbas - Setif-2011
- Mr Bouziane Rabeh, Dokkar Ayoub, « Conception et Réalisation d'un Suiveur Solaire Bi-axial à Base De capteurs de lumière » Projet de fin d'étude, Université Kasdi Merbah , Ouargla, 2016.
- Mr Boughoufala Mohamed, « Conception et réalisation d'un systeme de Suiveur Solaire pour des systèmes photovoltaïques » Université Mohamed Boudiaf, Oran, 2011.
- Mr Bouayed Abderrahmane, « Commande à deux axes d'un panneaux photovoltaïque » Université Abou Beker Belkaid, Tlemcen, 2014.
- K. Kassmi, M. Hamdaoui et F.Olivie, «Conception et modélisation d'un système photovoltaïque adapté par une commande MPPT analogique » Université Mohamed Boudiaf, Oran , Décembre 2007.
- Mr Dharif Ilias, « Optimisation du rendement énergétique d'un panneau solaire par un support suiveur » Université Sidi Mohamed Ben Abdallah, Fès, Maroc ,2014.

• Sites Internet

- <http://planet-terre.ens-lyon.fr/article/bilan-radiatif-terre2.xml>.
- http://www.ecosources.info/dossiers/Types_de_cellules_photovoltaiques.
- <http://tpe-energie-solaire.e-monsite.com/pages/science-et-technique/l-energiephotovoltaique/avantages-et-inconvenients-du-photovoltaique.html>.
- <https://www.energieplus-lesite.be/index.php?id=16690>
- http://www.mon-clubelec.fr/pmwiki_reference_arduino/pmwiki.php?n=Main.MaterielUno.
- <https://www.generationrobots.com/fr/152-arduino>
- <http://www.supinfo.com/articles/single/296-qu-est-ce-qu-servomoteur>

École Marocaine des Sciences De l'Ingénieur de Casablanca

Nom : Jalil
Prénom : Sohaib

Nom : Mouslih
Prénom : Ikram

Titre du mini-
projet :

« Suiveur de lumière »

Résumé :

Notre travail ayant comme objectif l'amélioration du gain en énergie électrique du panneau solaire mobile par rapport au système fixe. Notre dispositif d'instrument sera fixé sur une structure mécanique que nous avons réalisée à base de deux servomoteurs. Pour cette application, nous avons réalisé un système de photorésistance du type 'LDR' qui sont disposés de manière à ce que leur éclairage ne soit identique que si ce système est perpendiculaire avec la lumière. Les signaux issus des capteurs sont transmis aux entrées d'un microcontrôleur de type 'ATmega328' dans « Arduino » qui permet la comparaison des niveaux de tensions pour la commande des deux servomoteurs. En outre, il permet l'orientation du panneau vers le soleil. Enfin, le système que nous avons réalisé est capable de suivre fidèlement la position du Soleil. En plus, pour augmenter le rendement du panneau solaire mobile par rapport au système fixe, les résultats ont été extraordinaires.

Mots clés : Suiveur de lumière, LDR, Servo moteur, Arduino.