

KENTOURI MASSIN

VOISIN THIBAUT

PEIP 2 - G4

BIBLIOGRAPHIE

PROJET ARDUINO : FUTUINO

TABLE DES MATIÈRES :

- 0- INTRODUCTION**
- I- TYPE DE ROUES**
- II- TYPE DE MOTEURS**
- III- COMPOSANTS ELECTRONIQUES**
- IV- SYSTÈME DE TERRAIN FAÇON ARCADE**
- V- FORME DU ROBOT ET ERGONOMIE**
- VI- CONNEXION ENTRE CARTE ARDUINO/TÉLÉPHONE**
- VII- APPLICATION BLUETOOTH**
- VIII- ETAT DE L'ART**
- IX- SYSTÈME DE TIR**
- X- CONCLUSION**

POLYTECH NICE-SOPHIA ANTIPOLOIS

ANNEE SCOLAIRE: 2020-2021

0 - Introduction

L'objectif de notre projet Arduino est de créer deux robots pilotés à distance par nos téléphones qui vont s'affronter dans un match arcade de pseudo-football.

On va chercher à faire les meilleurs choix pour la construction de nos robots et de notre terrain arcade afin que la jouabilité soit digne des plus grands matchs de robot foot de l'histoire. En effet, nos recherches se sont articulées en différents axes.

Tout d'abord nous avons étudié les différents types de roue possible pour que le robot soit adapté à nos demandes concernant le projet. En effet, il doit être stable, dynamique, et pas trop imposant.

Ensuite nous avons étudié les différents choix de moteurs pour pouvoir faire avancer le robot et les mettre en lien avec les roues. De plus, nous avons réfléchi à un système afin que les moteurs soit capable de changer de sens pour qu'ils puissent aller en arrière et en avant automatiquement. Nous avons aussi fait la recherche de tous les composants électroniques nécessaires afin de composer le robot. Nos robots vont s'affronter sur un terrain de foot ils vont donc devoir être équipés d'un système de tir qui va permettre à la balle d'avancer. Tous ces choix nous ont conduit à élaborer l'ergonomie de notre robot qui doit allier stabilité, automaticité et efficacité. Enfin, nous avons réfléchis au système de connexion Bluetooth qui va permettre de piloter nos robots à distance.

Grâce à tous notre travail nous aurons pour objectif de faire un match final entre Thibaut et Massin.

I-Type de roues à utiliser

<http://nagashur.com/blog/2016/01/12/rea-roues-chenilles-transmission-de-puissance/>

I-1 - Deux roues motrices et une roue libre omnidirectionnelle

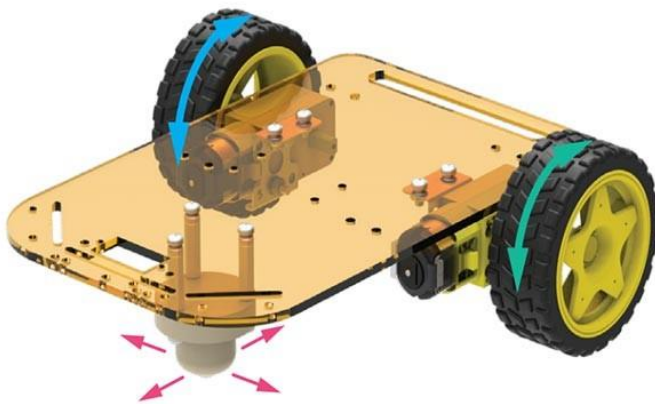
Si on choisit un robot à deux roues motrices, alors il faut un troisième point de contact avec le sol pour que le robot reste en équilibre. Ce point de contact doit être tel qu'il doit générer le moins de frottements possibles.

Ce point de contact peut être :

- une roue simple qui peut tourner des deux côtés comme une roue classique
- une bille (métallique ou en plastique) enfermée dans une base fixe. La bille ne peut donc pas d'échapper de la base mais reste libre dedans. C'est une solution facile à mettre en œuvre mais qui présente davantage de frottement qu'une roue.

Exemple d'un robot basique mobile sur :

<https://thetempedia.com/project/soccer-playing-mobile-robot/>



Pure translation occurs when both wheels move at the same speed

Figure 1 : Exemple d'un robot à roue folle

<https://thetempedia.com/project/soccer-playing-mobile-robot/>

Fonctionnement de la rotation :

Ce type de robot est un robot différentiel, c'est-à-dire un robot qui change de direction en faisant varier la vitesse relative de ses roues. C'est un choix simpliste ne nécessitant pas de mouvement de direction plus développé.

En effet, pour aller en ligne droite, il suffit de faire tourner les roues à la même vitesse dans la même direction.

Pour tourner sur place, il suffit de faire tourner les roues à la même vitesse dans des directions opposées.

Pour tourner, il faut faire tourner les roues à des vitesses différentes dans des directions différentes.

1-2 - 4 roues avec 2n roues motrices

Si on choisit un robot à 2n roues motrices, alors chaque roue possède un moteur (et donc une consommation élevée).

Dans ce cas-là, il faut synchroniser les quatre moteurs pour qu'ils puissent fonctionner ensemble de manière optimale.

Le couple associé aux moteurs est différent s'il s'agit de 4 roues motrices ou bien de 2 : pour 4 roues motrices chaque moteur doit fournir $x/2$ g/cm de couple pour faire avancer le robot tandis que pour 2 roues motrices chaque moteur doit fournir x g/cm de couple.

Le choix des quatre roues peut être intéressant dans le cas où l'endroit sur lequel le robot roule présente des aspérités plus ou moins prononcées ou bien si le robot est créé dans l'optique d'une recherche de vitesse.

Dans notre cas, le robot roulera sur un terrain lisse (terrain de foot). Donc le choix de quatre roues peut être un peu trop poussé dans le cas de notre projet.



Figure 2 : Exemple d'un robot avec $2n$ roues motrices

<https://www.robotshop.com/>

II-2 - Deux roues entraînant une chenille

Si on choisit un robot avec deux roues entraînant une chenille, alors le robot possède 2 ou 4 roues motrices entourées de chenilles devant être synchronisées (et donc un couple assez élevé).

Ce cas-là présente des roues avec une grande surface de contact entre le sol et la chenille.

Ce choix présente de nombreux avantages dans des cas bien précis, en effet la grande surface de contact permet une excellente adhérence avec le sol et un franchissement d'obstacle/ abond de pente importante très simplifié.

Le choix de ces roues peut être intéressant si les avantages proposés par ces roues nous étaient favorables, or ils ne nous sont en aucun cas utile dans l'élaboration de notre projet. Le choix de ces roues dans notre projet ne présenterait que des points négatifs. Donc le choix de ces roues est très peu propice dans le cas de notre projet.

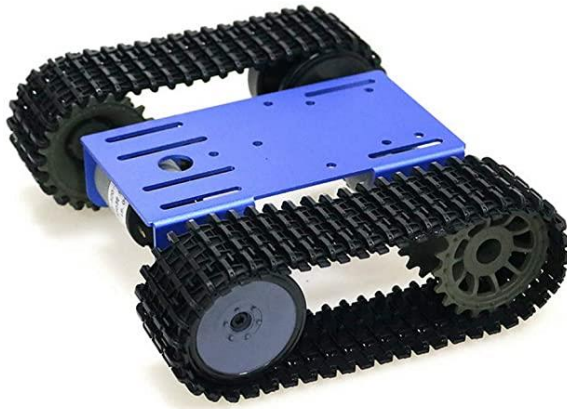


Figure 3 : Exemple d'un robot avec 2 roues entrainant une chenille

<https://www.amazon.fr/>

Conclusion :

Pour conclure sur le choix des roues, avec toutes les informations que nous avons réunies, nous avons opté pour le choix des 2 roues motrices ainsi qu'une roue libre omnidirectionnelle (roue folle). En effet ce choix nous paraît être le plus optimal dans le cadre de notre projet puisque nous ne cherchons pas d'atteindre des vitesses importantes (choix de $2n$ roues motrices). De plus nous ne cherchons pas à gravir d'importants obstacles (choix de deux roues entrainant une chenille). Nous avons donc opté pour ce choix. Nous pensons qu'en tant que 3eme point d'appui pour notre robot, nous allons choisir la roue folle au lieu d'une ball-caster. En effet, la ball-caster permet d'avoir une trajectoire plus améliorée avec plus de précision. Elle aurait été utile dans le cas ou notre robot devrait suivre une trajectoire bien précise, or ce n'est pas notre priorité dans notre projet. Le choix de la roue folle est donc adéquat, elle permet un franchissement d'obstacles amélioré dans l'optique d'éventuelles apérités présents sur les terrains de jeu. Finalement, la roue folle est suffisante dans le cas de notre projet pour un contrôle optimal. De plus elle permet un franchissement d'obstacles amélioré dans l'optique d'éventuelles apérités présents sur les terrains de jeu.

II-Type de moteur possible

<https://www.robot-maker.com/forum/>

Introduction :

Le moteur est une partie essentielle dans le robot pour lui permettre de faire fonctionner et actionner ses roues. Son choix va donc devoir être réfléchi en fonction des avantages et des inconvénients que celui-ci présente. Il existe un grand nombre de choix pour notre moteur en général mais dans le cas de notre projet d'Arduino, le choix se résume en 3 types de moteurs.

II-1 - Pour une rotation continue, les servo-moteur

https://zestedesavoir.com/tutoriels/686/arduino-premiers-pas-en-informatique-embarquee/747_le-mouvement-grace-aux-moteurs/3438_un-moteur-qui-a-de-la-tete-le-servomoteur/

Composition :

Le servo moteur a besoin de trois fils de connexion :

- 1 pour l'entrée du signal de commande
- 2 pour l'alimentation (1 à la masse et l'autre 5V)

Il possède plusieurs engrenages (correspond au réducteur) qui permettent de réduire la vitesse de rotation en sortie de l'axe et d'augmenter le couple en sortie du servomoteur.

Fonctionnement :

Il s'agit d'un moteur à courant continu (rotation infinie) pour pouvoir faire tourner les roues arrière de la voiture. La carte Arduino ne délivre pas assez de puissance pour le moteur donc on doit le brancher à une pile 9V à 12V.

Le servomoteur est un moteur qui fonctionne en courant continu d'intensité réglable. Il est plus précis qu'un moteur pas à pas, et donc plus cher naturellement. En général les moteurs puissants sont plutôt des servomoteurs.

Le moteur pas à pas seul ne peut pas faire l'équivalent d'un servo moteur, le circuit de commande connaît la position du mobile en comptant les pas, mais s'il y a surcharge et perte de pas. Il faut toujours qu'un moteur pas à pas soit largement plus puissant que le strict nécessaire pour éviter le problème.

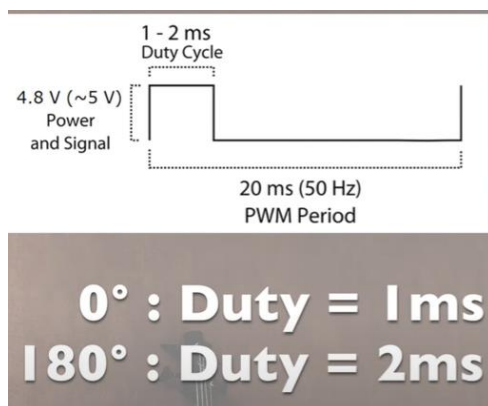
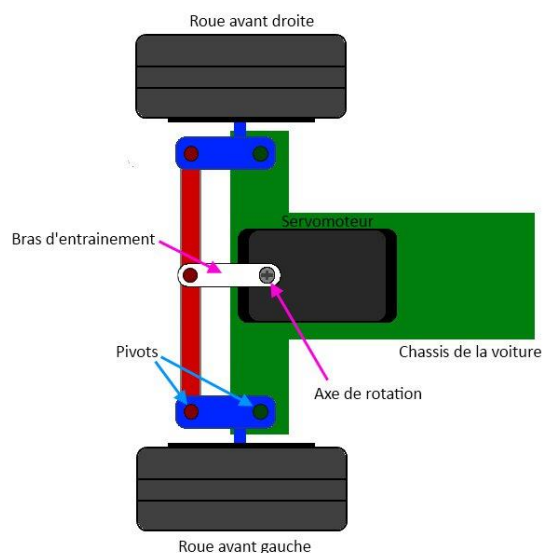


Figure 4 : Signal carré de période 20 ms

Le servo moteur fonctionne en se branchant avec un potentiomètre qui permet de réguler l'angulation avec carte Arduino.

L'utilisation du servo moteur nécessite aussi un programme qui permet de réguler l'angulation des roues.



Pour utiliser le servo moteur, chaque roue est positionnée sur l'axe de rotation (en bleu) qui repose que le châssis et la baguette (pour le parallélisme).

L'utilisation d'un servo moteur peut être intéressant car les deux roues pivotent du même côté en même temps que l'axe de rotation.

En effet lorsque l'axe de rotation se met à tourner il entraîne le bras de rotation lié à la baguette lié aux roues et les fait donc pivoter.

Figure 5 : Schéma simplifié de l'utilisation du servomoteur

<https://eskimon.fr/>

L'avantage d'un tel type de moteur est important, en effet si on devait mettre un moteur à courant continue dans le même setup à la place du servomoteur, cela ne fonctionnerait pas.

En effet le moteur à courant continue tournant sans s'arrêter, les roues peuvent pivoter plus loin que leur angle maximal et donc abimer voire casser le processus et le robot. De plus, cela nécessiterait l'utilisation d'un réducteur pour arriver à avoir une vitesse faible et un couple élevé.

Pour le faire fonctionner, on donne une position angulaire relié à son axe. C'est une utilisation plutôt simple pour faire bouger les roues en même temps du même côté. On remarque que l'angle du servomoteur ne peut pas aller à plus de 180°. On ne pilote pas directement le moteur mais on exige un résultat en sortie. Les roues doivent être parallèle au châssis lorsque le servo moteur est à 90°.

Exemple du servo moteur MG995 :



Figure 6 : Servomoteur MG995

<https://fr.aliexpress.com/>

Ce type de servomoteur est le type de moteur que l'on pourrait utiliser dans le cas de notre projet.

On sait qu'il est à notre disposition préalablement mais il est peut-être utile pour des voitures de plus gros gabarit. On l'essaiera pour voir s'il peut être adéquat pour notre projet. Mais on pense qu'il n'est pas approprié car trop gros.

Caractéristiques de ce moteur :

Dimension (en mm) : 40x19x43 - Vitesse : 0,13s/60deg (pour 6V) – tension : 4.8V – 7.2V

II-2 - Pour une rotation continue, les moteurs DC continu

<https://www.mdp.fr/documentation/lexique/courant-continu/definition.html/>

Ce type de moteur est une alternative au servo moteur, en effet le moteur CC ou DC est un moteur qui permet aussi une rotation continue. Contrairement à son protagoniste, le moteur DC n'a pas besoin de savoir envoyer une consigne précise.

Fonctionnement :

Contrairement aux servomoteurs, ce moteur n'est pas asservi, c'est-à-dire qu'il ne nécessite pas l'envoi de consignes.

Ce moteur ne possède donc que deux fils liés à la masse et à l'alimentation. Il suffit de l'alimenter pour le faire tourner en continu. Pour inverser le sens de rotation du moteur il suffit d'inverser la polarité.

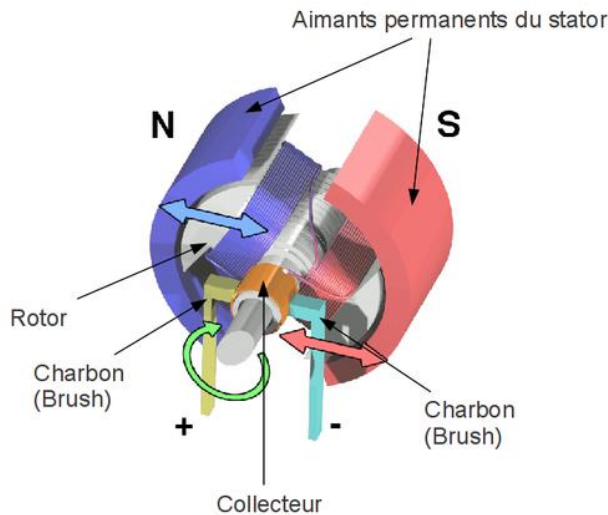


Figure 7 : Schéma représentatif d'un moteur à courant continu :

https://www.wikimeca.org/images/thumb/5/54/Schema_moteur_cc.png/500px-Schema_moteur_cc.png

Avantages et inconvénients :

L'avantage principal de ce type de moteur est son adaptation simple au moyen de régler et faire varier leur vitesse, leur couple ou leur sens de rotation. Ils ne nécessitent pas d'électronique pour les piloter. En effet ils sont facilement contrôlables à distance (Bluetooth) dans notre cas

Ces moteurs tournent généralement trop vite ce qui n'est pas forcément un bon point pour notre projet. Pour contrer ce phénomène ils sont généralement associés à des motoréducteurs permettant de réduire la vitesse et d'augmenter le couple du moteur.

Exemple : Le motoréducteur DC3-V :



Figure 8 : Moteur continue Arduino pour robot voiture

<https://fr.aliexpress.com/>

Ce type de moteur est un moteur à motoréducteur qui est le plus souvent utilisé dans la réalisation de petit robot voiture pour Arduino.

Il ne permet pas d'atteindre des vitesses très élevées mais il peut être adéquat pour notre projet.

Caractéristiques de ce moteur :

Tension : 3V~6V- Couple : 800gf cm/min (à 3V) – Tour par minute : 200 (à 4.5V) - Courant : 170 mA (à 4.5V)

Pour la roue les dimensions sont indiquées sur l'image

II-3 - Pour une rotation continue, les moteurs pas à pas

<https://pecquery.wixsite.com/>

Le moteur pas à pas regroupe les avantages des deux premiers moteurs évoqués précédemment. Il est très précis et est capable de faire plus d'un seul tour.

Composition :

Le moteur pas à pas est constitué :

- d'un rotor aimanté
- enroulements statoriques dans lesquels un champ magnétique est créé lorsqu'ils sont parcourus par un courant électrique.

Fonctionnement :

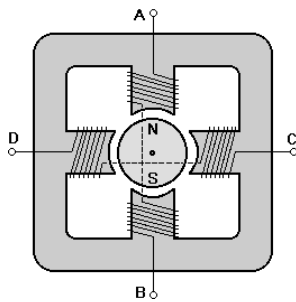


Figure 9 : Moteur pas à pas bipolaire

<https://pecquery.wixsite.com/>

Ce moteur utilise des pas, le pas est l'angle minimal de rotation. Cette notion de pas permet d'avoir un moteur avec une précision accrue.

Pour le piloter, cela nécessite deux fonctions électroniques : un séquenceur qui élabore l'ordre des phases et une interface de puissance pour amplifier le courant pour les bobinages.

Ce moteur possède 4 fils pour commander 2 enroulements opposés.

Le courant circule entre A et B et un autre entre C et D.

Avantages et inconvénients :

Le principal avantage de ce moteur est les pas qu'il possède, en effet cela permet de contrôler facilement le nombre de pas pour avoir une grande précision et ne nécessite pas de démultiplication de puissance. Contrairement à des moteurs CC, les moteurs pas à pas peuvent être piloter précisément et en boucle ouverte.

On remarque que dans le cas de notre projet, ces avantages ne nous sont que très peu profitables, de plus l'utilisation de ce moteur entraîne certains inconvénients comme une consommation de courant élevé ou encore de mauvaises performances lors des changements de charge.

Dans le cadre de notre projet, si on choisit ce moteur, on ne profitera que des inconvénients et que très peu des avantages.

Conclusion :

Pour conclure sur le choix du moteur, avec toutes les informations que nous avons réunies. Nous hésitons encore sur le choix du moteur. Son choix se fera donc en expérimentant deux types de moteur : le servo moteur MG995 ainsi que le motoréducteur DC3-6V jaune.

Nous avons retenu que ces deux types de moteurs car on trouvait que le moteur pas à pas n'était pas conforme à notre demande concernant ce projet.

Nous déciderons donc plus tard en fonctions des divers paramètres (puissance du moteur entraînant la vitesse du robot, consommation, synergie avec les roues ainsi que l'application Bluetooth) lequel de ces deux types de moteurs nous allons utiliser dans notre projet.

Nous avons remarqué que la plupart des projets similaires au notre utilisaient un motoréducteur DC3-6V, nous voulons quand même expérimenter le servo moteur pour en voir les capacités même si dans le cadre de notre projet, ce n'est pas forcément primordiale. Nous expérimenterons donc le servo-moteur mais nous gardons en tête que le motoréducteur DC3V-6V reste la priorité de notre projet, les roues s'emboitant directement sur le moteur cela est plus pratique. Il ne faudra donc pas s'attarder sur l'utilisation d'un servo-moteur.

III-Composants électroniques

<https://www.aranacorp.com/fr/utilisation-du-motor-shield-arduino/>

III-1 - Introduction

Le système du robot sera utilisé avec une carte Arduino UNO que nous avons utilisé en cours tout au long du premier semestre.

III-2 - Capteurs de distance à ultrason

Comme on a vu en cours, le capteur va nous permettre de gérer la distance entre nos deux robots pour pouvoir maintenir des conditions de jeux qui sont en adéquation avec notre cahier des charges. Pour cela, on va pouvoir utiliser la bibliothèque NewPing qui va permettre une prise en charge optimisée et simultanée de plusieurs capteurs de distance à ultrasons.

On utilisera 1 ou 2 par robot qui seront posé sur un servo moteur lui permettant de faire des rotations successives lui permettant de couvrir la zone la plus importante pour pouvoir éviter au maximum les obstacles (robots adverses).

On pourrait hypothétiquement choisir des capteurs de chocs qui vont permettre de détecter les chocs entre les robots. Mais notre objectif est que les robots ne doivent pas rentrer en contact donc les capteurs de distance vont nous permettre de travailler dans la prévention au contraire des capteurs de chocs.

Exemple de capteurs :

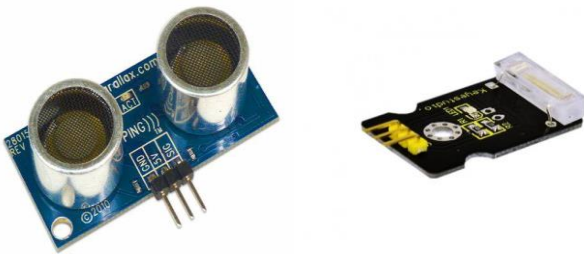


Figure 10 et 11 : Capteur de distance à ultrason et capteur de chocs

III-3 - Choix du pilotage des moteurs

Pour piloter notre robot le moteur du robot doit être capable de changer de sens automatiquement, pour cela l'utilisation d'un pont en H est obligatoire car notre voiture doit être automatique, on ne peut pas se permettre de devoir changer le sens des moteurs manuellement. Pour faire marche arrière et inverser automatiquement le sens du moteur il faut utiliser un système en pont en H.

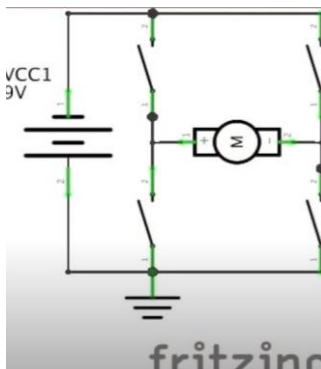


Figure 12 : Système en pont en H

On doit donc trouver un système qui nous permet d'utiliser le pont en H. Pour cela il existe deux composants le L298N et le L293D, qui vont permettre de contrôler la vitesse de rotation et le sens de notre moteur.

L298N :

Le Moteur Shield officiel d'Arduino est une carte d'extension pour le microcontrôleur Arduino UNO pour le pilotage de moteur CC ou encore des servomoteurs.

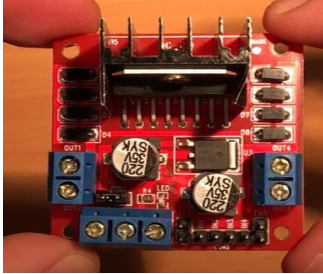


Figure 13 : Moteur Shield L298N

Il utilise le double pont en H L298. Il permet de piloter des moteurs en direction et en vitesse avec une tension nominale entre 5 et 12V, ce qui va entraîner l'ajout à notre système d'une pile de 9V et un courant de 2A, jusqu'à 4A avec une source de tension extérieure.

Ce Shield permet de piloter :

- Jusqu'à deux moteurs à courant continu ou un moteur pas à pas bipolaire.
- Deux capteurs analogiques.
- Deux sorties PWM comme des servomoteurs.

Cette solution serait adaptée pour n'importe quel choix de moteur que nous allons faire. De plus, il permet de piloter deux capteurs analogiques qui seront les capteurs de distance à ultrason dans notre cas. Il a un prix approximatif de 3 euros.

L293D :

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



Figure 14 : Module L293D

Le module Motor Drive Shield L293D est un module en lien avec le microcontrôleur Arduino UNO. Pour piloter nos moteurs nous allons utiliser ce module.

Il est équipé d'un circuit intégré L293D. Ce dispositif permet de piloter deux moteurs à courant continu dans les deux sens de rotation et pour faire de la variation de vitesse. Mais il ne permet pas de piloter des capteurs analogiques. Cette interface accepte une tension d'alimentation allant de 4.5v à 25V et une intensité maximale par moteur de 0.6 A.

Ce qui est en accord avec l'intensité avec l'intensité maximale que nous avons choisis pour nos moteurs. Il est équipé de diodes, de roues libres permettant de protéger les transistors servant à la commande du ou des moteurs.

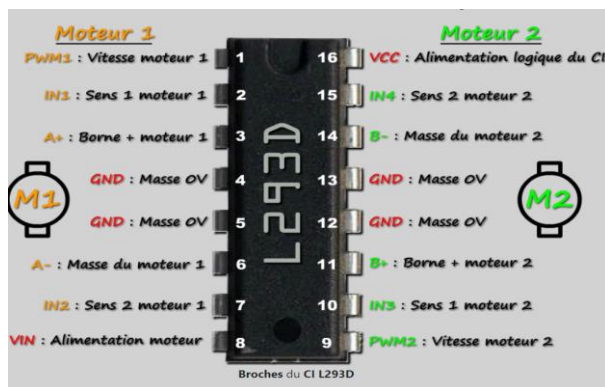


Figure 14 : Ports de branchement du module L293D

Il est équipé d'un pont en H permettant l'inversion de sens de rotation du moteur n°1 avec les bornes IN1, IN2 et IN3, IN4 pour le moteur n°2.

Des bornes EN1 et EN2 permettent de recevoir les informations de vitesse pour réaliser une variation de vitesse et les raccorder à des bornes PWM d'un Arduino.

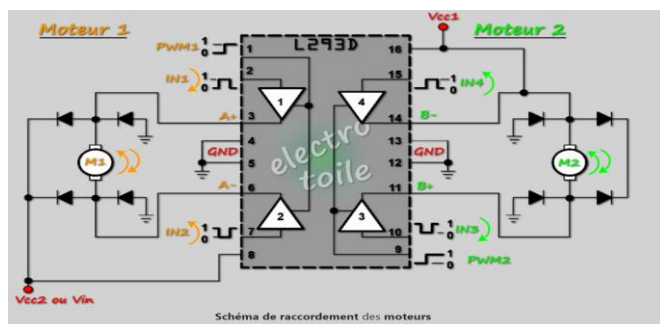


Figure 15 : Exemple de branchement de deux moteurs CC avec le module L293D

Le prix approximatif est de 4 euros.

Les deux moteurs Shield remplissent toutes les caractéristiques que nous avons besoin en termes de pilotages de moteurs et prix mais seul le Shield L298N permet de piloter des capteurs analogiques, c'est pourquoi le moteur Shield L298N semble plus adapté pour notre projet.

Matériel nécessaire (fois deux car deux robots)

-Une carte Arduino uno et son câble USB.

- Un Shield moteur L298N

https://fr.aliexpress.com/wholesale?catId=0&initiative_id=SB_20201203035037&SearchText=module+L298N

-Une pile 9V pour l'alimentation

https://fr.aliexpress.com/item/4000679954213.html?spm=a2g0o.productlist.0.0.3fa923afkx8Bq1&algo_pvid=7e76769c-7f4b-4104-a1fc-b070b4bf7d07&algo_expid=7e76769c-7f4b-4104-a1fc-b070b4bf7d07-12&btsid=2100bde116069962932801688e76a9&ws_ab_test=searchweb0_0,searchweb201602_,searchweb201603_

-Une breadboard

IV- Notre système du terrain façon arcade :

Nous allons avoir besoin de divers éléments pour réaliser notre terrain. Comme par exemple un système de comptabilisation de but. Cela passe par le biais de l'utilisation d'écran LCD pour afficher le score du match, LED colorés au moment d'un possible but, bruit sonore activé manuellement, contrôlé par un bouton sur notre application au moment d'un but.

L'objectif est de construire une petite maquette où nos robots vont pouvoir s'affronter dans un match arcade avec un système de comptage de point et d'arbitrage.

On utilisera pour ce système une carte Arduino UNO qui pilotera l'ensemble du système.

Un système de LED de couleurs peut être intéressant, en effet une LED verte pourrait s'allumer en cas de but, elle serait donc reliée à un capteur de distance pour voir si la balle a franchi la ligne de but. A chaque but, un bruit de but pourrait donner un côté plus sensoriel à notre match, si le but est validé cela activera un son sur l'application du téléphone (créer un bouton célébration sur l'application qui permet d'activer le son).

De plus un écran affichera le score en direct.

En cas de contact entre les voitures où une distance entre elles trop petite, une LED rouge pourrait s'allumer pour remettre la balle au centre et redémarrer l'action, on utilisera pour cela les capteurs de distance précédemment étudiés. Or on sait que ces capteurs fonctionnent en fonction d'une distance minimale. Il faudrait donc pouvoir gérer la distance entre les deux capteurs de deux robots et non la distance avec un éventuel obstacles (par exemple la balle). Car si on gère uniquement la distance entre un obstacle, le robot ne pourrait pas s'approcher de la balle. Ou si ce n'est pas une solution possible nous allons devoir surélever le capteur de distance pour qu'il ne capte pas les obstacles situés sur le sol et uniquement l'obstacle dit en hauteur, c'est-à-dire l'autre robot.

On peut utiliser un capteur de distance qui calcule la distance entre les poteaux de la cage si le ballon passe à travers, la distance va se réduire il y aura donc but.
Ce système est faisable avec les connaissances que l'on a acquise durant le premier semestre d'électronique.

La balle choisie pour le match sera étudiée dans la pratique pour savoir quelle taille exacte nous avons besoin, cependant une balle de babyfoot en polystyrène semble adaptée pour sa taille de rayon 1.5 cm et sa matière plutôt molle. Nous possédons déjà ce type de balle.

Pour la maquette nous allons choisir une planche de bois dont la taille va être déterminée dans la pratique selon la jouabilité. Mais il nous faudrait une planche on va utiliser une planche de bois afin de faire le sol de notre terrain arcade. On a choisi un bois de type medium car il allie résistance, prix, ergonomie et stabilité.

Matériel :

- Une carte Arduino UNO
- Minimum une LED rouge
- Plusieurs (2 minimums) LED verte
- Une balle de babyfoot en polystyrène
- Deux capteurs de distance à ultrason
- Un écran LCD
- Une planche de bois de cette taille : 50 cmx70 cm, (70cmx2cm) fois 2, (46.6cmx2cm) fois 2, (10cmx2cm) fois 4
- Petite cage de foot : (10cmx2cm) fois 4

On trouve le bois que nous avons besoin sur ce lien :

<https://www.leroymerlin.fr/v3/p/produits/panneau-medium-mdf-naturel-ep-18-mm-x-l-250-x-l-122-cm-e154566>

V-Forme du robot et ergonomie

Dans cette section nous allons voir les formes disponibles pour la réalisation de notre projet. Dans notre cas, nous savons que nous allons réaliser notre robot avec 3 roues (dont une roue folle). Il faut donc adapter nos recherches par rapport à ce critère. Nous éviterons de faire des recherches inutiles et nous nous concentrerons donc sur ce détail.

Pour le choix d'un châssis il faut le choisir tel qu'il soit le plus fin possible. En effet la longueur importe peu. Mais il doit être fin dans la largeur car le terrain sur lequel on compte le faire jouer, même s'il est de taille assez grande, ce n'est pas assez pour faire le choix de prendre de gros robots. En tout cas le choix d'un robot imposant n'est pas nécessaire et ne nous apportera que des aspects négatifs.

Pour autant, le choix d'un matériel aérodynamique ne nous est pas du tout bénéfique. Donc la largeur est la seule contrainte que l'on se fixe.

On sait que nous avons la possibilité de faire notre châssis nous même dans l'atelier, nous allons voir dans cette rubrique les types de châssis qu'il existe et décider de quelle manière nous voulons qu'il ressemble.

On sait qu'il existe d'autres matériau pour le choix du châssis mais nous en avons retenu que 3 qui étaient les plus probables et adéquat dans le cas de notre projet.

Nous prenons les types de châssis sur : <https://fr.aliexpress.com>

Exemple n°1 :

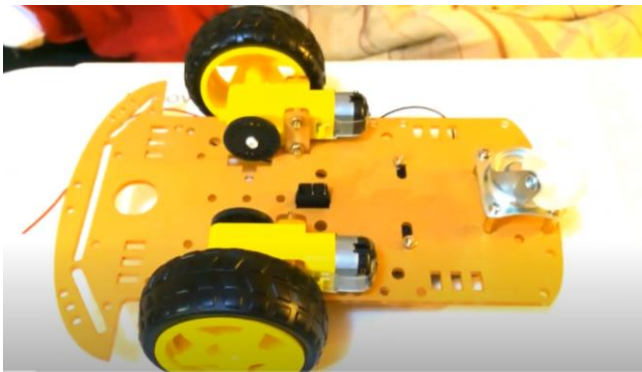


Figure 16 : Ce châssis est réalisé en PVC, sa forme est rectangulaire à l'avant et possède une forme ovoïde à l'arrière. Dans la longueur il suit une forme droite.

Dans un premier exemple :

- on remarque qu'il faut faire des petites entrées qui permettent de faire passer les différents fils pour relier à la carte.
- Deux espaces pour insérer les deux moteurs (jaunes dans ce cas-là) eux-mêmes reliés aux roues.
- Des entrées pour insérer la carte Arduino sur le dessus
- des entrées pour fixer la roue folle au châssis.

Exemple n°2 :



Figure 17 à 19 : Châssis réalisé en bois de forme ovoïde à l'avant et à l'arrière. Dans la longueur il suit une forme droite.

Le changement par rapport au châssis n°1 se trouve au niveau des roues : on remarque des sortes de rajout de matière permettant aux roues d'être un peu plus protégées par les contacts.

Le reste des entrées sont assez similaires.

Rajout de matière

Exemple n°3

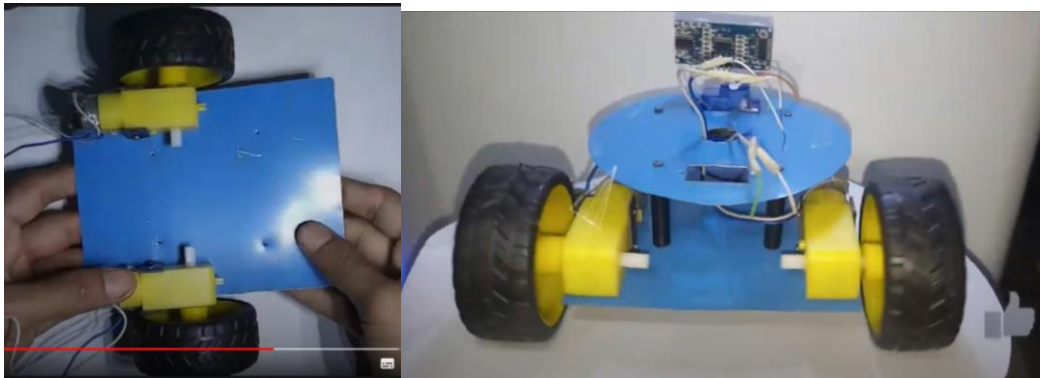


Figure 20 et 21 :
Châssis réalisé en tôle de métal bleu. La forme de l'étage du bas est carré tandis que la forme de l'étage du haut en ronde.

Dans ce dernier exemple : On remarque que ce châssis est réalisé manuellement donc le nombre d'entrée est fixé précisément pour relier les fils à la carte. De plus des entrées sont créées comme sur les exemples précédents afin d'insérer les moteurs.

On remarque une différence par rapport aux deux autres exemples, en effet on remarque que ce châssis est réalisé avec un étage permettant de mettre des outils supplémentaires ainsi de permettre aux éléments du robot d'être plus aérés.

Sur cet étage on remarque que le capteur de distance est fixé à l'aide de petites entrées de nouveau.

Conclusion :

Pour conclure sur le choix de la forme de notre robot, nous avons décidé de prendre les aspects qui nous semblaient les plus adéquats pour réaliser notre projet parmi ces exemples. En comparant ces différents exemples nous avons pensé que le meilleur moyen d'obtenir ce que l'on veut vraiment c'est de réaliser le châssis nous-même.

On pense donc que la forme du robot doit donc être rectangulaire avec les angles arrondis (pour limiter le risque que le robot ne se coince) dans la forme générale mais on pense essayer de faire une forme totalement arrondie sur l'arrière de la voiture. Sur l'avant de la voiture nous souhaitons faire une sorte de cuve pour pouvoir avoir le contrôle du ballon. Sur l'avant de la voiture, on ajoutera le système de tir. On pense aussi faire des rajouts afin de permettre aux roues d'être protégées davantage.

On sait que notre robot devra être équipé d'un capteur de distance ainsi qu'un système de tir (pour la balle). C'est donc pour cela que nous souhaitons réaliser le châssis de notre robot sur deux étages (comme sur l'exemple 3). Mais nous ne pensons pas que la forme ronde du 2ème étage soit adéquate. Nous disposerons donc le capteur de distance sur l'étage du haut et le système de tir au plus bas pour éviter la détection de ce dernier.

Pour ce qui est de la matière de notre robot, nous pensons que nous allons le réaliser en bois (contreplaqué sûrement 3 ou 4 mm). Car c'est la matière qui est allée au mieux l'aspect de la modularité et de la résistance pour notre robot. En effet si on va être amené à modifier le châssis du robot, cette matière est très modulable (collage, découpage ...).

VI- Connexion entre la carte Arduino UNO-Téléphone

<https://www.tutoriel-arduino.com/utiliser-hc-05-arduino/>

<https://arduino.blaisepascal.fr/bluetooth/>

<https://www.adafruit.com/product/1697>

<https://play.google.com/store/apps/details?id=eu.jahnestacado.arduinoorc&hl=fr&gl=US>

https://www.tubefr.com/cours-sur-mit-app-inventor-et-arduino_4.html#title

VI-1 - Modules de connexion

Les modules Bluetooth permettent à l'Arduino de communiquer avec un appareil Android. Ils sont différents les uns des autres et n'ont pas les mêmes compatibilités avec les applications Bluetooth ni aussi le même fonctionnement. Ils permettent un contrôle à distance de nos voitures.

On va se questionner sur le choix du module entre le HC05-06 et nRF8001.

Module HC-05 :

Il permet l'association (ou appairage) avec un autre périphérique et doit se faire directement depuis le module.

Ce module possède 6 sorties c'est pour cela qu'il peut être « maître » (il peut proposer à un autre élément Bluetooth de s'appairer avec lui) ou bien « esclave » (il ne peut que recevoir des demandes d'appairage).



Figure 22-23 : Photo du module HC-05

Pour connecter le module et le téléphone, il faut les mettre en lien à travers un programme. Il faut un code qui permet d'allumer la LED reliée à la sortie n°13 quand la carte Arduino reçoit le caractère « a » et de l'éteindre quand il reçoit le caractère « b » grâce au module Bluetooth qui communique « a » à son tour avec le périphérique.

```

char junk;
String inputString="";

void setup() {
  Serial.begin(9600);
  pinMode(13, OUTPUT);
}

void loop()
{
  if(Serial.available()){
    while(Serial.available())
    {
      char inChar = (char)Serial.read(); //Lire l'entrée
      inputString += inChar;           //Construit une chaîne de caractère à partir des caractères
    }
    Serial.println(inputString);
    while (Serial.available() > 0)
    { junk = Serial.read(); }
    if(inputString == "a"){           //Dans le cas où l'entrée est 'a' la LED s'allume
      digitalWrite(13, HIGH);
    }
    else if(inputString == "b"){      //Dans le cas où l'entrée est 'b' la LED s'éteint
      digitalWrite(13, LOW);
    }
    inputString = "";
  }
}

```

Figure 24 : Exemple de code

Le code est simple et assez efficace, il devra cependant être complété par la suite pour pouvoir diriger la voiture mais cela demeure une base solide pour une première connexion.

De plus, selon Ali Express (https://fr.aliexpress.com/price/bluetooth-hc05-module_price.html) le prix du module est d'environ 2 euro ce qui reste totalement abordable.

Module HC-06 :

Ce module possède 4 sorties c'est pourquoi il ne peut être qu'esclave car l'association (avec un périphérique) doit se faire depuis l'autre périphérique.

Pour le reste c'est le même fonctionnement que HC-05, le branchement va légèrement changer car il n'a que 4 sorties.



Figure 24 : Module HC-06

Lien pour acheter un module HC05 et un module HC06 :*

<https://fr.aliexpress.com/item/1005001621899428.html?>

Module nRF8001 :

Adafruit Bluefruit nRF8001 Breakout permet d'établir une liaison sans fil facile à utiliser entre votre Arduino et tout appareil compatible iOS ou Android (4.3+). Il fonctionne en simulant un appareil UART sous la surface, en envoyant des données ASCII entre les appareils, permettant de décider quelles données envoyer et quoi en faire à chaque extrémité de la connexion.

De plus, cette connexion n'utilise pas le Bluetooth classique mais le Bluetooth Low Energy, c'est pourquoi il est plus difficile à utiliser.

Il est compatible avec IOS, ce qui est assez important car IOS domine clairement le marché des smartphones.

Le prix est d'environ 20 Euros, ce qui reste très cher pour un module.

De plus, il est compatible seulement avec Bluetooth Low Energy ou Bluetooth 4.0. Ce qui réduit le choix d'application utilisable.

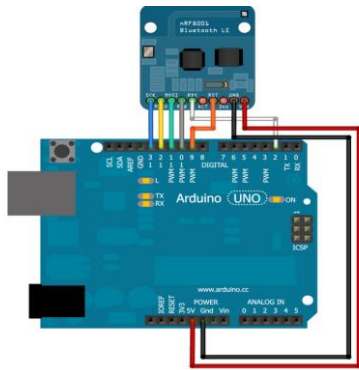


Figure 25 : Exemple de câblage module nRF8001 avec adafruit

Conclusion :

Le module HC-06 et HC-05 apportent plus de garanties en termes de prix et de qualité d'utilisation, car ils sont compatibles avec quasiment toutes les applications Bluetooth. Le choix entre les deux modules HC se fera dans la pratique et dans le choix des applications.

VII- Application Bluetooth

Il existe une multitude d'application qui permettent de faire le lien entre le module et le périphérique. On va s'intéresser à ces applications comme par exemple MIT app Inventor et Arduino Bluetooth Controller.

VII-1 - Arduino Bluetooth Controller

Cette application Android permet d'établir une connexion avec tout projet / microcontrôleur Arduino qui implique un module Bluetooth. Il permet à l'utilisateur de définir un UUID de son propre module Bluetooth pour connecter l'application Android avec ses projets. L'UUID de base est celui du module HC-06.

L'application possède un mode véhicule, qui correspond à nos besoins, ils fonctionnent de la façon suivante :

En appuyant sur le bouton "joystick", l'utilisateur peut contrôler à distance un véhicule avec des gestes spécifiques. Pour l'interprétation des gestes de commandes, l'application utilise le capteur accéléromètre intégré de l'appareil intelligent. Il y a 8 gestes différents disponibles (avant, arrière, gauche, droite, avant gauche, avant droite, arrière droite, arrière gauche). Il y a également un bouton arrêt qui permet d'arrêter la voiture.

En outre, les options de menu, fournissent un bouton de changement de sensibilité à modifier le point où l'application attrape les événements de mouvement pour ce mode. Comme le titre l'indique, ce mode est spécialement conçu pour contrôler à distance un véhicule qui utilise un module Bluetooth et un microcontrôleur.



Figure 26 : *Icone de l'application sur Play Store de Arduino Bluetooth Controller*

VII-2 - MIT app Inventor

MIT App Inventor permet créer facilement des interfaces graphique Android pour contrôler des applications robotiques comme une voiture ou un bras.



Figure 27 : *Visualisation directement sur Smartphone*

En téléchargeant l'application "MIT App Companion", il est possible de synchroniser l'éditeur visuel de MIT App Inventor et son smartphone.

Cela permet de constater immédiatement le résultat du design sur votre smartphone.

Il y a 4 gestes différents disponibles (avant, arrière, gauche, droite).

Conclusion :

Pour conclure les deux applications sont semblables, les deux permettent un contrôle de nos voitures de qualité, cependant MIT semble avoir quelques spécificités en plus notamment en termes d'affichage et de pratique.

C'est pourquoi le choix de MIT App Inventor semble plus cohérent mais le choix se réalisera lorsque que l'on testera réellement les deux applications. Mais si un choix doit être réalisé dans l'immédiat, on se penchera dans un premier sur l'application MIT Inventor car elle semble plus facilement utilisable par rapport à Arduino Bluetooth Controller. Si l'application nous convient nous resterons sur notre premier choix. Dans le cas contraire nous changerons d'application.

VIII – Etat de l'art : projet déjà existant ressemblant

<https://thestempedia.com/project/soccer-playing-mobile-robot>









Introduction

Ce projet utilise comme nous plusieurs robots qui s'affrontent lors d'un match de foot. Le projet présente deux styles de robots : un gardien et un joueur. Les deux styles de joueurs sont associés à deux styles de robots différentes (un robot pouvant tirer et un autre pouvant renvoyer les balles).

Montage et éléments du corps du robot

Le montage du corps du robot, il s'agit de fixer plusieurs éléments (roues, moteurs, élément de tir/arrêt, carte Arduino ...) à l'aide de vis, boulons et supports fixé sur le châssis.

Voici le matériel nécessaire dans la réalisation du corps de ce type de robot :

	Partie acrylique: plaque de base	1	Kit de démarrage
	Support moteur BO	2	Kit de démarrage
	Boulons M3 de 8 mm de longueur	sept	Kit de démarrage
	évoquer	4	évoquer
	évoquer	5	évoquer
	Écrous M3	8	Kit de démarrage
	Roue BO	2	Kit de démarrage
	Roue de roulette	1	Kit de démarrage

On nous présente l'ensemble des éléments qui sont utilisés dans le cas de leur projet. Il s'agit d'une liste exhaustive et nous permet d'avoir une idée sur l'ensemble du matériel qu'il faut choisir et utiliser pour réaliser notre robot.

Le matériel simpliste qui ne nécessite pas de connaissances précises, le montage se fait à partir de la partie acrylique qui représente le châssis sur lequel on va fixer les éléments du robot.

Figure 28 : matériel utilisé dans le cadre de ce projet

Ajouts incorporés sur les robots :



Figure 29 : Il s'agit d'un ajout permettant de customiser le robot en goal. Pour cela il y a l'ajout d'un support composé d'élastique pour faire rebondir la balle si jamais cette dernière arrive dessus. Cet ajout du robot peut être un supplément dans le projet principal



Figure 30 : Il s'agit d'un ajout permettant de customiser le robot en joueur. Pour cela il y a l'ajout d'un support composé de Brix Bricks (similaire au lego) ainsi qu'une tête asservissement fixé à l'aide de colle. Cet ajout du robot peut être un élément dans le projet principal utile à son fonctionnement.

Circuit du robot :

Comme on a pu voir dans le matériel nécessaire, ce projet incorpore un boîtier evive qui permet au robot de se connecter en Bluetooth avec un téléphone portable Android.

Il s'agit d'une alternative puisque ce boîtier ne contient pas de fil, tous les éléments Arduino ne sont donc pas nécessaires puisque déjà inclus dans le boîtier.

Evive est utilisé couplée avec Arduino méga pour développer les compétences de programmation en robotique. Ce boîtier est livré avec un micrologiciel qui fournit préalablement une gamme de fonctionnalités telles que le contrôle du moteur, commande.

IX-Système de tir

Nous allons étudier les différents choix possibles pour faire avancer notre balle. Même si ce système n'est pas le plus important de notre projet car le robot seul muni d'aucun système pourrait faire avancer la balle, nous avons étudié différents systèmes qui pourraient s'implémenter sur notre robot.



Figure 30

Il y a l'ajout d'un support composé de Brix Bricks (similaire au lego) ainsi qu'une tête asservissement fixé à l'aide de colle.

Ce fonctionnement pourrait être intéressant car il peut se rétracter et seulement avoir la partie planche qui permettrait un contact avec la balle en cas d'un contact avec vitesse. Si le contact est sans vitesse alors le petit bras pourrait s'actionner et redonner de la vitesse à la balle. Il fonctionnerait avec un servo moteur qui régulerait l'angulation de la batte piloter par la carte Arduino UNO, le servo moteur serait insérer dans le châssis.

Ce système paraît donc en adéquation avec nos problématiques car le système n'est pas trop sophistiqué et laisse une place nécessaire à un capteur de distance à ultrason positionné au-dessus du système.

X- Conclusion finale

Grâce à nos diverses recherches nous avons réussi à trouver l'ensemble des réponses de nos axes de recherche. Ces choix sont réalisés certaines fois par théorie où nous avons réussis à faire une conclusion précise comme pour les le choix des roues ou encore le choix du Shield moteur où nous avons un composant bien précis.

Certaines fois ce choix devra se faire par le biais de l'expérimentation. Nous avons réussi à confronter nos idées mais seul la pratique saura vraiment répondre à nos demandes comme pour le choix des moteurs par exemple.

Pour conclure, ce projet nous tiens particulièrement à cœur car c'est un projet ambitieux qui va nous permettre de nous affirmer dans le domaine de l'électronique tout en s'intéressant à notre sport préféré : le football.

On propose ci-dessous l'ensemble du matériel que nous allons avoir besoin pour réaliser notre projet :

Moteur de réducteur de vitesse axe simple DC 3-6V (fois 4) : Moteur TT 130, 2 roues 65mm et moteur DC3V-6V sur :

https://fr.aliexpress.com/item/32898809340.html?spm=a2g0o.productlist.0.0.4ddf4777ihMe3n&algo_pvid=c59b2b75-5358-4691-89e5-81ef4d8a6134&algo_expid=c59b2b75-5358-4691-89e5-81ef4d8a6134-44&btsid=2100bdd816067299824954329ebc46&ws_ab_test=searchweb0_0,searchweb201602_,searchweb201603_

Servo moteur pour capteur de distance (fois 4) : SG90 servo sur :

https://www.banggood.com/SG90-Mini-Gear-Micro-Servo-9g-For-RC-Airplane-Helicopter-p-1009914.html?abprots=0&p=DQ30066511122014069J&utm_campaign=educ8stv&utm_content=3216&akmClientCountry=FR&cur_warehouse=CN

Un Shield moteur L298N

https://fr.aliexpress.com/wholesale?catId=0&initiative_id=SB_20201203035037&SearchText=module+L298N

Une pile 9V pour l'alimentation

https://fr.aliexpress.com/item/4000679954213.html?spm=a2g0o.productlist.0.0.3fa923afkx8Bq1&algo_pvid=7e76769c-7f4b-4104-a1fc-b070b4bf7d07&algo_expid=7e76769c-7f4b-4104-a1fc-b070b4bf7d07-12&btsid=2100bde116069962932801688e76a9&ws_ab_test=searchweb0_0,searchweb201602_,searchweb201603_

Lien pour acheter un module HC05 et un module HC06 :

<https://fr.aliexpress.com/item/1005001621899428.html?>

On trouve le bois que nous avons besoin sur ce lien :

<https://www.leroymerlin.fr/v3/p/produits/panneau-medium-mdf-naturel-ep-18-mm-x-l-250-x-l-122-cm-e154566>

Kit pour certains aspects :

https://www.banggood.com/Geekcreit-DIY-L298N-2WD-Ultrasonic-Smart-Tracking-Moteur-Robot-Car-Kit-for-Arduino-products-that-work-with-official-Arduino-boards-p-1155139.html?abprots=0&utm_campaign=7858775_1155139&utm_content=1087&p=CS120478587752016125&akmClientCountry=FR&cur_warehouse=CZ

C'est un kit de construction de voiture qui va nous permettre d'assembler les différents composants de notre robot, il est constitué de composants dont on a besoin tels que :

Le module L298N

Le servo moteur 9G

2 Motoréducteurs jaunes

1 capteur à ultrason

1 carte Arduino

Toute la visserie utile à attacher nos composants sur le châssis.