

Projet Bachelor A2 : Groupe numéro 2

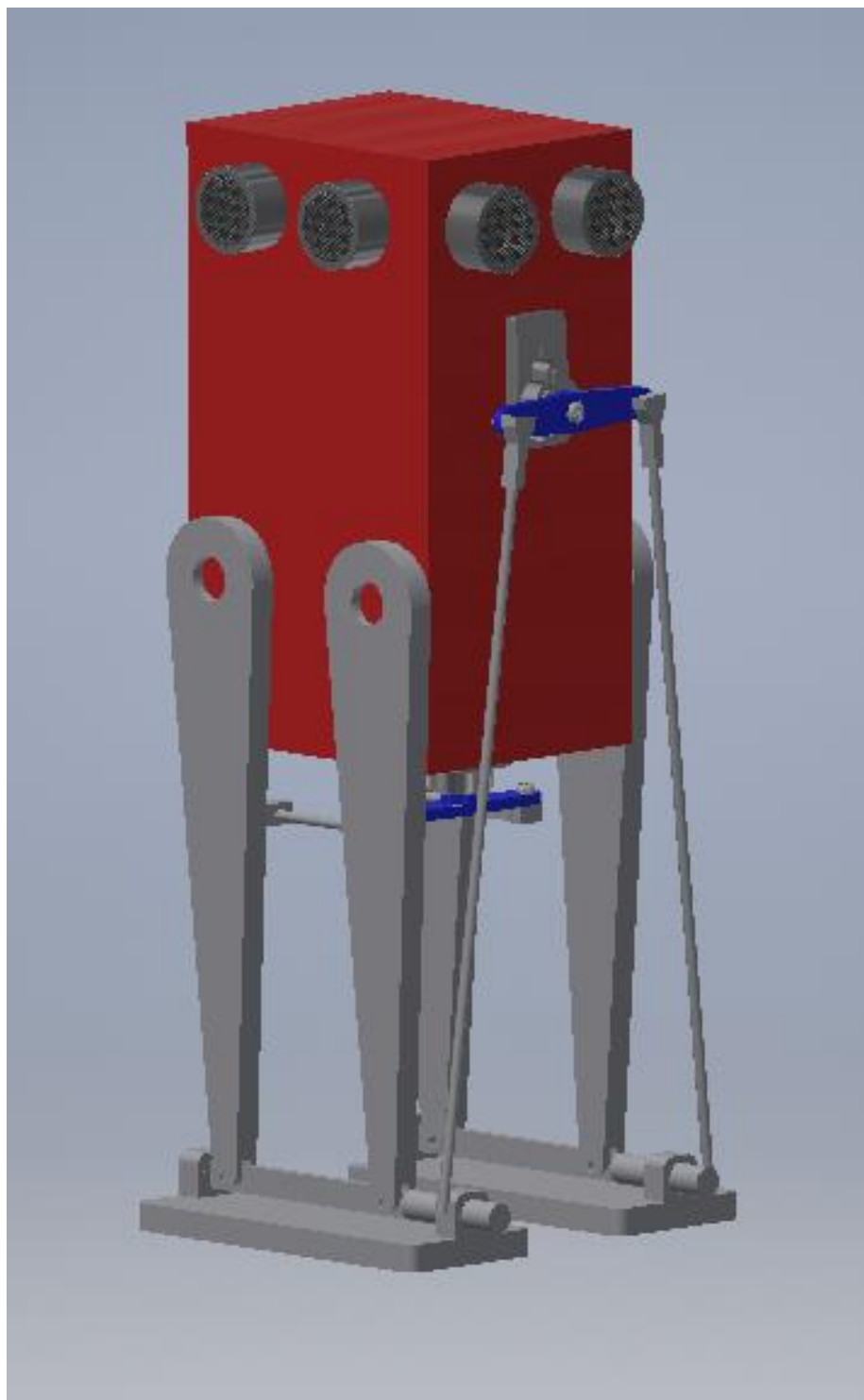


Table des matières

1/ Introduction :	3
Nos excuses :	3
Le projet :	3
Le groupe :	3
2/ Partie Mécanique	4
A) Les composants électroniques :	4
<i>La carte Arduino Uno :</i>	<i>4</i>
<i>Les servo-moteurs :</i>	<i>5</i>
<i>Les capteurs Ultrasons :</i>	<i>6</i>
<i>Gyroscope :</i>	<i>6</i>
<i>La pile 9Volts :</i>	<i>7</i>
B) Les composants structurels :	8
• Le châssis (corps principal) :	9
• Les Jambes :	9
• Les Pieds :	10
3/ Partie Electronique	11
4/ Partie Informatique	13
Les fonctions principales :	13
L'algorithme pour le labyrinthe :	16
LE CODE ARDUINO :	17
5/ Assemblage	17
Assemblage mécanique :	18
Assemblage électrique :	19
6/ Simulation	20
Simulation mécanique :	20
Simulation informatique :	21
7/ Conclusion	22

1/ Introduction :

Nos excuses :

Tout d'abord, avant de vous parler de notre projet, nous voulions nous excuser par rapport au retard de ce document qui aurait dû vous être livré plus tôt. Nous avons pris en compte ce délai supplémentaire et nous avons décidé d'essayer de rendre un rapport complet. Nous espérons que celui-ci sera à la hauteur de vos espérances.

Le projet :

Notre projet consiste à la réalisation d'un Robot autonome en faisant appel à nos différentes compétences en mécanique, informatique et électronique. Notre robot doit répondre à différentes contraintes tels que :

- Doit être bipède
- Pouvoir sortir d'un labyrinthe
- Doit être complètement autonome
- Avoir 2 « bras »

Malheureusement à cause de l'épidémie de coronavirus, nous n'avons pas pu réaliser concrètement le projet mais néanmoins nous avons fait en sorte qu'en lisant le projet, le fonctionnement du robot soit facilement compréhensible. Pour réaliser ce projet nous avons scindé notre équipe en plusieurs sous parties.

A) Le groupe :

Notre groupe est composé de 5 membres :

- Charles TEGONI
- Calvin PIERRE-JOSEPH
- Théo FAU
- Quentin RIMBAUT
- Ali YESILKAYA

Pour être le plus optimal possible à la réalisation du projet, nous avons décidé d'attribuer des tâches particulières pour chaque membre d'équipe en fonction de leurs affinités avec le domaine.

Nous avons décidé que Quentin RIMBAUT et Ali YESILKAYA s'occuperont de la partie mécanique du projet. Ils ont dû adapter la partie mécanique du Robot au cahier des charges. Pour cela ils ont dû établir une solution permettant que notre robot puisse se déplacer dans tous les sens possibles mais aussi en permettant une marche arrière. En plus du cahier des charges ils avaient comme contrainte le fait que le robot devait être le plus simple à produire pour réduire au mieux les couts.

La partie électronique a été attribuée à Charles TEGONI. Il doit s'occuper d'analyser les différents éléments électroniques tels que les Servomoteurs et autres, et d'établir un système électrique capable de répondre au cahier des charges et donc à notre configuration mécanique.

Puis viens la partie informatique avec Théo FAU et Calvin PIERRE JOSEPH. Ils sont les principaux développeurs et ont programmer les différentes fonctions permettant à notre robot d'avancer, de reculer, de tourner, etc...

Ces trois sous-groupes ont dû fonctionner et réfléchir en collaboration pour que notre avancement puisse répondre à une logique globale, mais aussi répondre à une cohésion afin qu'aucune partie ne soit en retrait.

2/ Partie Mécanique

Dans cette partie Mécanique, nous allons décrire les différents éléments que nous utiliserons, puis ça sera dans la partie « Assemblage » où nous réaliseront l'assemblage de ses différentes parties. Nous avons divisé notre partie mécanique en deux sous parties. Nous avons donc une partie où nous décrivons les composants électroniques puis une partie où nous décrivons nos composants mécaniques avec toute la structure du robot. De plus, nous avons décidé d'ignorer quelques contraintes afin d'augmenter au maximum la fiabilité du projet. Nous avons donc décidé d'ignorer les "bras" du projet pour l'instant. Néanmoins, nous souhaitons l'ajouter plus tard. Tous les composants ont été modélisés sur Autocad/SolidWorks. Nous commencerons donc par détailler le matériel électronique dont nous disposons.

A) Les composants électroniques :

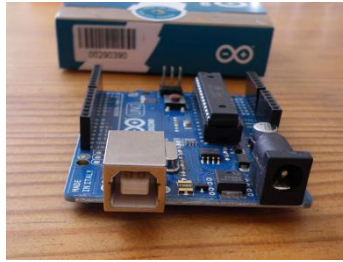
Les composants électroniques ont été choisis minutieusement afin de répondre adéquatement à nos différentes contraintes. Nous avons précédemment fait un rapport sur les différents composants en répondant sur le pourquoi nous utilisons ces composants. Nous allons donc voir maintenant les caractéristiques mécaniques et leurs intégrations dans notre robot.

La carte Arduino Uno :

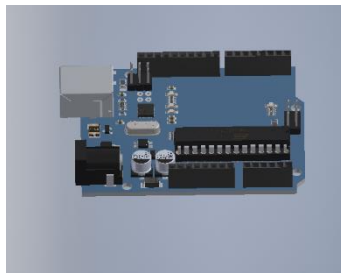
Nous avons une carte Arduino Uno qui fait environ 7.5x5x1.5cm. C'est le cerveau de notre robot ou tous nos programmes sont stockés. De plus, tous les câblages se réunissent dessus ; nous avons donc dû penser plus minutieusement à sa place dans notre robot. Nous le

modéliserons sur Autocad/SolidWorks avec un simple rectangle. La carte n'est pas encombrante et peut être fixée simplement grâce à des vis situées aux extrémités.

Image réelle :



Modélisation :



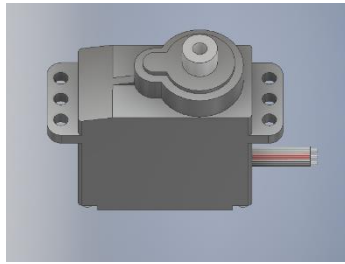
Les servo-moteurs :

Les servomoteurs sont notre principale source de mouvement. Nous avons pour nos servomoteurs des HS-422 et sont très performants. Ils ont une taille de 4x3x1.7 cm. Ce sont des servomoteurs classiques mais avec un défaut particulier, ils sont encombrants dans le sens où, la rotation ne peut se faire que d'un côté particulier et que le reste du corps prend de la place. Néanmoins, la fixation reste simple avec de simples vis situées sur la face des servomoteurs.

Image réelle :



Modélisation :



Les capteurs Ultrasons :

Notre première source d'information pour évaluer la distance d'un objet grâce aux ultrasons. Les capteurs ultrasons que nous utilisons sont des HC-sr04 qui sont très petites et peuvent être facilement fixés grâce à 2 vis situés sur les deux extrémités opposées avec une taille de 4.5x2x1.4 cm.

Image réelle :



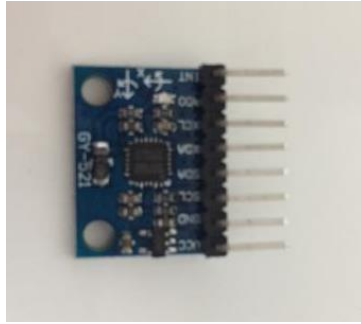
Modélisation :



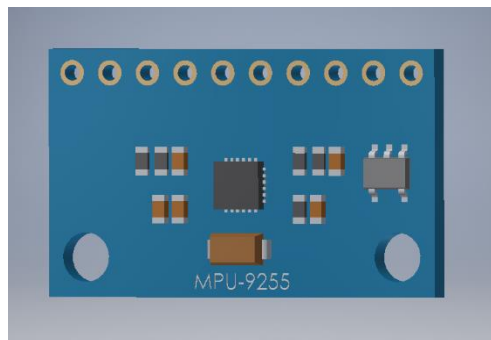
Gyroscope :

Le gyroscope est le second capteur que nous utilisons afin de recevoir le maximum d'informations possible avec notamment l'accélération de notre robot. Il a une taille très petite 2x2x0.5 cm. La fixation reste aussi assez simple avec des vis.

Image réelle :



Modélisation :



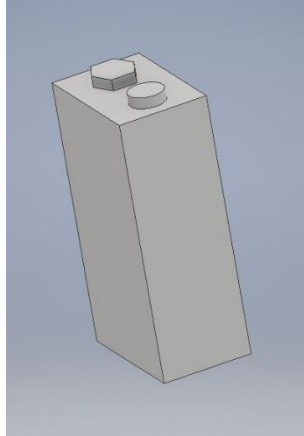
La pile 9Volts :

Notre unique source d'énergie est une pile de 9Volts qui alimentera à lui toute seule tous les composants électroniques de notre projet. Il est préférable d'utiliser cette source d'énergie nous le verrons dans la partie électronique. Il est très petit avec une taille de 2x4x1 cm et nous le fixons avec de la colle.

Image réelle :



Modélisation :



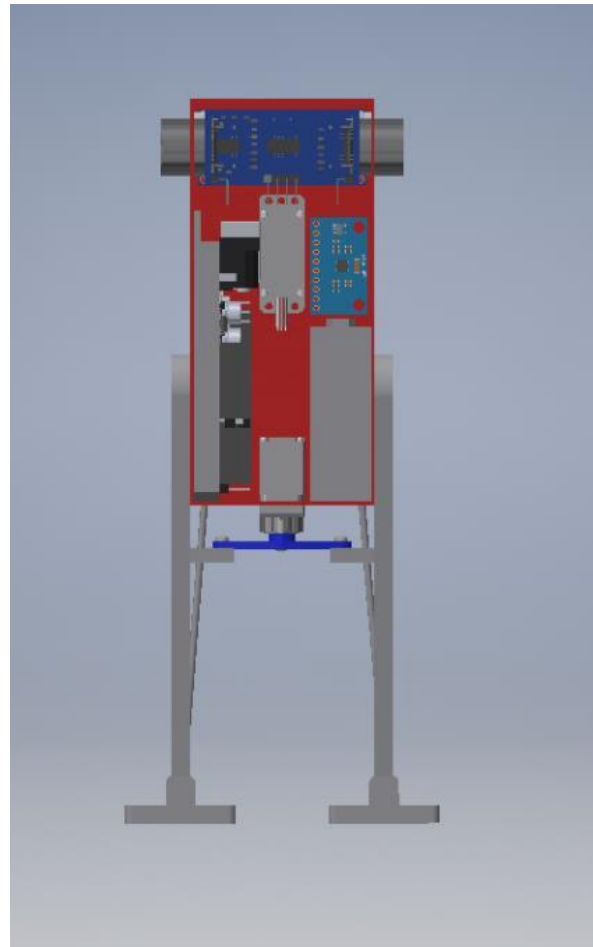
Nous pensons que pour éviter toutes erreurs possibles du a une mauvaise gestion de la taille ; définir et modéliser les différents composants pourra nous aider à éviter des erreurs classiques. Après avoir défini nos principaux composants électroniques et surtout leurs tailles. Nous passons donc à la description du squelette de notre robot ou tous les différents composants seront intégrer dedans et tout en expliquant chaque partie.

B) Les composants structurels :

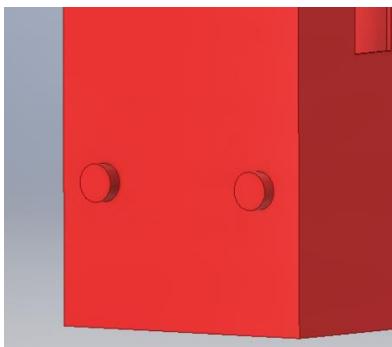
Notre robot est donc composé d'un squelette particulier. Encore une fois, nous avons modélisé toutes les parties sur Autocad / SolidWorks. Notre robot se décompose en 3 parties structurelles. Ces parties ont été pensées pour pouvoir être assemblé et répondre aux déplacements demandés. Nous avons donc comme parties ; le châssis, les jambes et enfin les pieds. Toutes les parties structurelles sont censées être imprimés en 3D grâce à une imprimante 3D.

- Le châssis (corps principal) :

Voici le corps de notre robot qui contiendra tous les organes principaux nécessaires à son bon fonctionnement. Nous sommes parties sur une base de rectangle permettant de contenir tous les composants électroniques ainsi que l'alimentation. Nous avions pour objectif une liberté de mouvement totale. Nous voulions une vraie autonomie pour notre robot afin qu'aucun câble ou autre puisse gêner les déplacements du robot. Nous voulions aussi réduire au maximum le poids du robot, pour cela nous avons optimiser au maximum la taille de chaque partie mécanique dans le but de miniaturiser le robot le plus possible. La principale contrainte était la taille de la carte Arduino avec 10 cm de hauteur, 5cm de largeur et pour une profondeur de 5cm qui prenait une place conséquente. En effet le boîtier devait pouvoir accueillir tous les autres composants tel que les servomoteurs, capteurs, alimentation et câbles tout en étant le plus minimaliste possible. Vous pouvez donc apercevoir l'organisation choisie pour accueillir les composants électroniques et ainsi que les points d'accroches situés en bas du châssis pour s'accorder aux jambes. Les composants électroniques seront fixés simplement avec des vis ou collé directement sur le châssis. Le châssis portera donc la carte Arduino, les servomoteurs, la pile, le gyroscope ainsi que les 3 capteurs ultrasons.



- Les Jambes :



Les jambes sont liées directement aux boîtiers grâce à la hanche et à la partie des pieds grâce à la cheville. Nous avons donc une première partie avec les jambes qui sont simplement 4 barres hautes, 2 de chaque cotées, qui relie le châssis et les pieds du robot. La liaison au niveau du châssis reste assez simple avec des vis et boulon car nous avons besoin que les jambes puissent profiter une rotation permettant le mouvement. La liaison entre les jambes et les pieds reste aussi relativement simple et respecte une certaine liberté en rotation.

- Les Pieds :

Les pieds se décomposent en deux parties. Une partie dites cheville qui relie la jambe aux pieds et une autre qui relie la cheville au pied. La partie du pied repose sur le sol et est reliée au deuxième servomoteur. Cette liaison permet l'élévation verticale de la partie basse du corps. La liaison entre le pied et le servomoteur se fait directement par une tige. La tige est liée au servomoteurs grâce à un support et des visse.

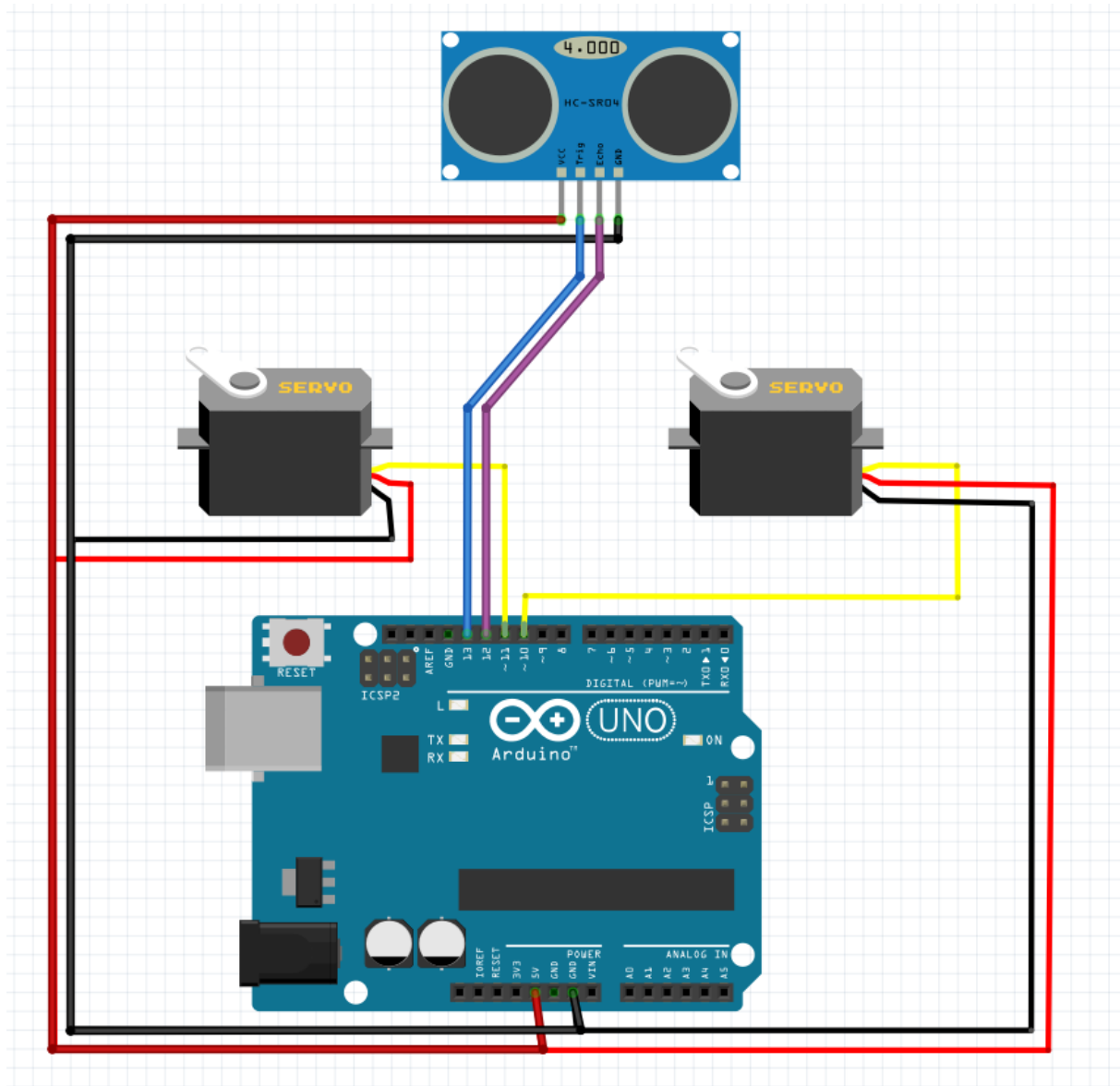
Voici donc comment nous voyons les différentes parties de notre robot pour une réalisation concrète et simple. Vous pouvez donc voir que pour effectuer des mouvements horizontaux et verticaux, nous n'utilisons que 2 servomoteurs. La réalisation du châssis, jambes, pieds, tiges et les différentes liaisons étaient toutes prévues avec une imprimante 3D. Toutes ces réalisations mécaniques ont été pensées avec toutes les contraintes électroniques.



3/ Partie Electronique

Dans cette partie, nous allons vous présenter les différents circuits électroniques que nous avons mis au point pour le fonctionnement du robot et leurs évolutions que nous avons modélisées sur Fritzing.

Tout d'abord voici le premier montage que nous avons mis au point :

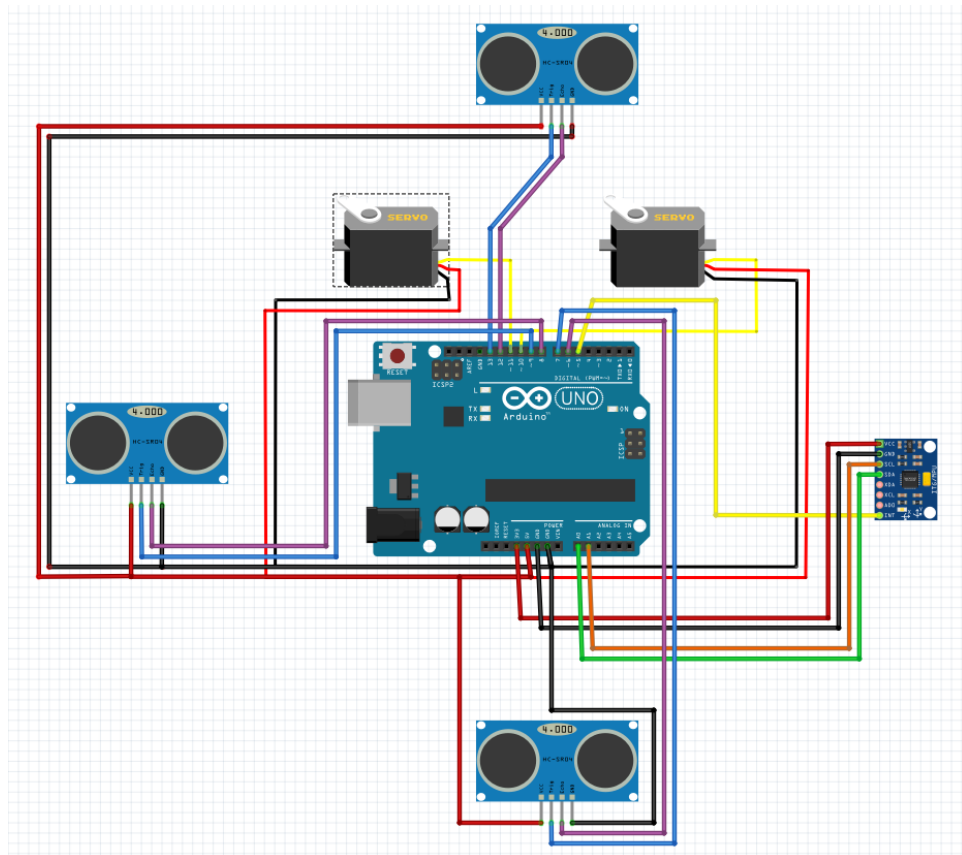


Le montage modélisé ci-dessus correspondait aux premiers plans du robot qui comportait une tête rotative sur 180° (ceci explique la présence d'un unique capteur à ultrasons, la tête était utilisée pour orienter le capteur sur l'avant et les côtés du robot dans le but de détecter les murs et déterminer si le robot continue sur son chemin ou bien s'il devait tourner)

Ce montage était aussi composé de deux servomoteurs qui actionnaient les mouvements de la tête et des jambes de l'appareil, seulement nous nous sommes rendu compte que l'idée de la tête

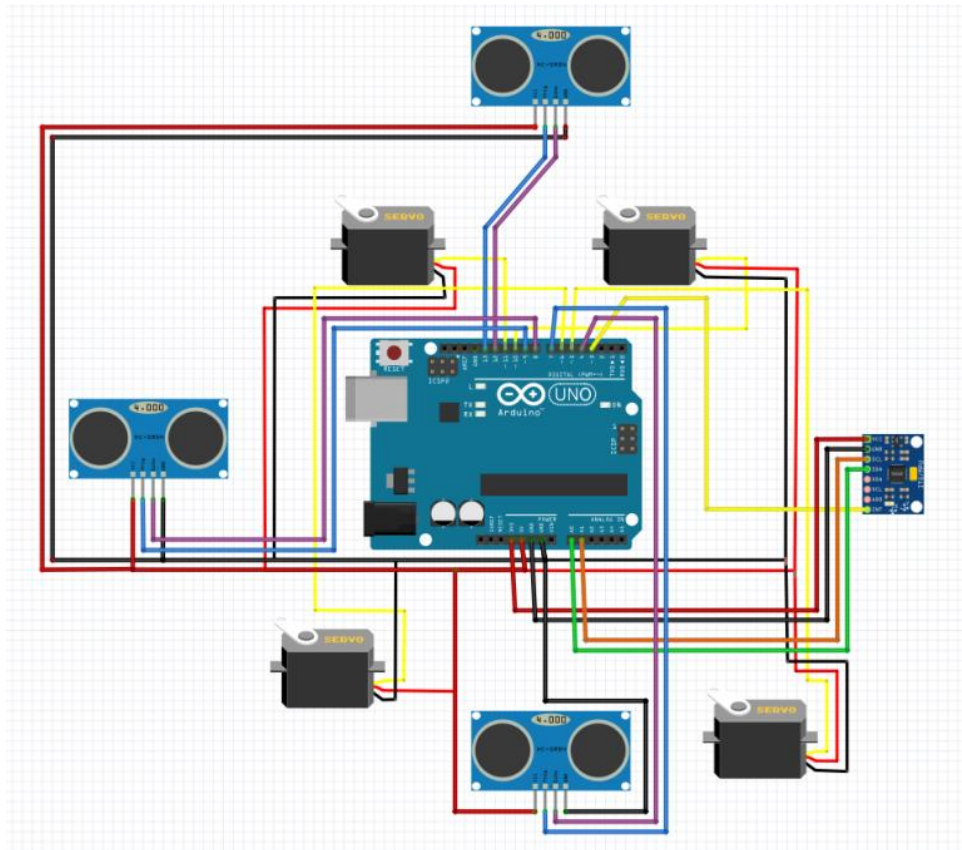
rotative n'était pas très fiable en pratique et que le robot manquait de stabilité lors de la marche, c'est pourquoi nous avons ajouté des composants et pris une direction différente vis-à-vis du système de repérage dans l'espace du robot. Ici tous les composants sont prévus pour être utilisés sur un VCC = 5V, c'est pour cela qu'ils sont branchés sur le même circuit et qu'aucune résistance n'est utilisée.

Voici le second montage réalisé pour le projet :



Les améliorations par rapport aux anciens plans consistent en l'ajout d'un accéléromètre branché sur le circuit 3.3V et sur une masse différente conformément à la documentation du composant et pour donner plus de clarté au montage qui commence à être chargé en composants et en câbles car l'utilisation d'une volumineuse breadboard est contre-productif vis-à-vis du respect des dimensions imposées dans le cahier des charges. Ce dernier possède aussi un gyroscope intégré pour assurer une meilleure réponse face aux instabilités potentielles lors de la marche de l'appareil et l'intégration de deux capteurs à ultrasons supplémentaires afin d'avoir un retour constant sur les objets/obstacles se situant devant et sur les côtés du robot.

Nous allons à présent conclure la partie électronique avec le 3e prototype de montage que nous avons réalisé :



Ce prototype final possède deux servomoteurs supplémentaires, dans le but de faire bouger les bras du robot à la fin de l'épreuve. Nous avons choisi d'ajouter des moteurs au circuit 5V parcequ'au premier abord, il semblait que c'était la méthode la plus fiable pour bouger les bras, cependant un détail nous as échappé : la carte Arduino UNO manque de puissance pour faire tourner autant de composants sans shield ou sans une plus grosse alimentation externe, et donc cela rajoute de l'encombrement dans le corps exigü du robot, et malheureusement nous n'avons pas la place à pour installer un tel système, c'est pourquoi nous avons choisi d'adapter mécaniquement le montage numéro deux pour faire lever les bras à l'appareil à la fin de la course.

4/ Partie Informatique

Dans cette partie, nous allons détailler les différents codes que nous allons utiliser pour la programmation de notre robot. Pour pouvoir être claire dans les explications des codes, nous allons les détailler pour chaque fonction puis dans la partie simulation nous essayerons d'adapter les codes en fonction de l'assemblage final. Donc dans cette partie, les valeurs numériques ont été choisies arbitrairement pour qu'au moins la fonction marche seule.

Les fonctions principales :

Nous commencerons par détailler la fonction pour avancer. Cette fonction fait appelle à plusieurs autres fonctions que nous avons dû créer. Dans l'idée, nous avons donc 2 jambes avec 2

servomoteurs. Comme détaillés dans la partie mécanique, pour avancer, nous séparons l'action en 2 parties composés de 3 fonctions :

La fonction principale :

```
void Avance() {  
    delay(t_long);  
    Avance_droite();  
    Avance_gauche();  
    delay(t_long);  
}
```

Les 2 fonctions pour faire avancer les 2 jambes :

Nous avons donc 2 fonctions qui sont :

```
void Avance_gauche() {  
    Leve_jambe_gauche();  
    Avance_jambe_gauche();  
    Pose_jambe_gauche();  
}  
  
void Avance_droite() {  
    Leve_jambe_droite();  
    Avance_jambe_droite();  
    Pose_jambe_droite();  
}
```

La première partie, la jambe droite.

- La première pour lever la jambe droite avec le servo1 :

```
void Leve_jambe_droite() {  
    for (pos = 90; pos <= 180; pos += 1) {  
        servo_1.write(pos);  
        delay(15);  
    }  
    delay(t_long);  
}
```

- La seconde pour avancer la jambe droite avec le servo2 :

```
void Avance_jambe_droite() {  
    for (pos = 90; pos <= 135; pos += 1) {  
        servo_2.write(pos);  
        delay(15);  
    }  
    delay(t_long);  
}
```

- La troisième pour poser la jambe droite avec le servo1 :

```
void Pose_jambe_droite(){
    for (pos = 180; pos >= 90; pos -= 1) {
        servo_1.write(pos);
        delay(15);
    }
    delay(t_long);
}
```

Voilà pour le mouvement de la première jambe

La seconde partie pour la jambe gauche qui est exactement la même que pour la première partie avec :

- La première pour lever la jambe gauche avec le servo1 :

```
void Leve_jambe_gauche(){
    for (pos = 90; pos >= 0; pos -= 1) {
        servo_1.write(pos);
        delay(15);
    }
    delay(t_long);
}
```

- La seconde pour avancer la jambe gauche avec le servo2 :

```
void Avance_jambe_gauche(){
    for (pos = 90; pos >= 45; pos -= 1) {
        servo_2.write(pos);
        delay(15);
    }
    delay(t_long);
}
```

- La troisième pour poser la jambe gauche avec le servo1 :

```
void Pose_jambe_gauche(){
    for (pos = 0; pos <= 90; pos += 1) {
        servo_1.write(pos);
        delay(15);
    }
    delay(t_long);
}
```

Nous avons donc un premier prototype pour avancer.

- La fonction pour tourner à gauche et à droite :

```
void tourner_a_droite(){
    delay(2000);
    Avance_gauche();
    delay(2000);
    Avance_jambe_droite();
    Avance_gauche();
    delay(2000);
    Avance_jambe_droite();
}
```

```

void tourner_a_gauche() {
    Avance_droite();
    Avance_jambe_gauche();
    Avance_droite();
    Avance_jambe_gauche();
}

```

- Et une dernière fonction pour l'arrêt :

```

void arret() {
    servo_1.write(90);
    servo_2.write(90);
}

```

Comme vous pouvez le voir, nous avons des délais et des positions de départ particulier. Nous avons obtenu ces valeurs grâce à une première simulation. Ces valeurs sont très approximatives car nous n'avons pas pu tester notre programme sur un prototype réelle qui aurait sans doute donné des résultats et valeurs plus précis.

➤ L'algorithme pour le labyrinthe :

Pour sortir du labyrinthe nous avons choisi d'utiliser l'algorithme de Pledge.

-Pourquoi ?

Très facile d'usage et de compréhension, grande communauté.

-Ce qu'il fait :

Avancer en ligne droite en comptant les changements de direction. En supposant que tous les angles soient droits, on a alors deux possibilités :

- Tourner à droite
- Tourner à gauche

On compte alors les changements de direction en augmentant d'un point lorsqu'on tourne à gauche et en diminuant d'un point lorsqu'on tourne à droite.

Le compteur est initialisé à 0, on répète alors le schéma suivant :

1 : Aller tout droit jusqu'au mur puis passer à l'instruction 2.

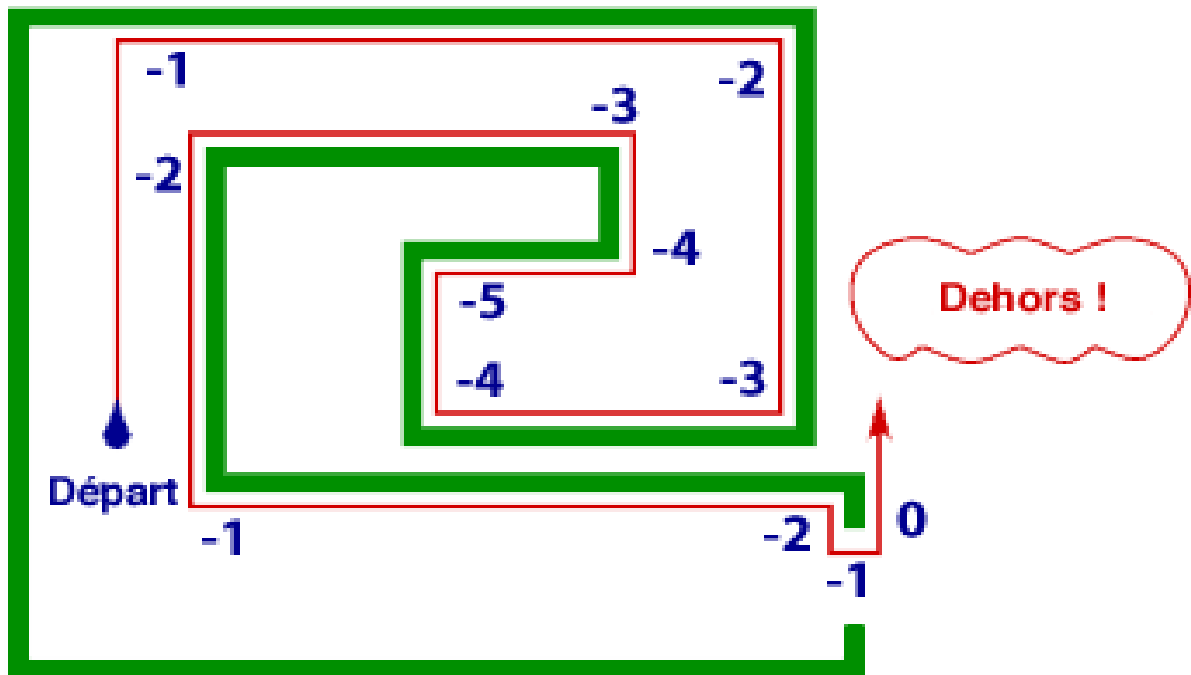
2 : Longer le mur par la droite jusqu' à ce que le décompte des changements de direction atteigne zéro puis passer à l'instruction 1.

LE CODE ARDUINO :

Pour visualiser le code, nous vous l'avons joint avec le rapport car le code seul prend plusieurs pages.

Petits commentaires en plus.

Pour sortir du labyrinthe, on appelle notre fonction `sortir_labyrinthe()` pour que notre robot suive les instructions 1 et 2 détaillées ci-dessus.



Nous avons choisi cet algorithme pour que la vitesse de notre robot soit la plus rapide possible. Malheureusement, cette méthode ne fonctionne pas à tous les coups. En effet, lorsque le robot est confronté à une configuration dite « îlot », c'est-à-dire lorsque le chemin que doit emprunter le robot nécessite qu'il se détache du mur, le robot se voit dans l'incapacité de trouver la sortie.

5/ Assemblage

Comme nous ne pouvons pas réaliser concrètement notre projet car nous n'avons plus accès aux imprimantes 3D du FabLab pour créer les pièces nécessaires, nous ne pouvons parler que de l'aspect théorique. Néanmoins, nous avons tout de même essayé avec des objets de récupération mais les résultats sont décevants et donc une imprimante 3D est impérative pour la réalisation des parties les plus compliquées.

Par exemple, voici un début de châssis :

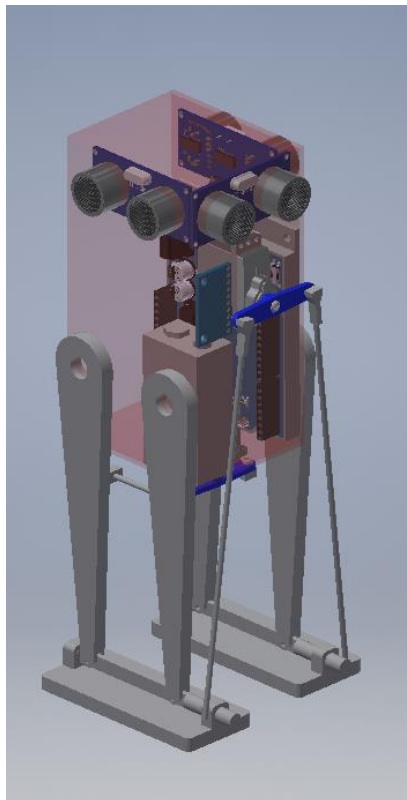


(Très décevant)

Sans import de matériaux, de matériels et de l'impression en 3D des pièces comme les jambes et chevilles, la réalisation reste très technique voire impossible.

Assemblage mécanique :

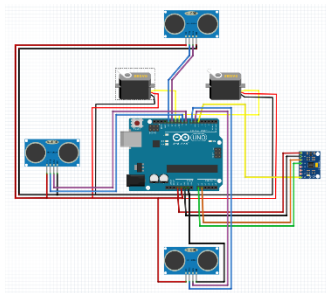
Pour commencer l'assemblage du robot, nous commencerons par l'insertion du matériel électronique dans le châssis :



Nous utiliserons des vis pour fixer les composants électroniques sur le châssis. Le châssis a été prévu pour pouvoir contenir ces différents éléments. Puis vous pouvez directement voir sur le schéma si dessus que nous utiliserons aussi des vis/boulons pour fixer les jambes au châssis. Puis ensuite, nous avons la fixation des tiges sur les servomoteurs qui se fait directement grâce à des vis. L'assemblage mécanique en soit n'est pas complexe si nous réussissons à réaliser correctement nos différentes pièces.

Assemblage électrique :

Dans cette partie, nous parlerons des câblages ainsi que du schéma que nous utiliserons. Les câblages seront simplement liés comme dans le schéma électrique. Nous utiliserons le soudage pour pouvoir lier les différentes connections nécessaire. Dans cette partie, nous devrons souder les câbles qui apportent l'énergie (5V) entre tous les composants qui utilisent 5V et ainsi qu'une masse unique. Le soudage est donc très important pour créer ces nœuds. Puis pour en revenir sur le schéma que nous utiliserons, étant donné que nous n'utilisons pas de « bras ». Donc, nous utiliserons que 2 servomoteurs. Nous utiliserons le second schéma prévu pour le robot.



6/ Simulation

Comment avance et marche notre robot ? Pour répondre à cette question, nous avons essayé de réaliser des simulations sur la partie mécanique et informatique. Toutefois nous aurions préféré faire ces tests sur un vrai prototype.

Simulation mécanique :

Pour comprendre comment avance notre robot sur l'aspect mécanique nous pouvons revenir sur les liaisons entre les différentes parties ainsi que sur la conséquence des mouvements des servomoteurs. Nous avons donc 4 liaisons importantes qui sont celle des 2 servomoteurs avec les tiges, ainsi que de la jambe et le châssis puis enfin, celle de la jambe et des pieds. Nous avons un premier mouvement verticale grâce au servomoteur situé sur l'avant du robot. Ce servo permet de pencher le robot sur un des cotes tout en levant l'autre pied. Ceci créer un premier mouvement vertical qui lève le pied. Nous avons un second mouvement horizontal grâce au second servomoteur situé en dessous du robot. Après que le pied est levé, nous pouvons donc exécuter le servomoteur qui fera un mouvement horizontal. Puis après ce mouvement horizontal, nous avons juste à refaire appelle au premier servomoteur qui déposera le pied. Voilà comment nous réussissons à faire avancer, tourner, reculer notre robot.

Après avoir assemblé notre robot sur Autocad, nous avons pu simuler ces mouvements qui encore une fois dans l'aspect théorique est fonctionnelle.

Simulation informatique :

Pour la partie simulation informatique nous avons donc opter pour l'utilisation de Thinkercad, nous avons donc pu tester notre code en utilisant tous nos composants sauf le gyroscope qui n'était pas présent sur la plateforme. Les simulations ont été concluantes en ce qui concerne le code malgré le fait que ce soit difficile de bien voir comment le robot avance et comment le code fonctionne.

Afin de s'assurer que chaque fonction du code fonctionne correctement nous avons divisé en plusieurs parties les simulations.

Tout d'abord on s'est assuré que les servomoteurs effectuent les bonnes rotations, suite à cela nous avons pu valider les fonctions permettant au robot de tourner et d'avancer.

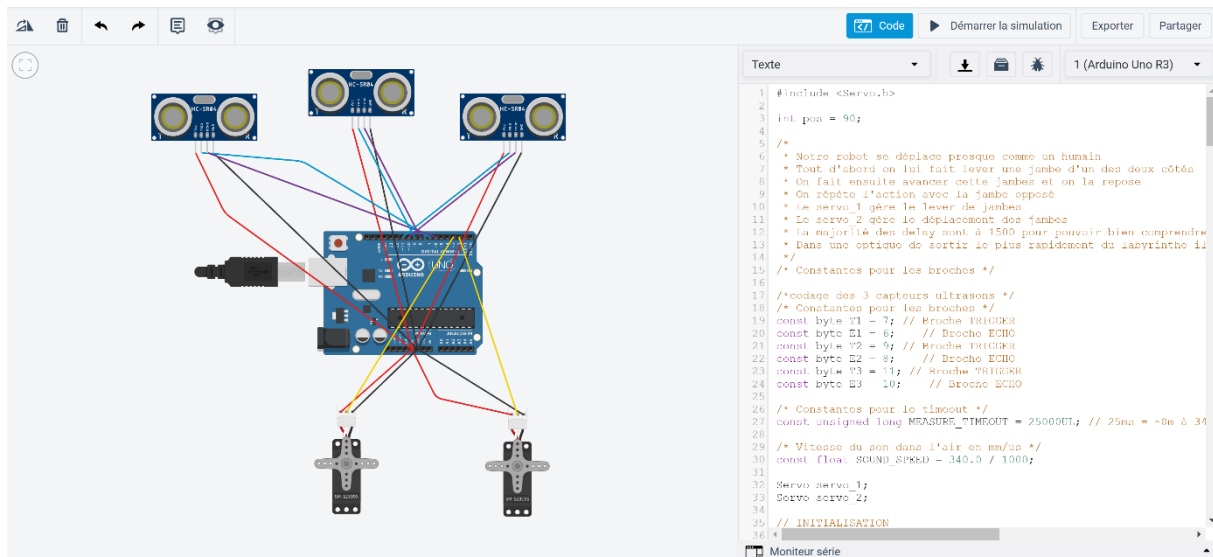
Dans un second temps, nous avons testé la détection de distance avec les capteurs ultrasons, notre robot comporte 3 capteurs de ce type, nous avons donc simulé le code pour un capteur puis on l'a réitéré pour les 2 autres capteurs.

Et pour finir nous avons exécuté tout le code en même temps qui comprend les déplacements du robot, la détection de collision et l'algorithme permettant la sortie efficace et rapide du labyrinthe.

Globalement, nous n'avons pas rencontré beaucoup de problème lors des simulations, juste quelques erreurs basiques concernant le code.

Toutes ces parties sont décrites dans la partie précédente avec les détails du code.

Voici le plan final des simulations :



Si vous souhaitez tester la simulation :

[Lien vers la simulation Thinkercad](#)

7/ Conclusion

Nous avons réalisé qu'encore une fois, tout est une question de préparation, d'analyse et surtout de communication pour mener à bien un projet d'équipe. Nous avons pu découvrir des technologies intéressantes comme le gyroscope où la modélisation 3D tout au long de l'année dans le cadre du projet. Cependant nous avons rencontré plusieurs problèmes. Tous d'abord, lorsqu'on a commencé à répondre aux différentes questions liées au cahier des charges, nous nous sommes rendu compte que nous devrions passer outre certaines contraintes afin de réaliser un projet plus fiable. Nous avons donc ignoré les bras du robot afin d'alléger le schéma électrique ainsi que l'aspect mécanique et le code. Nous avons aussi principalement utilisé les capteurs ultrasons comme principale, voir unique source d'apport d'information en négligeant l'utilisation du gyroscope.

De plus, dû au confinement, la réalisation d'un prototype a été compromis et nous n'avons pu travailler que sur l'aspect théorique. Les simulations et l'assemblage ne restent que théorique et nous n'avons donc pas pu corriger/vérifier le bon fonctionnement du robot. Néanmoins, en ayant augmenté au maximum la fiabilité du robot, nous pensons que le robot pourra, au moins grâce à la structure mécanique, électronique et du code, exécuter les fonctions principales tel que :

- Mouvements principaux : avancer, tourner, s'arrêter.
- Prendre des informations sur l'environnement et les obstacles grâce aux 3 capteurs.
- Pouvoir sortir d'un labyrinthe grâce aux mouvements et capteurs.

Nous avons donc négligé :

- L'utilisation d'un second capteur : gyroscope.
- La création de bras avec l'utilisation d'1 servomoteur en plus avec toute l'aspect mécanique.
- La création d'un prototype.

Pour finir, nous avons essayé de faire en sorte qu'en lisant ce rapport, le prototype soit réalisable sur tous les aspects (mécanique, électrique, information).

Nous vous remercions pour votre attention sur notre rapport et encore une fois nous vous présentons nos excuses pour le retard du rapport.