# Rapport de séance

## Séance du 10/12/18

Aujourd'hui, nous avons travaillé en groupe avec Hansi pour bien se mettre les idées au clair sur l'avancement du projet.

Nous avons rédigé le cahier des charges ainsi que le diagramme de Gantt (qui seront joints), et nous avons réfléchi aux problématiques de notre sujet.

Nous avons appris que le piston n'était pas une solution pour le système de tir, donc nous avons cherché une solution alternative. Les servomoteurs n'ont pas forcément une grande puissance, mais si on veut plus tard mettre en place un deuxième robot "gardien", il faut avoir le temps de le diriger pour arrêter la balle.

L'enseignant nous a également parlé d'une méthode avec un élastique qui est tendu puis lâché. Nous avons nous eut l'idée de créer un piston à partir de deux servomoteurs qui poussent une planche qui elle-même pousserait le ballon contrôlé entre des "bras" (pas forcément articulés).

Nous allons nous rendre en fin de semaine au FabLab afin de mettre en place un premier châssis pour le robot joueur, ainsi qu'une petite maquette pour notre modèle de piston.

Nous avons essayé de tester la puissance de deux servomoteurs sur une balle de golf, mais nous avons rencontré beaucoup de problèmes de téléversement.

## Séance du 17/12/18

Aujourd'hui, nous continuons de travailler en commun afin de trouver le système de tir le plus adapté possible. Nous avons rencontré des problèmes pour mettre deux servomoteurs sur une seule carte Arduino, nous avons donc fait des recherches sur la vitesse de rotation de tous les servomoteurs disponibles pour déterminer celui qui nous permettrait de mettre le plus de puissance dans la frappe de balle. Nous allons donc utiliser le Servomoteur : Mighty mini Hitec HS-225MG.

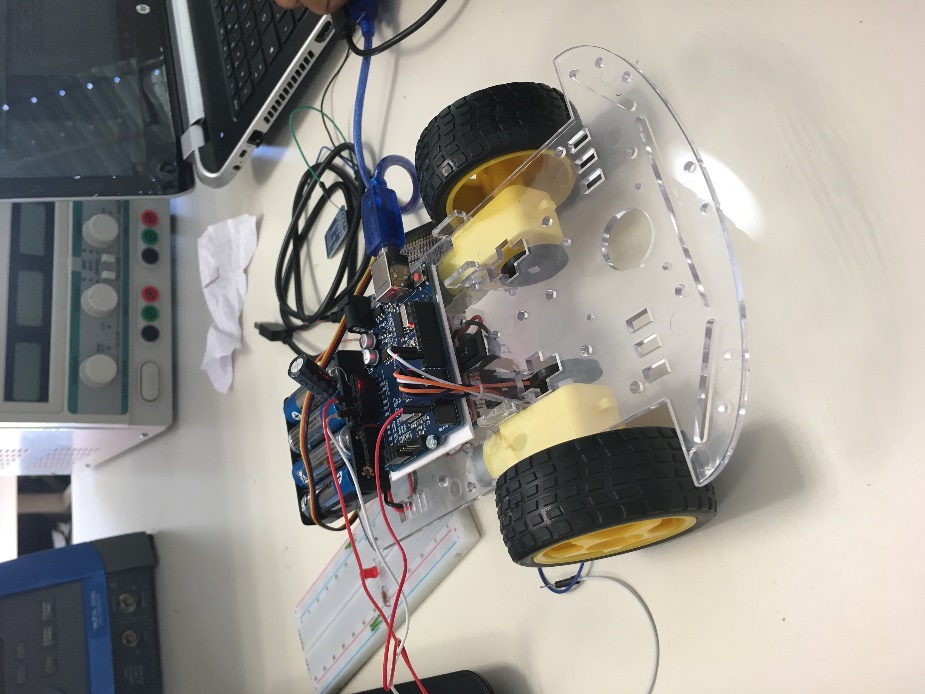
Nous avons improvisé (avec du papier et du scotch) une manière de rallonger la pale du servomoteur afin de gagner encore un peu de puissance, et nous sommes arrivés à un assez bon résultat : En enroulant du papier autour de la pale, la balle parcourt environ 30cm, ce qui est assez bien pour le système utilisé. De plus, la balle part à peu près droit, ce qui peut encore être améliorer en faisant une pièce plus travaillée et plus adaptée pour le contact entre la pale et la balle. Ci-joint une vidéo montrant notre frappe.

<https://youtu.be/uZQ_cibHLWE>

Nous avons également travaillé sur le châssis : après réflexion, nous avons pensé que le plus adapté était le châssis des voitures utilisées en TD. Ceci va nous permettre de contrôler les moteurs (et peut être les servomoteurs du système de contrôle de balle) directement sur la carte Arduino UNO qui est sur ce robot.

Nous allons rajouter à l'avant du robot tout notre système de tir, ainsi que notre système de contrôle (en effet le robot est un peu bas, la balle ne passe pas dessous). Ce robot va donc être notre base, et nous allons rajouter tout ce dont nous avons besoin pour notre projet.

Nous travaillons maintenant sur des options pour pouvoirs utiliser plusieurs servomoteurs sur une même carte, notamment en utilisant l’alimentation des piles afin de pouvoir utiliser des servomoteurs pour notre système de contrôle de balle.



Ici, on a une vue de l’avant du robot que nous allons utiliser comme base. Nous avons démonté les composants qui ne nous servent à rien, et nous avons donc une certaine place pour nos composants à l’avant.

## Séance du 07/01/19

Maintenant que nous avons réglé le système de tir, nous nous occupons du déplacement du robot. A partir des anciens TD (TD 8), nous avons pu assez rapidement réussir à avoir un bon déplacement (avant, arrière, gauche, droite), malgré la limite de la portée du Bluetooth. Nous pensons tout de même que cela ne posera pas de problème avec les dimensions du terrain. Voici une vidéo de pilotage de notre robot :

https://www.youtube.com/watch?v=n10RO3F-3z0

Nous avons donc apporté des modifications aux programmes pour avoir un robot adapté à nos besoins, en inversant par exemple l’avant et l’arrière du robot. En effet, nous réfléchissons donc à la disposition des éléments pour que notre robot soit efficace, malgré le fait que nous ne puissions pas changer la forme étant donné que nous utilisons un robot de TD. Nous allons donc créer un module à l’avant afin de contrôler et tirer la balle, car le châssis est trop bas pour que la balle passe dessous.

Nous continuons également de chercher des solutions pour utiliser plusieurs servomoteurs, car en testant avec un programme simple nous n’arrivons pas à en faire marcher deux sur la même carte, alors que nous avons besoin qu’ils soient sur le même module Bluetooth que les moteurs.

Nous allons donc essayer une solution qui consiste à les alimenter directement depuis les piles pour avoir un courant supérieur ce qui permettrai de faire fonctionner les deux moteurs. Nous allons réaliser les branchements dans la semaine, et si cela fonctionne nous mettrons en place un module à l’avant du robot adapté aux deux servomoteurs que nous avons.

## Séance du 14/01/19

Les branchements que nous avons essayé durant la semaine ne fonctionnent pas, nous n’avons donc pas encore la certitude du matériel utilisé, donc nous n’avons pas pu faire le module à l’avant car il doit être adapté au matériel utilisé.

Aujourd’hui, je m’occupe de la partie programmes, en cherchant des solutions adaptées à nos besoins pour les servomoteurs. En attendant de réussir à utiliser les deux servomoteurs sur une même carte (rôle d’Hansi aujourd’hui), je crée les programmes pour contrôler l’ouverture et la fermeture de la trappe ainsi que le système de tir par Bluetooth. Une fois cela terminé, en attendant de pouvoir essayer, je m’occupe de mettre en place un début de programme complet du projet, avec le contrôle du déplacement et des servomoteurs dans le même programme, et des observations sur les parties de code.

Ensuite, Hansi réussit le branchement de la deuxième carte directement reliée aux piles, ce qui permet de faire fonctionner nos deux servomoteurs en même temps sur la carte. Nous avons réussi à les utiliser en même temps mais nous n’avons pas réussi à les faire fonctionner tous les deux comme nous le souhaitions par Bluetooth :

* Un doit faire un quart de tour et revenir à sa position initiale, ce sera celui dédié au tir.
* L’autre doit faire un demi-tour, rester en position jusqu’à ce qu’on le ferme (deuxième ordre que l’on donne par Bluetooth également).

Nous avons donc rajouté trois boutons sur notre configuration Bluetooth pour tirer, ouvrir et fermer la trappe, mais notre module Bluetooth ne fonctionne pas très bien et se déconnecte souvent.

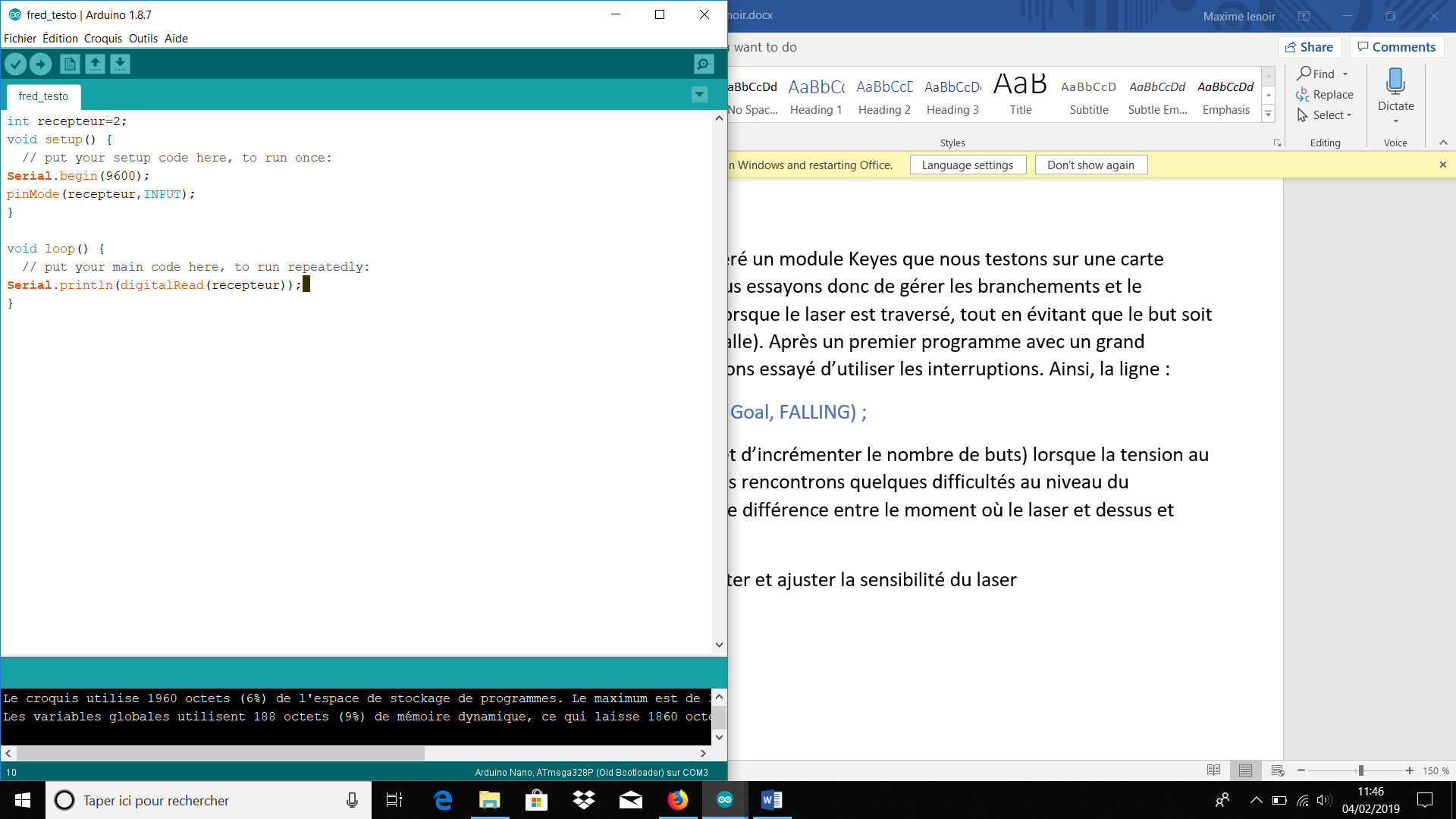
Il nous reste seulement ce léger soucis à régler afin de terminer le travail électronique sur la voiture, il ne restera ensuite plus qu’à créer le module pour l’avant au FabLab pour terminer cette partie centrale du projet.

## Séance du 04/02/19

Aujourd’hui nous travaillons sur le laser. J’avais préalablement fait des recherches pour le fonctionnement de celui-ci, et je l’avais étudié dans le cas d’ue alarme qui sonne lorsqu’on le traverse. Nous avons récupéré un module Keyes que nous testons sur une carte Arduino nano qui sera la carte à utiliser pour nos cages. Nous essayons donc de gérer les branchements et le programme pour avoir un compteur de but qui augmente lorsque le laser est traversé, tout en évitant que le but soit compté plusieurs fois (par exemple lorsqu’on récupère la balle). Après un premier programme avec un grand « delay » pour avoir le temps de récupérer la balle, nous avons essayé d’utiliser les interruptions, à l’aide du site LOCODUINO(https://www.locoduino.org/spip.php?article64). Ainsi, la ligne :

attachInterrupt(2, Goal, FALLING) ;

nous permet d’appliquer la fonction Goal() (qui nous permet d’incrémenter le nombre de buts) lorsque la tension au niveau du récepteur passe de HIGH à LOW. Cependant, nous rencontrons quelques difficultés au niveau du récepteur, qui avec ce second programme ne marque pas de différence entre le moment où le laser et dessus et celui où il n’y est pas.

En faisant un second programme simple, nous avons pu tester et ajuster la sensibilité du laser, afin d’être sûrs de son fonctionnement. Il faut alors maîtriser les interruptions pour avoir un programme qui fonctionne, car même si le laser fonctiponne (on a mis un testeur dans le programme), nous n’arrivons pas à incrémenter le nombre de but, c’est-à-dire l’action qui se fait dans l’ISR (Interrupt Service Routine), lorsqu’on passe de LOW à HIGH (il faudra donc mettre RISING en troisième argument de attachInterrupt).

Finalement, nous avons trouvé la solution, et le module de franchissement de ligne de but ainsi que le compteur fonctionne. Il nous reste, pour ce qui est des cages, à écrire le nombre de but(s) sur un écran LCD qui sera situé à côté des cages (j’ai fait les branchements, il ne reste que le programme à faire).

Nous allons pouvoir nous attaquer à la mise en forme du robot en assemblant les différents modules.