

Projet Arduino - PEIP2

Année scolaire 2018-2019

Rapport de projet : Le vélo télécommandé

Étudiants :

Sacha PSALMON et Enzo ISNARD

Encadrants :

Mr. Pascal MASSON et Mr. Nassim ABDERRAHMANE

REMERCIEMENTS

Nous remercions tout particulièrement Monsieur Pascal MASSON et Monsieur Nassim ABDERRAHMANE pour leur disponibilité et leur aide précieuse, sans oublier l'école Polytech Nice-Sophia pour nous avoir permis de réaliser ce projet. Nous remercions aussi toute l'équipe du Fablab SoFAB, dont Marc Forner sans qui la réalisation de la maquette n'aurait pas été possible.

SOMMAIRE

Introduction.....	4
Description du projet et de son fonctionnement.....	5
Réalisation du projet.....	9
Conclusion et perspectives.....	13
Bibliographie.....	15

INTRODUCTION

Dans le cadre de notre enseignement “Électronique avec Arduino” nous avons dû réaliser un projet durant huit séances au total étalées sur environ quatre mois. Après avoir beaucoup hésité sur la nature de notre projet - nous voulions au départ réaliser un système de communication en code Morse via radio, puis un avion télécommandé - nous nous sommes arrêtés sur l'idée du “vélo télécommandé”. C'est un projet plus complexe qu'il en a l'air. En effet, la position d'équilibre d'un vélo est extrêmement instable et pour pouvoir être dirigé correctement il doit être le plus droit possible.

Nous n'avons pas trouvé énormément d'exemples sur Internet (tout au plus, une dizaine de vidéo youtube de démonstrations de projets de fin d'études), et la réalisation d'un tel projet était très peu documentée. Seul le site d'étudiants indiens fournissait des informations concrètes (voir point [1] de la bibliographie). Cependant, étant donné qu'ils présentaient un projet non fonctionnel dans leurs vidéos de démonstration, nous n'avons pas pris leurs conseils pour des vérités absolues.

Avant de nous lancer, nous avons défini le cahier des charges suivant : Le vélo doit être capable de s'équilibrer grâce à un volant d'inertie contrôlé par un régulateur PID. Le vélo doit pouvoir maintenir cette position d'équilibre en toutes circonstances, qu'ils soit en train de tourner, sur un plan incliné ou bien face à des obstacles minimes (petites bosses, etc ...).

Nous pouvions donc diviser la réalisation en plusieurs parties :

- La gestion de l'équilibrage du vélo : qui comprend la création de la roue d'inertie par nos soins, l'utilisation d'un module MPU 6050 comprenant un gyroscope et un accéléromètre, la gestion du PID et donc de l'asservissement d'un moteur 12V qui entraîne la roue.

- La gestion de ses déplacements : montage de roues GMPW sur un moteur 12V, gestion d'un module bluetooth HC-06 pour contrôler les déplacements du vélo. Ainsi que le contrôle d'un servomoteur contrôlant la roue avant pour permettre à l'ensemble de tourner.

- La réalisation de la maquette : construction à l'aide de pièces Meccano et de contreplaqué de la structure du vélo.

Nous allons dans ce rapport détailler le fonctionnement concret du vélo télécommandé tout en abordant sa réalisation. Enfin, nous concluerons par une synthèse de ce que nous avons produit au cours de ces quatre mois ainsi que par un détaillage des choses à revoir ou à améliorer, ou encore de ce que nous aurions fait si tout était à refaire.

DESCRIPTION DU PROJET ET DE SON FONCTIONNEMENT

Tout d'abord, il faut savoir que notre vélo peut être modélisé par un pendule inversé, à savoir une tige rigide (en théorie de masse négligeable) dont l'essentiel de la masse est située en hauteur. Il est facile de voir que le système est instable. Si le vélo s'incline légèrement il subira un moment de force à cause son poids, ce qui l'entraîne vers le sol. Une des manières de rééquilibrer le vélo est d'exercer un moment opposé à celui du poids, ce qui peut se faire via une roue d'inertie. On a que l'angle entre le vecteur position allant jusqu'à la roue et le poids est $\theta - \pi$. Ainsi on a que:

$$M_0(\vec{P}) = ||\vec{OM} \wedge \vec{P}|| = m \cdot g \cdot l \cdot \sin(\pi - \theta) = m \cdot g \cdot l \cdot \sin(\theta) \quad (1)$$

Au niveau du sens on utilise la règle de la main droite. Dans la situation illustrée sur le schéma le moment du poids sort de l'écran. Cela veut dire que pour rééquilibrer le vélo il faudra créer un moment qui rentre dans l'écran, et donc faire tourner la roue dans le sens horaire.

La norme du moment de force créé par la roue a comme expression:

$$M_{roue} = I_O \cdot \alpha \quad (2)$$

Où I_O est le moment d'inertie de la roue relativement au point O et α son accélération angulaire. Pour calculer I_O on peut d'abord commencer par calculer le moment d'inertie par rapport au centre de la roue qu'on appellera I_C . Comme notre roue est un cylindre plein de rayon r I_C a comme expression:

$$I_C = \frac{1}{2} m \cdot r^2 \quad (3)$$

(Pour le détail des calculs, voir point [2] de la bibliographie)

Ensuite pour obtenir I_O on utilise le théorème de Huygens-Steiner et on a que:

$$I_O = I_C + m \cdot l^2 = m \left(\frac{1}{2} r^2 + l^2 \right) \quad (4)$$

On peut dorénavant trouver l'accélération angulaire minimale pour rééquilibrer le vélo en égalisant (1) et (2):

$$\alpha = \frac{g \cdot l \cdot \sin(\theta)}{0,5 \cdot r^2 + l^2} \quad (5)$$

On voit que cette expression ne dépend pas de la masse la roue, ce qui nous a fait longtemps penser que la masse la roue n'avez pas d'importance. En fait elle a un rôle cruciale puisque pour que les calculs soient justes il faut que le centre de masse du système se trouve au centre de la roue. En plus on s'appuie sur un modèle assez approximatif donc il est probable que la masse intervienne au final. Ce modèle ne prend pas aussi en compte le fait que le moteur ait un certain délai pour changer de direction et qu'il faille donc anticiper le moment où on arrive au point d'équilibre. Pour cela on doit utiliser un régulateur, comme le PID.

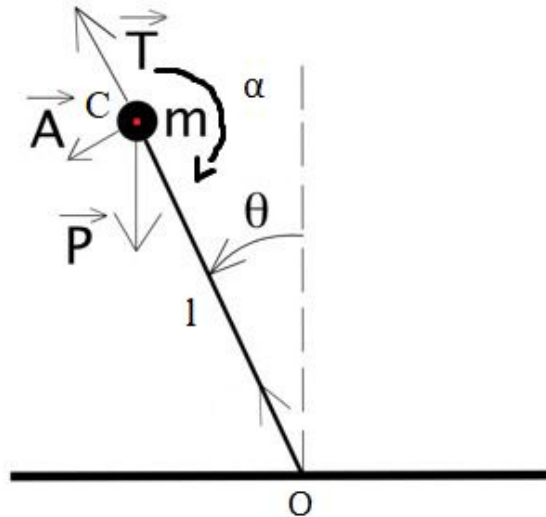


Schéma d'un pendule inversé possédant une roue d'inertie en son sommet.

Pour ce qui est du PID, c'est un système qui permet d'améliorer les performances d'un asservissement. C'est un des régulateurs les plus utilisés dans l'industrie et il est assez facile à mettre en place dans le cadre d'un projet incluant une carte Arduino grâce à des bibliothèques simplifiant son utilisation comme celle créée par Brett Beauregard (que nous avons utilisé). Ce régulateur fonctionne grâce à trois modules, Proportionnel, Intégral et Dérivé qui fonctionnent dans le cadre de notre projet en parallèle (l'entrée de chacun des modules est identique, contrairement à un fonctionnement en série). On donne au régulateur une valeur consigne, ici la position droite du vélo. Son objectif est de la maintenir peu importe les perturbations. Pour se faire, le système mesure à chaque boucle son décalage par rapport à la valeur consigne, c'est "l'erreur" qu'il va devoir corriger. Un système PID classique est contrôlé par trois paramètres, un pour chaque modules, K_p , K_i et K_d .

Le Proportionnel multiplie l'erreur par K_p donc plus le système est éloigné de sa position d'équilibre, plus la correction apportée (donc l'accélération angulaire de la roue d'inertie dans un sens ou dans l'autre) sera importante. Un coefficient K_p trop important entraînerait un dépassement de la valeur consigne à chaque correction et ferait donc osciller le système autour de sa position d'équilibre.

L'Intégral intègre l'erreur en fonction du temps et divise l'intégrale par K_i . Ce module est surtout utile dans les systèmes ayant un point d'équilibre stable. Il permet de réduire l'erreur

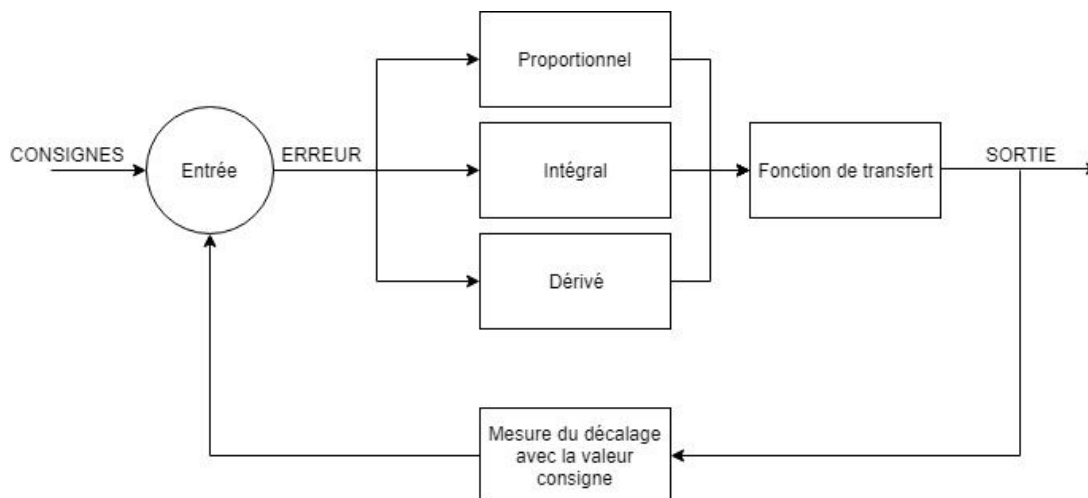
statistique mais dans notre cas l'augmentation de K_i ferait davantage osciller le système, il faut donc garder cette valeur relativement basse.

Le Dérivé dérive l'erreur en fonction du temps et multiplie la dérivée par K_d . Il permet d'anticiper les mouvements du système et d'appliquer une correction inverse pour éviter le dépassement de la valeur consigne. Un coefficient K_d trop grand résulte en des réactions violentes du système dues à une trop grande sensibilité et le rendront donc impossible à équilibrer correctement.

Il faut donc équilibrer avec minutie ces trois paramètres pour obtenir une correction satisfaisante. Dans notre cas, la seule façon de faire est la méthode empirique, on augmente tout simplement les coefficients un par un jusqu'à obtenir un bon résultat. D'autres méthodes existent, consistant en des calculs de périodes et d'amplitude d'oscillation mais la méthode empirique est la plus simple à mettre en place. L'industrie dispose aussi d'appareils permettant de régler ces coefficients automatiquement.

Pour résumer, avec un système PID, plus l'erreur est grande et plus l'erreur est là depuis longtemps, plus on la corrige. Moins l'erreur est grande, moins on la corrige. Et si les paramètres ne sont pas bien réglés, le système oscille ou n'est tout simplement pas stable.

La portion du programme gérant l'équilibrage du vélo ne peut pas réellement être représentée par un algorithme étant donné que son squelette n'est composé que d'une boucle PID (détaillée dans le schéma ci-dessous). La mesure du décalage par rapport à la valeur consigne et faite par le gyroscope via notre fonction `readGyro()` (les mesures étant légèrement lissées grâce à la fonction `smooth()`), le calcul du résultat de chaque module et de la valeur donnée en sortie est donné par la fonction `PID.compute()` (comprise dans la librairie `PID`).



La fonction de transfert d'un régulateur PID étant donnée par : (avec p la variable de Laplace)

$$C(p) = K_p + \frac{1}{K_i} \cdot \frac{1}{p} + K_d \cdot p$$

Quant à la gestion du moteur de la roue arrière, la vitesse du moteur est donnée grâce à l'algorithme suivant, pour envoyer des messages via Bluetooth nous utilisons un smartphone et l'application Bluetooth Electronics.



Pour contrôler un servomoteur afin de modifier l'angle de la roue avant, l'algorithme est plus ou moins le même. Le message commence par une autre lettre (C par exemple) et l'entier qui suit la lettre correspond à l'angle voulu.

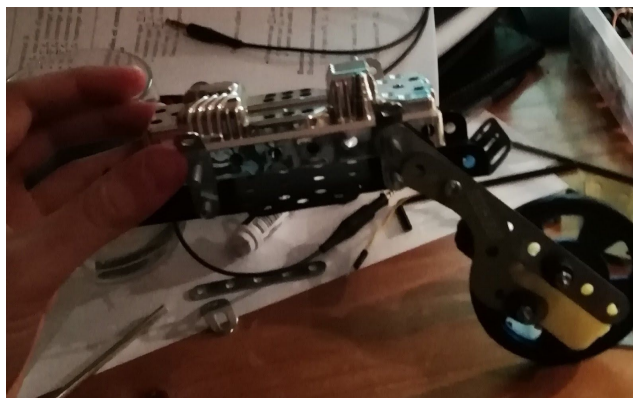
RÉALISATION DU PROJET

Nous avons consacré les premières séances dédiées à la réalisation des projets à l'établissement d'un planning (représenté ci-dessous par un diagramme de Gantt), et à nous remettre dans le bain au niveau de la mécanique (vous pouvez voir dans la partie précédente les calculs que nous avons réalisé).

	Semaine 1	Semaine 2	Semaine 3	Semaine 4	Semaine 5	Semaine 6	Semaine 7	Semaine 8	Semaine 9
Définition	Nous deux								
Achats	Nous deux								
Gyroscope		Sacha	Sacha	Sacha					
Roues		Enzo	Enzo	Enzo					
Volant inertie		Sacha	Sacha	Sacha	Sacha	Sacha			
Déplacement		Enzo	Enzo	Enzo	Enzo	Enzo			
Programmation du volant			Sacha	Sacha	Sacha	Sacha	Sacha	Sacha	
Tourner					Enzo	Enzo	Enzo		
Bluetooth				Sacha	Sacha	Sacha			
Structure			Enzo	Enzo	Enzo	Enzo			
Correction des bugs du 1 ^{er} semestre					Le concerné	Le concerné			
Tests, corrections et optimisation							Le concerné	Le concerné	
Finalisation et dernières vérifications									Nous deux

Ce planning n'a pas été respecté mais nous reviendrons sur ça plus tard. Après avoir géré les achats à faire (nous avons commandé notre matériel sur le site <https://www.robotshop.com/>), nous avons commencé à tester les pièces fournies par notre professeur, à savoir le moteur qui entraîne notre roue d'inertie, notre roue ainsi ainsi que le gyroscope. Tout s'est déroulé sans accrocs sauf pour le gyroscope. Nous avons trouvé un programme de calibration adapté au MPU 6050 (réalisé par mais nous ne lisions que des caractères incompréhensibles à cause d'une différence de vitesse de communication entre votre programme arduino et le moniteur série. Le problème fut vite réglé.

Pendant que la construction du vélo avançait nous avons trouvé un exemple de programme d'équilibrage d'un pendule inversé relativement compréhensible (avec du recul, il était sûrement même trop simpliste) dont nous nous sommes inspirés pour créer le nôtre car notre configuration n'était pas du tout identique à la leur (voir point [3] de la bibliographie). Voilà l'état dans lequel était le vélo aux environs du 11 janvier 2019 :



Le moteur entraînant la roue d'inertie (qui à cette époque était un cercle en contreplaqué sur lequel nous avons vissé une petite plaque d'aluminium pour y encastrer le rotor du moteur, illustration ci-dessous) s'est soudainement mis à ne plus fonctionner mais ce n'était qu'un problème de branchements. Nous avons abandonné ce design pour la roue d'inertie car la répartition non uniforme du poids faisait trembler tout le système ce qui n'est pas souhaitable. De plus, les écrous fixant la roue arrière du vélo n'étaient pas tout à fait à la bonne taille donc tout ne tenait pas bien.



En outre, nous avons des problèmes avec les fixations de la roue avant. Seul un seul type de vis rentrait dans la roue et cela ne permettait pas à la fois de laisser la roue tourner et faire en sorte qu'elle ne sorte pas de son axe. Au final nous avons agrandi le trou au centre de la roue et utilisé une tige de fer comme axe. Nous avons aussi bloqué le tout avec des morceaux de caoutchouc mais le frottement avec la roue était trop important. Elles ont été remplacées par des écrous qui ne maintenaient cependant pas assez la roue. Elle était donc trop libre mais nous n'avons pas trouvé de solutions à ce problème.

Après une idée de monsieur Nassim ABDERRAHMANE nous avons décidé d'utiliser des colliers de serrage pour fixer la roue ainsi que l'essentiel de la structure. En effet, certaines pièces Meccano se pliaient à cause du poids du vélo. La fonctionnalité prônant sur l'esthétisme, c'était une bonne solution.

Nous avons commencé à réfléchir à comment fixer le volant d'inertie sur le vélo. Nous avons passé un bon moment à dessiner sur ordinateur les pièces qui serviraient à cela. Nous avons pour cela utilisé le logiciel Fusion 360. Quelques jours plus tard nous sommes allés au fablab pour les découper dans du contreplaqué à l'aide de la découpeuse laser. Mis à part quelques erreurs de mesures peu gênantes, nous avons fait du bon travail. Nous avons donc maintenant un panneau dans lequel nous colleront notre moteur, fixé sur la structure en

mécano par un “pont” et des équerres, ainsi qu’une nouvelle roue d’inertie pour éviter les tremblements.

Voici donc à quoi ressemblaient les pièces à la sortie de la découpeuse :



Après montage et collage nous avons donc enfin pu tester le programme et l’affiner avant de nous rendre compte de ce qui sera notre plus grosse erreur. Pendant toute la réalisation du projet nous avons négligé quelque chose de fondamental, à savoir que les calculs que nous avons fait n’étaient juste que si le centre de masse de la structure. Nous avons pendant des mois négligé l’importance du poids de la roue d’inertie, ce qu’il ne fallait pas faire. Les autres étudiants le sachant ont donc privilégié des roues d’inerties très lourdes comparées à leur structure pour maximiser son effet. L’effet de notre roue d’inertie très légère était totalement négligeable devant l’action du poids et notre système malgré un code correct ne pouvait pas s’équilibrer. Nous avons, dans le doute, essayé de découper une autre roue d’inertie cette fois ci bien plus massive. Mais notre structure en Meccano était beaucoup trop lourde pour que notre petit moteur puisse faire tourner une roue efficace.

Nous n’avions pas le temps de refaire toute la structure, de fabriquer une roue d’inertie pesante et d’attendre qu’un moteur adapté arrive (il ne nous restait pas plus de deux semaines avant la soutenance). Nous avons donc essayé de boucler le reste de notre cahier des charges.

Nous avons très vite compris qu’il serait impossible d’adapter un servomoteur pour faire tourner la roue avant sur la structure que nous avons produit. L’intégration du module Bluetooth fut relativement simple, nous avons ensuite fait du “cable management” pour fixer tout notre matériel sur la structure du vélo.

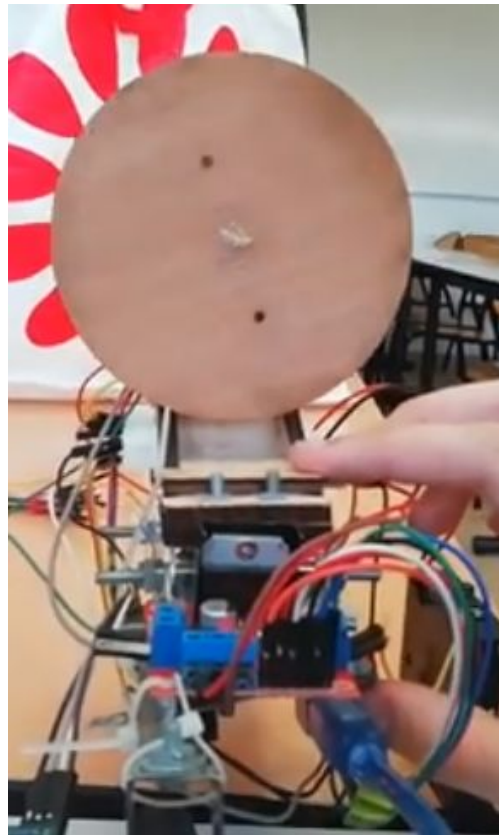
Le planning fixé au début du projet n’a pas été respecté. Nous avons en effet beaucoup sous estimé le temps qu’il nous faudrait pour construire un structure convenable, grande partie de ce que nous avons prévu de faire à mi-projet s’est retrouvée décalée à la fin. Nous avons

donc dû faire beaucoup de choses en peu de temps, le résultat n'est pas aussi satisfaisant que ce que nous aurions pu espérer en commençant l'aventure.

Pour comparer, voici le planning final que nous avons suivi :

	Semaine 1	Semaine 2	Semaine 3	Semaine 4	Semaine 5	Semaine 6	Semaine 7	Semaine 8	Semaine 9
Définition	Nous deux								
Achats	Nous deux								
Gyroscope		Sacha							
Roues		Enzo	Enzo	Enzo					
Volant inertie				Sacha	Sacha	Sacha	Sacha	Sacha	
Déplacement			Enzo	Enzo	Enzo	Enzo			
Programmation du volant			Sacha	Sacha	Sacha	Sacha	Sacha	Sacha	
Tourner									
Bluetooth								Sacha	Sacha
Structure		Enzo	Enzo	Enzo	Enzo	Nous deux	Nous deux	Enzo	
Correction des bugs du 1 ^{er} semestre									
Tests, corrections et optimisation									
Finalisation et dernières vérifications								Nous deux	Nous deux

Pour conclure, voici une photo de face du vélo à la fin de ce projet :

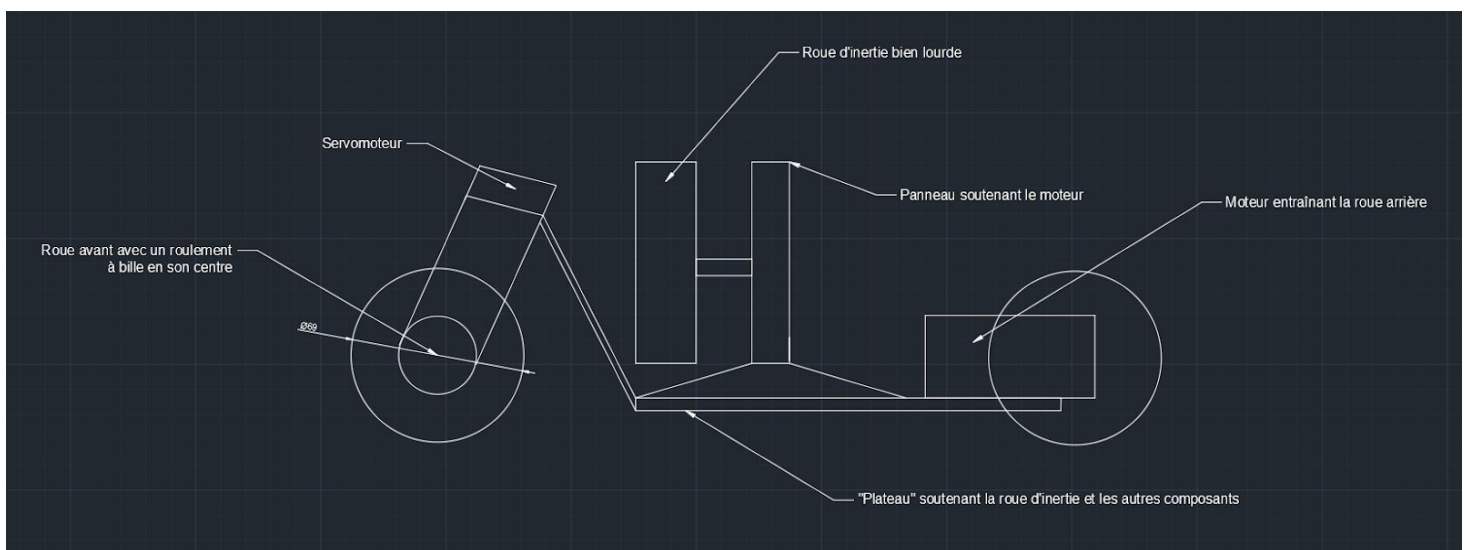


CONCLUSION ET PERSPECTIVES

Finalement, notre projet, bien que non fonctionnel nous aura beaucoup apporté. Le cahier des charges n'a pas été respecté dans son entièreté. Malgré que la roue d'inertie réagisse convenablement à l'inclinaison du vélo, le vélo ne parvient pas à s'équilibrer à cause d'un gros problème de design. Le vélo est télécommandable mais il ne peut pas tourner. De plus, nous avons bien compris les concepts physiques derrière régissant le projet ainsi que le fonctionnement du régulateur PID. Nous avons donc réalisé ce que nous pourrions appeler un prototype de vélo télécommandé. Nous avons aujourd'hui toutes les clés en main pour réaliser ce projet dans son intégralité en un minimum de temps.

Cependant, le plus gros problème que nous avons rencontré est un soucis d'organisation. Nous n'avions pas réalisé que c'était construire le vélo en lui même qui allait nous prendre le plus de temps. De plus, sachant qu'on ne pouvait pas tester le programme ni même les moteurs avant la finalisation de la maquette, nous aurions dû nous focaliser dessus à deux dès le départ, chacun s'occupant des différentes parties du vélo.

Beaucoup de choses peuvent être améliorées. Concernant un nouveau design de vélo, il faudrait réussir à créer une structure légère ayant un centre de masse bas (plus le centre de masse est bas, plus l'action du poids sur la structure est faible) et centré sur la roue d'inertie. Il serait fait en contreplaqué (découpé grâce à la découpeuse laser du fablab) car ce type de matériau pèse beaucoup moins lourd que les pièces Meccano. Le contreplaqué est de plus beaucoup plus "malléable" et nous permettrait d'être plus inventifs. Nous aurions d'ailleurs une structure beaucoup plus rigide. Nous avons réalisé un schéma décrivant la structure de vélo que nous adopterions si nous devions recommencer le projet aujourd'hui.



Un roulement à bille placé au centre de la roue avant nous permettrait de bien la fixer sur son axe et donc de pouvoir faire tourner le vélo avec beaucoup de précision.

Au niveau du matériel utilisé, seul le moteur entraînant la roue d'inertie changerait. Nous pourrions utiliser un vieux moteur de disque dur ou bien tout simplement un moteur possédant un gros couple. Nous n'avons pas encore d'idée fixée quant à la fabrication de la roue, nous pensons qu'un disque en métal ferait l'affaire mais nous ne savons pas comment le fabriquer.

Par rapport au programme, nous aurions pu utiliser un filtre de Kalman pour que nos mesures soient plus précises, ce qui est souhaitable s'il on veut que le système soit très stable. Ce dernier -même s'il est beaucoup plus complexe mathématiquement que le PID- est utilisable en Arduino grâce à des bibliothèques simplifiant son utilisation.

BIBLIOGRAPHIE

[1] Blog des étudiants Indiens :

[:https://iedprojects2015.blogspot.com/2015/03/self-balancing.html](https://iedprojects2015.blogspot.com/2015/03/self-balancing.html)

[2] Article wikipedia sur le moment d'inertie:

https://fr.wikipedia.org/wiki/Moment_d%27inertie#Le_cylindre_plein

[3] Site d'un projet sur le pendule inversé(dont le code sur le PID est en partie issu):

<https://www.instructables.com/id/Inverted-Pendulum-Robot-Using-a-Reaction-Wheel/>

[4] Vidéo nous ayant influencé dans notre choix de projet:

<https://www.youtube.com/watch?v=SUVtObDFFWY>

[5] Wiki sur la librairie que l'on a utilisée pour le PID:

<https://playground.arduino.cc/Code/PIDLibrary>

[6] Wiki sur la librairie du gyroscope:

<https://playground.arduino.cc/Main/MPU-6050>

[7] Cours de Monsieur Pascal MASSON :

<http://users.polytech.unice.fr/~pmasson/Enseignement-arduino.htm>

[8] Vidéo d'un test du projet:

<https://www.youtube.com/watch?v=cLNMnoQY6PE>