



*Rapport de projet*

**Hélicoptère avatar**

*Calendini-Clouvel*

*Encadrant : Monsieur Pascal Masson*

Ecole Polytechnique Universitaire de Nice Sophia-Antipolis, 1645 route des Lucioles,

Parc de Sophia Antipolis, 06410 BIOT

# *Remerciements*

*Nous voudrions remercier Monsieur Philippe Duroure (entreprise :* ***2iADP****) pour son aide précieuse dans la réalisation de l’usinage d’une partie du châssis ainsi que Monsieur Xavier Esa (entreprise :* ***DEBARGES****) pour nous avoir aidés lors de l’usinage des palonniers.*

*Nous tenions à remercier particulièrement Monsieur Pascal Masson (enseignant en électronique) ainsi que Monsieur Nassim Abderrahmane pour leurs conseils avisés tout au long de l’avancement du projet.*

# *Table des matières*

[*Remerciements* 2](#_Toc3804428)

[*Table des matières* 3](#_Toc3804429)

[*Introduction* 4](#_Toc3804430)

[*Matériel utilisé* 5](#_Toc3804431)

[*Déroulement du projet* 7](#_Toc3804432)

[*Conception* 7](#_Toc3804433)

[*Réalisation* 8](#_Toc3804434)

[**Electronique :** 8](#_Toc3804435)

[**Mécanique :** 10](#_Toc3804436)

[*Mise en commun* 12](#_Toc3804437)

[*Problèmes rencontrés* 13](#_Toc3804438)

[*Problème lié au Bluetooth :* 13](#_Toc3804439)

[*Problème lié à la stabilisation :* 13](#_Toc3804440)

[*Problème lié au jeu dans la liaison mécanique :* 14](#_Toc3804441)

[*Conclusion* 15](#_Toc3804442)

[*Annexes* 16](#_Toc3804443)

[*Cahier des charges* 16](#_Toc3804444)

[*Planning* 16](#_Toc3804445)

[*Bibliographie* 17](#_Toc3804446)

# *Introduction*

Nous présentons dans ce rapport notre projet d’Electronique de deuxième année d’étude. Ce projet consiste à concevoir et réaliser, en binôme, dans un temps imparti, un bicoptère. Nous détaillerons les étapes permettant d’atteindre nos objectifs, à savoir l’envol et la stabilité du système.

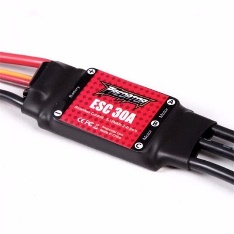
We are going to present you in this report our electronic project which goal is to conceive and realise, in pairs and in three months, a dualcopter. You will see all the steps we went through to reach our intent, namely taking off and being stable.

# *Matériel utilisé*

Afin de pouvoir réaliser ce projet, nous avons utilisé le matériel suivant :



* 2 moteurs brushless, 920 kv



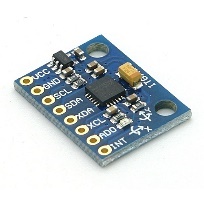
* 2 ESCs de 30A



* 2 servo-moteurs hitech
* 1 châssis dont une partie réalisée sur un centre d’usinage (CNC)



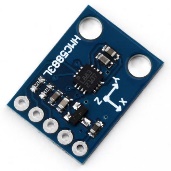
* 4 palonniers (dont 2 usinés)



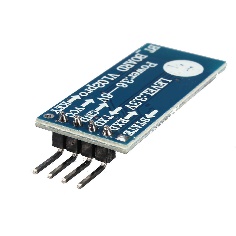
* 1 accéléromètre/gyroscope GY-521



* 1 carte arduino



* 1 magnétomètre
* 1 émetteur/récepteur radio

**

* 1 module bluetooth



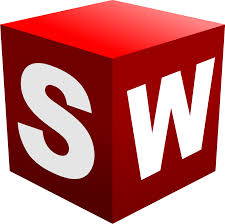
* 1 carte de contrôle KK2.1.5
* 1 tube en aluminium



* 2 hélices



* 3 batteries Lipo, 3S (12V)
* 1 batterie 5V



* Logiciel SolidWorks

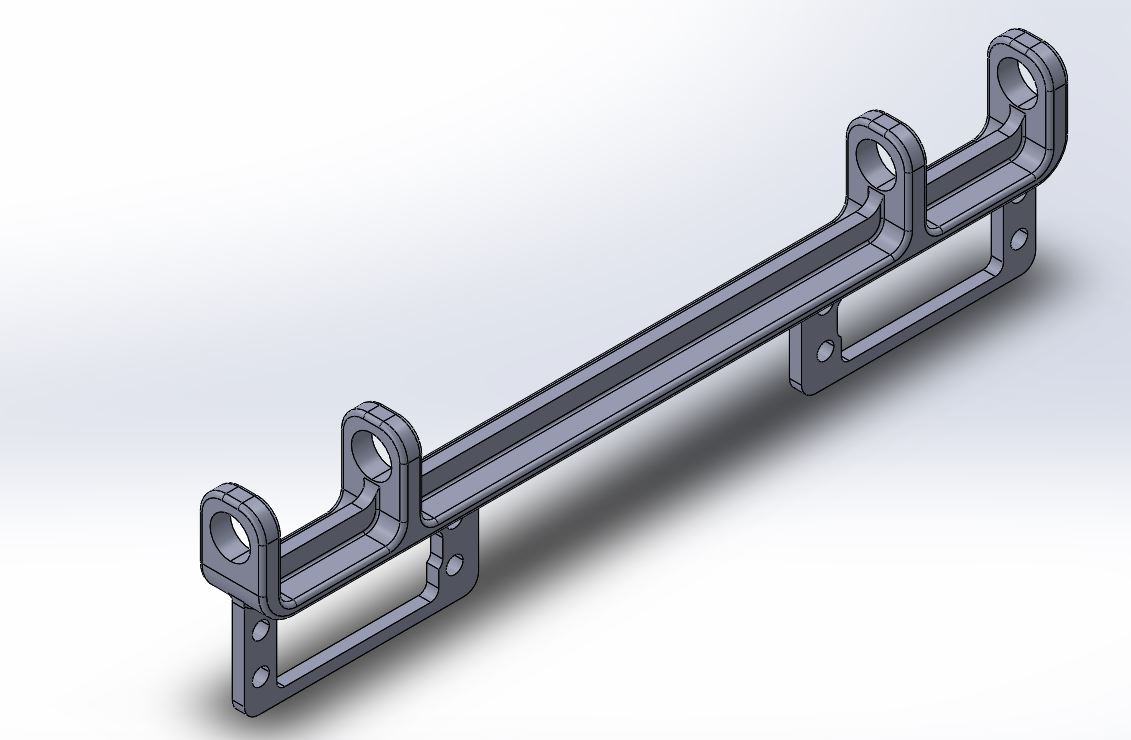
# 

# *Déroulement du projet*

## *Conception*

Le premier mois, nous avons défini notre stratégie de définition, d'étude et de réalisation, après réflexion nous nous sommes mis d’accord pour diviser le projet en 2 parties, l’électronique et la mécanique. Dans un premier temps nous avons rassemblé les informations nous permettant de comprendre le fonctionnement de ce système et pouvoir appréhender des solutions technologiques répondant aux contraintes.

Après avoir défini les mouvements nécessaires à l’orientation du bicoptère dans l’espace, nous avons dessiné la partie du châssis permettant la rotation des moteurs. Le premier schéma s’est fait sur papier puis sur SolidWorks pour permettre la réalisation sur un centre d’usinage numérique 3axes.



## *Réalisation*

### ***Electronique :***

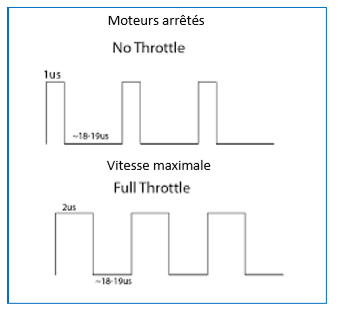
1. *Moteurs et servo-moteurs :*

Info PWM (Pulse Width Modulation) : Le PWM utilisé doit avoir une fréquence de 50Hz et c’est la durée du temps haut qui permet de coder la vitesse de rotation des moteurs :

- Pour un temps haut de 1000ms les moteurs seront éteints.

- Pour un temps haut de 2000ms les moteurs seront en vitesse max.

Les signaux PWM de 50Hz sont très utilisés en modélisme pour le contrôle de tout type de moteur. En Arduino, la librairie Servo.h permet de générer ces signaux.



Nous avons commencé par coder les moteurs brushless et les servo-moteurs mis à notre disposition. Grâce aux éléments de cours ainsi que nos recherches sur internet, nous avons rapidement maitrisé cette technique.

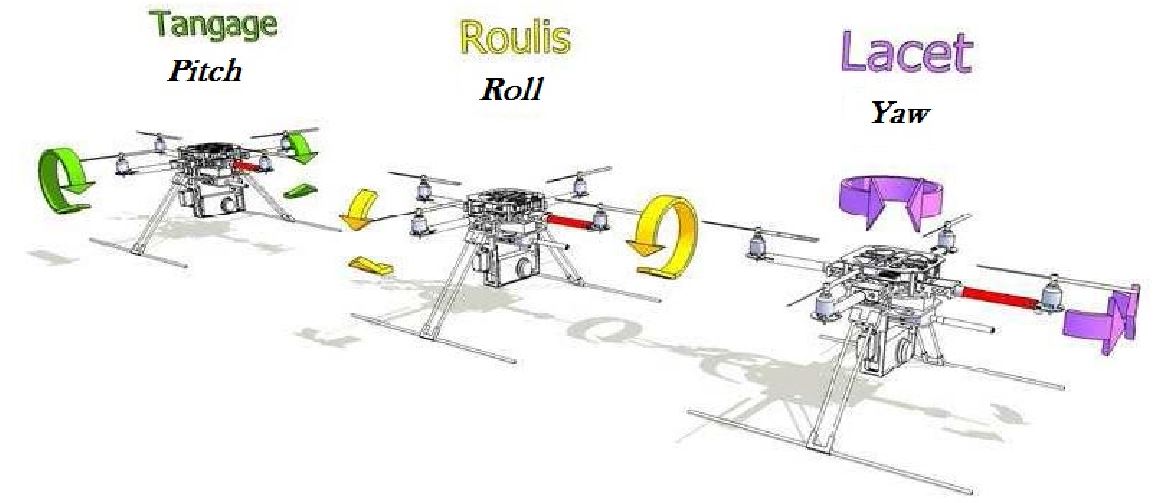
On a ensuite ajouté le module Bluetooth afin de permettre le contrôle à distance de chaque composant.

Une fois les codes mis en commun, nous avons rencontré un premier problème : les servo-moteurs et les moteurs utilisant la même librairie, on ne pouvait contrôler qu’un seul élément à la fois.

Grâce à la documentation Arduino, nous avons compris que cela venait des ports de sortie. Il suffisait juste de brancher les composants sur des ports bien choisis.

1. *Asservissement :*

Le PID est un système de contrôle permettant la stabilisation de notre bicoptère. Celui-ci agit en fonction des valeurs d’angle que l’accéléromètre et le magnétomètre peuvent fournir.



Dans notre cas, l’axe x correspond au Roulis, l’axe y au Tangage et l’axe z au Lacet

La stabilisation doit se faire suivant l’axe x, y et z. Pour ce qui est des axes x et y nous avons utilisé l’accéléromètre. Celui nous permet d’avoir l’angle (entre 0 et 90°) que fait le bicoptère avec le sol suivant ces 2 axes. Ces informations sont ensuite envoyées au PID et celui-ci va permettre la stabilisation, grâce à trois asservissements :

• Proportionnel

• Intégral

• Dérivé

L’asservissement proportionnel est le plus important, les autres asservissements permettent de gérer les erreurs afin que la stabilisation soit optimale.

Dans notre cas, pour l’axe x, le PID va agir sur les moteurs. En effet, lorsque le bicoptère va pencher vers la gauche, le moteur de gauche va accélérer en conséquence afin de le remettre droit, et inversement.

Pour l’axe y, le PID agit sur les servo-moteurs. Lorsque le bicoptère penche vers l’avant, les servos s’orientent vers l’arrière, ce qui permet à ce dernier de rester parallèle au sol.

La stabilisation sur ces 2 axes a été correctement codée.

Pour la stabilisation suivant z, le PID va recevoir un angle du magnétomètre et non de l’accéléromètre (celui-ci ne permet pas d’avoir la valeur de l’angle du lacet).

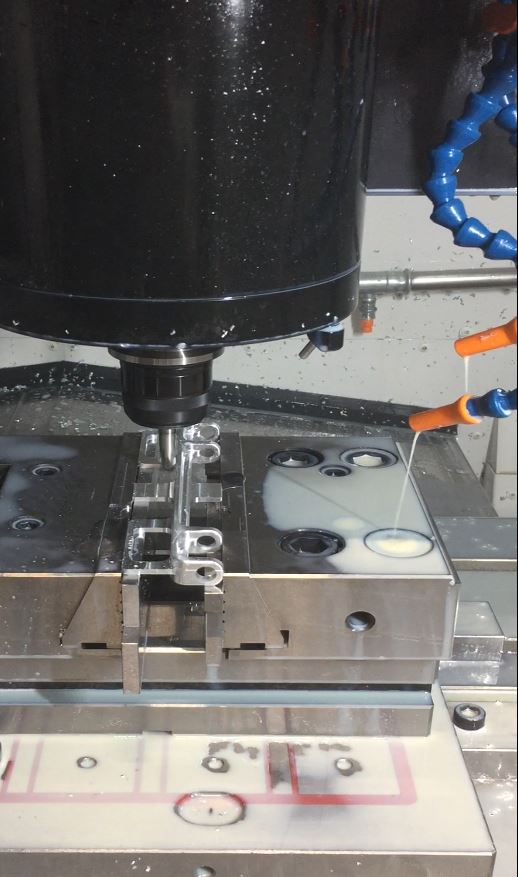
Le magnétomètre est en réalité une boussole, le zéro correspond au Nord puis en fonction de comment on tourne notre magnétomètre, l’angle varie entre 0 et 360°.

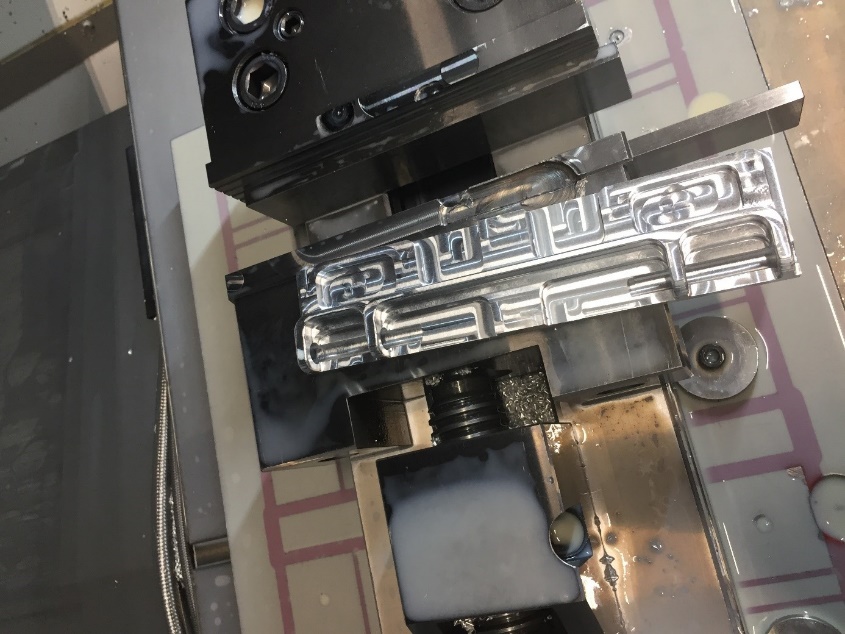
Pour cet axe z, le PID va agir sur les servos encore une fois mais dans ce cas, il les orientera dans le sens opposé, afin que la rotation soit compensée et que le bicoptère ne tourne pas sur lui-même.

La stabilisation suivant z n’a pas pu se faire car cela devenait trop compliqué pour nous.

### ***Mécanique :***

Après avoir dessiné sur le logiciel de CAO (Conception assistée par Ordinateur ici SolidWorks) la partie du châssis à usiner dans un bloc d’aluminium il faut maintenant ouvrir ce document dans un logiciel de FAO (Fabrication Assistée par Ordinateur ici TopSolid) afin de générer un parcours d’outils pour chaque phase de l’usinage (Ebauche, Semi-finition, Finition) qui sera envoyé au centre d’usinage à commande numérique 3axes.





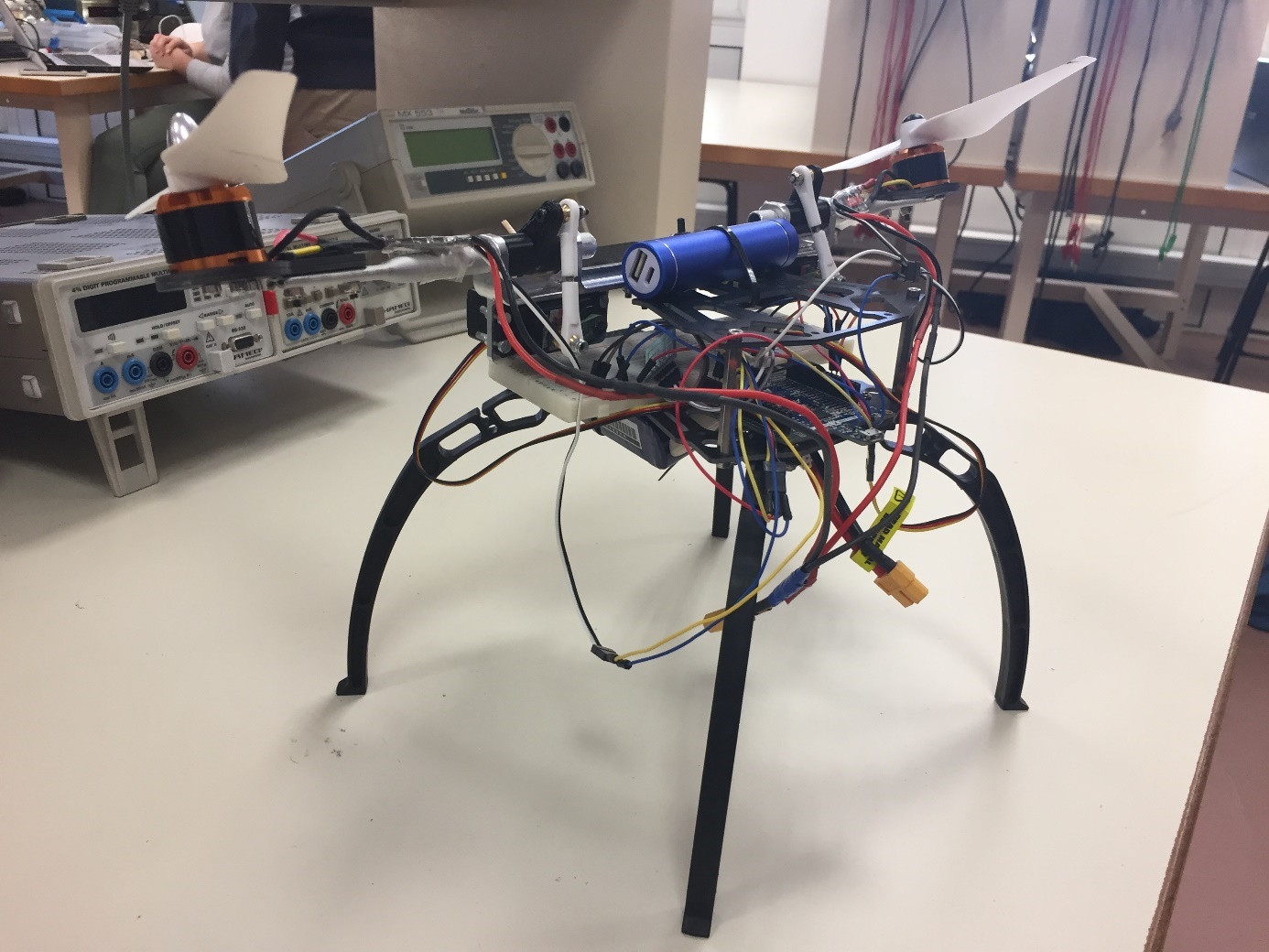
Nous avons dans le même temps dessiné les palonniers des arbres des moteurs brushless qui permettent par un montage en chappe, d'une rotule et d’un servo-moteur l’orientation de ces moteurs.

Ensuite nous avons sollicité le Fablab et leur imprimante 3D pour réaliser ces derniers puis notre professeur nous a fourni un squelette de drone que nous avons adapté à la partie déjà usinée.



# *Mise en commun*

Une fois l'assemblage mécanique réalisé, nous avons intégré l’électronique, puis paramétrer l’amplitude d’action des servo-moteurs pour ne pas endommager les liaisons mécaniques. Dans la mesure où notre projet est soumis à la gravité, nous avons équilibré les composants pour que le centre de gravité soit le plus proche de l’axe des moteurs~~.~~



# *Problèmes rencontrés*

## *Problème lié au Bluetooth :*

Nous avons contrôlé le drone avec un module Bluetooth connecté à un téléphone. Cette connexion n’était pas assez rapide et instable, en effet les modules Bluetooth peuvent changer de canal jusqu’à 1600 fois par seconde. Le drone devenait ainsi incontrôlable et difficile à commander. Les composants électroniques étaient mal gérés ce qui provoquait beaucoup de saccades.

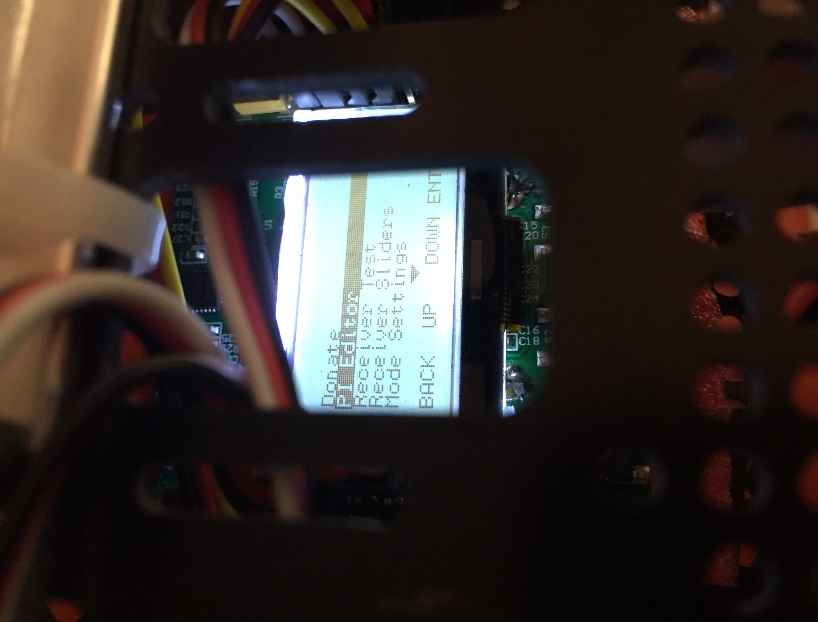
Pour régler ce problème, nous avons donc opté pour une radiocommande de modélisme, ce qui nous a permis d’avoir une connexion adaptée aux besoins qu'exige notre système. Après avoir développé un code dont nous avons trouvé la base sur internet nous sommes parvenus à mettre au point notre ensemble.

Finalement notre code Arduino comporte 2 fonctions principales :

* La boucle comprenant le PID
* Une fonction permettant le contrôle de la télécommande

## *Problème lié à la stabilisation :*

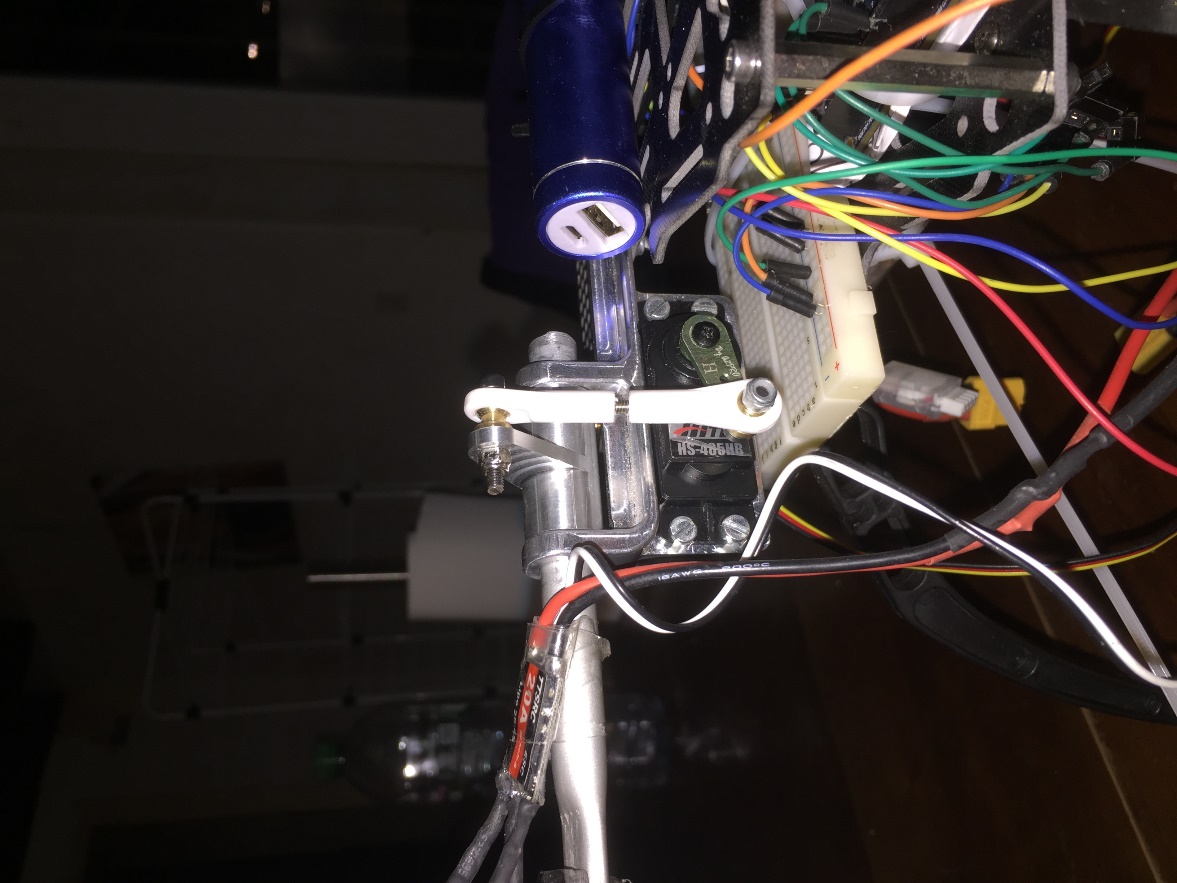
Nous n’avons pas réussi à régler le problème de la stabilisation suivant z (Lacet). Après de nombreuses recherches, nous avons décidé de changer de carte, avec l’accord de notre professeur. Cette carte, la Hobbyking KK2.1.5, nous permet, en la paramétrant complètement, de gérer la stabilisation suivant les 3 axes.



Une fois la carte en configuration d’usine il faut préciser quelles sont les entrées et sorties utilisées. Puis définir notre système (pour nous ce sera donc un bicoptère) et régler tous les paramètres c’est-à-dire les constantes du PID (cette carte ne permet de régler que la proportionnelle et l’intégrateur ce qui s’avèrera suffisant par la suite). Enfin il faut réaliser de très nombreux tests de vol afin d’affiner les réglages du PID et de prendre notre système en main (ce qui n’est pas évident).

## 

## *Problème lié au jeu dans la liaison mécanique :*

Les palonniers des arbres moteurs imprimés au Fablab ont une précision assez mauvaise et cela a engendre un jeu important dans l’orientation des brushless, rendant le bicoptère très instable. Nous avons donc réalisé ces palonniers au centre d’usinage dans un bloc d’aluminium pour avoir une précision bien meilleure et donc un jeu quasiment nul. Nous avons dessiné les palonniers sur le logiciel de CAO et en passant par un logiciel de FAO puis nous avons réalisé ces pièces sur le centre d’usinage 5axes.

Et voici le résultat final de la liaison mécanique :

# *Conclusion*

Ce projet nous a permis d’apprendre beaucoup de choses dans différents domaines, et nous appris à être autonome.

Après 3 mois de travail, notre bicoptère remplit la fonction principale de notre cahier des charges qui est de voler stablement.

Certaines fonctions contraintes sont aussi vérifiées, la résistance à l’environnement à été prouvée car même après environ 30 chutes le système est toujours intact. La sécurité est suffisante car il faut faire une manipulation spéciale pour « armer » le système (maintenir les deux joysticks de la radiocommande dans le coin inférieur gauche pendant quelques secondes), de plus le système se désarme automatiquement après 20 secondes d’inactivité. Cependant le drone n’est pas autonome car les moteurs brushless ne sont pas assez puissants pour soulever le drone (670grammes) plus une batterie (330grammes), les vols ont été effectués grâce à une alimentation secteur.

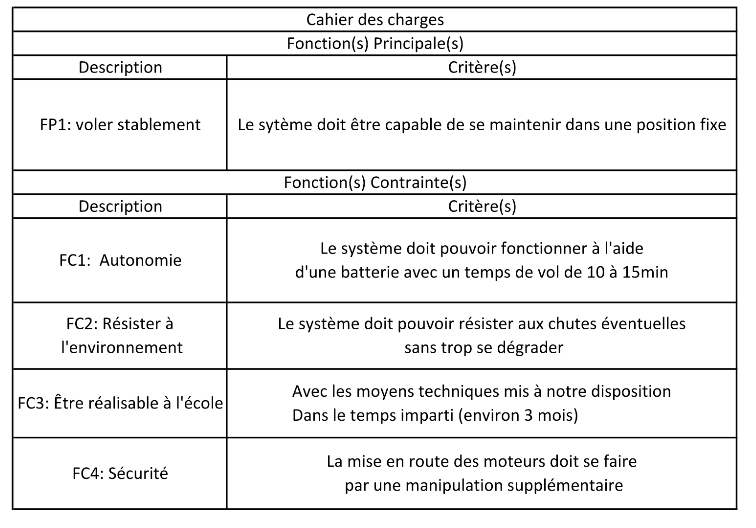
Nous sommes très satisfaits du résultat obtenu, bien qu’environ 200 heures ont été nécessaires à la réalisation de ce projet, il peut encore être perfectionné, au niveau de l’esthétique et de l’autonomie. Cela sera sûrement fait d’ici la fin de l’année scolaire.

Voici une vidéo du système final en vol : <https://www.youtube.com/watch?v=pWN4PxkNXiw&feature=youtu.be>

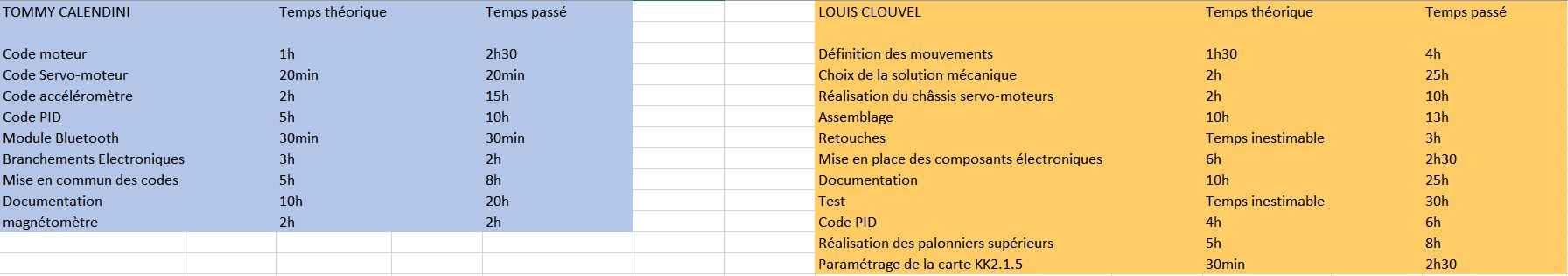
# 

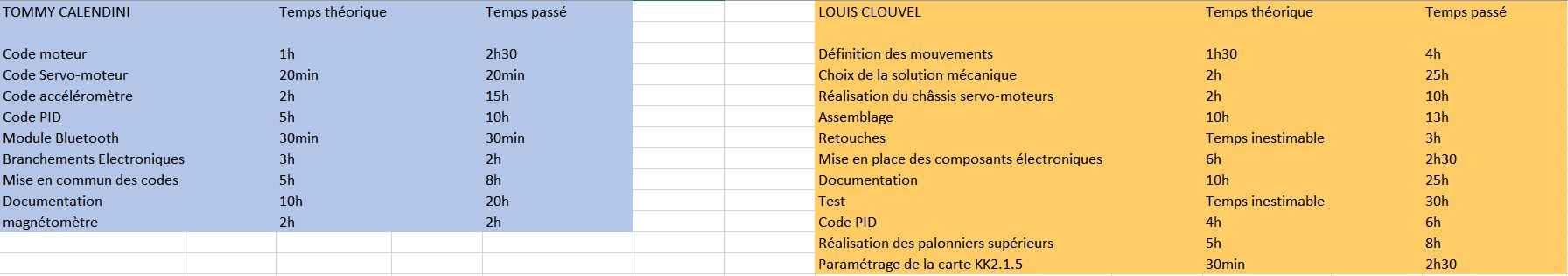
# *Annexes*

## *Cahier des charges*

**

## *Planning*





## *Bibliographie*

En ce qui concerne la documentation sur les fonctions et composants Arduino nous avons utilisé ce site : <https://www.arduino.cc/>

Lors du commencement du projet, ce forum nous a bien éclairé sur la conception du bicoptère : <https://www.helicomicro.com/2013/08/31/walkera-pandora-warrior/>

Le squelette du code de la radio commande nous a été expliqué et nous avons pu le récupérer sur ce forum : <https://www.firediy.fr/article/utiliser-sa-radiocommande-avec-un-arduino-drone-ch-6?fbclid=IwAR07gagBvPOGKWFAyB7cgFcOHnws4W7CAwKVZ1GRNn07RXQFWZP6fjG5DL8>

Pour ce qui est du choix de la carte électronique qui est maintenant sur le bicoptère ces deux pages web nous ont contribué au bon choix de carte :

<https://www.youtube.com/watch?v=jRPUO8T1Iwk>

<https://www.instructables.com/id/Bicopter-Dualcopter/>

Nous vous mettons le lien de notre chaine Youtube pour que vous puissiez votre l’avancement en vidéo de notre projet : <https://www.youtube.com/channel/UC59cZ8Wts5xyrMVUgzJZBDQ/videos?view_as=subscriber>