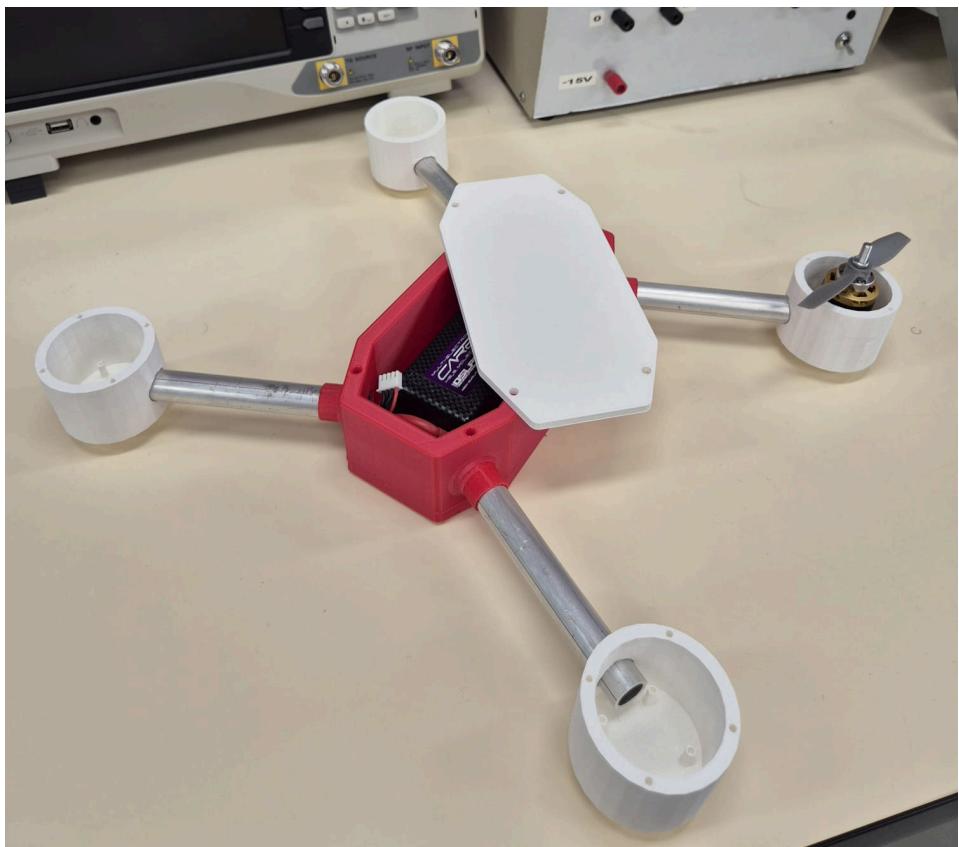


**Projet : Drone amphibie**



**Réalisés par :** LOISELET Geoffroy, MA David , KABASSI Yahya , ZUBAROGLU Benjamin

**Encadré par :** M. Kasbari

## **Sommaire :**

<b>-I) Conception du drone .....</b>	<b>p.4</b>
- a) Choix des composants et dimensionnement	
- b) Architecture électronique du Flight Controller	
<b>-II) La manette du drone.....</b>	<b>p.12</b>
- a) Architecture électronique de la manette	
- b) Programmation pour les différents composants	
<b>-III) Modélisation 3D .....</b>	<b>p.15</b>
- a) CAO du drone	
- b) CAO de la manette	
<b>-Conclusion.....</b>	<b>p.17</b>

## **Introduction:**

Notre projet consiste à réaliser un drone amphibie. C'est-à-dire un drone qui puisse se mouvoir à la fois dans l'air et dans l'eau. Ce projet comporte plusieurs problèmes : à la fois ceux d'une conception d'un drone classique, mais aussi d'un appareil qui doit rester étanche.

Notre réflexion de départ peut se décomposer en plusieurs étapes : réaliser un drone contrôlable dans l'air (un drone classique), puis les fonctionnalités de l'appareil sous l'eau.

Dans l'air, il s'agit d'un drone classique contrôlable avec une communication RF et avec une autonomie suffisante pour durer quelques minutes.

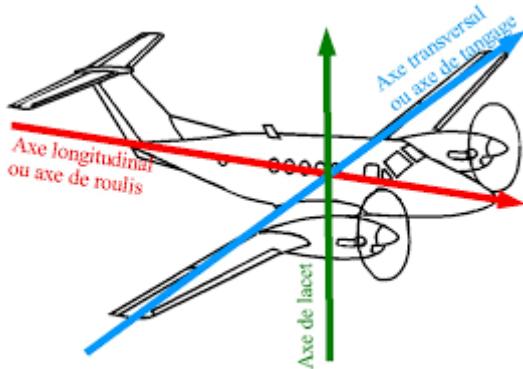
Dans l'eau, plusieurs contraintes s'ajoutent. En effet, il y a des contraintes sur les moteurs par rapport à la vitesse de rotation des moteurs, mais aussi par rapport à la communication RF puisque les ondes sont en majorité absorbées à la surface de l'eau. De là, plusieurs solutions ont été pensés :

- essayer un autre moyen de communication (par exemple onde acoustique), mais donc avoir un système à part qui puisse faire la liaison avec les ondes RF et l'autre moyen de communication;
- Ou programmer le drone de sorte à ce qu'il exécute des mouvements prévus lorsqu'il rentre dans l'eau puis ressort de l'eau tout seul.

Au vu des contraintes des 2 milieux, il faut choisir les composants pour qu'ils soient à la fois adaptés dans l'air et dans l'eau.

## **I) Conception du drone**

### **a) Choix des composants**



*Figure 1 : Axes de rotation d'un appareil aéronautiques*

## -Choix des moteurs :

**Motorisation : Technologie Brushless et KV adapté**

### **Choix de la technologie**

Nous avons opté pour des moteurs **BLDC (Brushless DC)**. Comparés aux moteurs à balais classiques, ils offrent un couple par poids supérieur, une fiabilité accrue (pas de frottements mécaniques des balais) et une meilleure efficacité énergétique. Ces caractéristiques sont indispensables pour compenser la surcharge liée à l'étanchéité du châssis et au mouvement sous-marin.

### **Le compromis Couple/Vitesse (KV)**

Le paramètre critique est le KV (constante de vitesse), qui détermine la vitesse de rotation pour un Volt donné **RPM = KV \* Tension**.

La contrainte : Dans l'eau, la résistance est forte. Selon la courbe Couple/Vitesse, si le couple demandé augmente, la vitesse doit diminuer pour conserver la puissance. Un moteur à trop haut KV (très rapide mais peu coupleux) risquerait de "caler" ou de surchauffer dans l'eau.

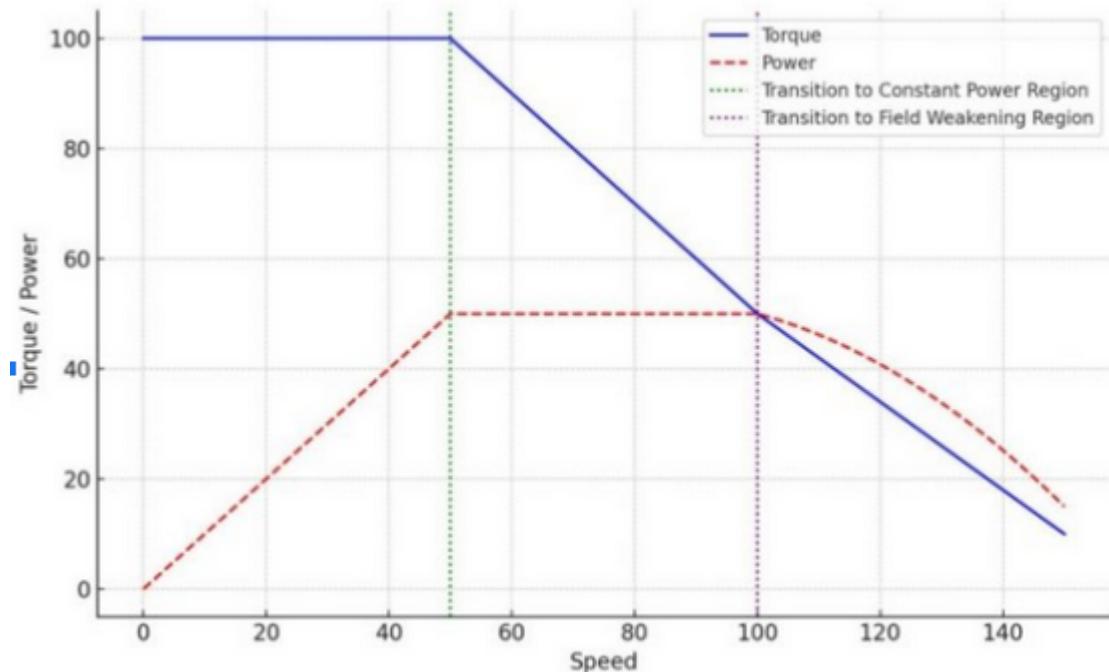


Figure 2 : Tracé du couple/puissance en fonction de la vitesse

Source: <https://mechtx.com/blog/understanding-torque-speed-curves-of-bldc-motors>

Notre choix s'est porté sur des moteurs de **1000~1500 KV**.  
Le moteur choisi est : **Extron 2212/20 - 65g - 1300Kv - 180W.**



*Figure 3 : Photo du moteur utilisé*

Ce choix permet une vitesse de rotation théorique d'environ 14430 tr/min sous 11.1V (Batterie 3S).

C'est un point d'équilibre optimal : assez rapide pour le vol aérien, mais offrant suffisamment de couple pour tourner dans l'eau sans saturer le contrôleur.

### -Choix de batterie :

Le choix de la batterie est primordial dans la conception de notre drone car il définit la durée de vol mais pose des contraintes de masse.

La batterie doit fournir un courant de décharge supérieur à la somme des courants des 4 moteurs. L'autonomie de vol est directement liée à la capacité choisie selon la formule :

$$Temps = \frac{\text{capacité}}{\text{consommation}} \times 60$$



*Figure 4 : Photo de la batterie (3s) utilisé dans le drone*

Notre choix final s'est porté sur une batterie LiPo 3S 3800mAh.

**Ci-dessous les détails de nos calculs :**

PFD (projection sur l'axe de lacet) :  $P = mg = 9,81 \text{ N}$   
Ainsi, chaque moteur doit pousser :  $9,81/4 = 2,45 \text{ N}$ .

Pour atteindre une autonomie de vol de **10 minutes**, objectif cohérent avec l'usage de notre drone, ainsi que pour pouvoir porter **1 kg** de charge totale, chaque drone porte donc 250g, soit 30% de leur maximum d'après la datasheet ou 4,5A par moteur.

Courant total  $I_{\text{tot}} = 4 \times 4,5 = 18 \text{ A}$ .

Capacité réelle de la batterie (80%) :  $C_{\text{réel}} = C_{\text{max}} \times 0,8 = 3040 \text{ mAh}$ .  
**Autonomie** =  $I_{\text{tot}}/C_{\text{réel}} = 0,169 \text{ h}$  ou **10 minutes**.

### -Capteurs :

Quelles informations utiles pour le vol d'un drone, et comment les capter ?

Afin de pouvoir faire voler le drone dans l'espace entier 3D, le choix d'un module accéléromètre+gyroscope, le MPU6050 a été fait.

Ce choix se base sur les disponibilités de l'ENSEA, permettant de limiter les dépassements de coûts du drone, ainsi que son utilisation très répandue dans la fabrication de drones.



*Figure 5 : Capteur accéléromètre MPU6050*

Caractéristiques du MPU6050 : Capteur IMU 6 axes (il renvoie donc 6 valeurs) :

- accéléromètre : mesurer l'accélération linéaire axes X,Y,Z (permet de détecter l'orientation par rapport à la gravité)
- gyroscope : mesure vitesse angulaire X,Y,Z (sert à détecter les rotations et stabiliser le drone)

Moyen de communication par I<sup>2</sup>C

## -Communication : NRF24 :

Le choix du moyen de communication est fondamental dans notre projet : notre système doit pouvoir s'éloigner du pilote, par conséquent, une communication RF 2,4GHz semble adaptée.

On a donc choisi le module nrf24l01 car il permet cette communication RF, de plus, ce module est souvent associé à ce type d'usage ce qui en fait un outil adapté et facile d'accès pour notre projet. Il joue aussi bien le rôle d'émetteur que de récepteur, et dans le cas du drone, on l'utilise comme récepteur.

L'absorption du signal de commande dans l'eau n'affecte pas le drone car il est prévu de réaliser un processus de navigation dédié à l'eau sans communication avec l'extérieur.

Fonctionnement du module :

La conception d'un drone amphibie impose des contraintes sévères sur la chaîne de propulsion. Contrairement à un drone classique optimisé uniquement pour l'air, notre système doit fournir une poussée efficace dans deux fluides aux densités radicalement différentes.

## -Hélices :

### **Hélices : Le choix du profil 7x5**



*Figure 6 : Photo des hélices*

Les hélices sont en contact direct avec le fluide. Nous avons retenu un modèle de dimension 7x5 (7 pouces de diamètre, 5 pouces de pas).

**Diamètre** (7 pouces / 17.8 cm) : Un diamètre suffisant pour brasser un volume d'air important et générer la poussée nécessaire au décollage vertical (Poussée > Poids).

**Pas** (5 pouces / 12.7 cm) : Le pas représente la distance parcourue par l'hélice en un tour. Un pas de 5 est considéré comme modéré/faible.

Justification Amphibie : Un pas faible favorise la traction (couple) plutôt que la vitesse pure. C'est idéal pour l'eau, car une hélice à fort pas offrirait trop de résistance hydrodynamique.

Bilan : L'association Moteur 1000 KV + Hélice 7x5 est une configuration éprouvée ("Long Range") qui garantit une excellente efficacité énergétique et évite la surchauffe des composants lors des phases de transition.

## Contrôleurs de Vitesse (ESC) et Alimentation

### **Dimensionnement des ESC**



*Figure 7 : Photo de l'ESC pour le moteur brushless*

Pour piloter ces moteurs, les ESC (Electronic Speed Controllers) ont été choisis pour supporter les pics de courant, notamment lors de l'immersion brutale où la résistance augmente instantanément. Nous appliquons un coefficient de sécurité de 1.2 à 1.5 par rapport à la consommation maximale du moteur.

Critère :

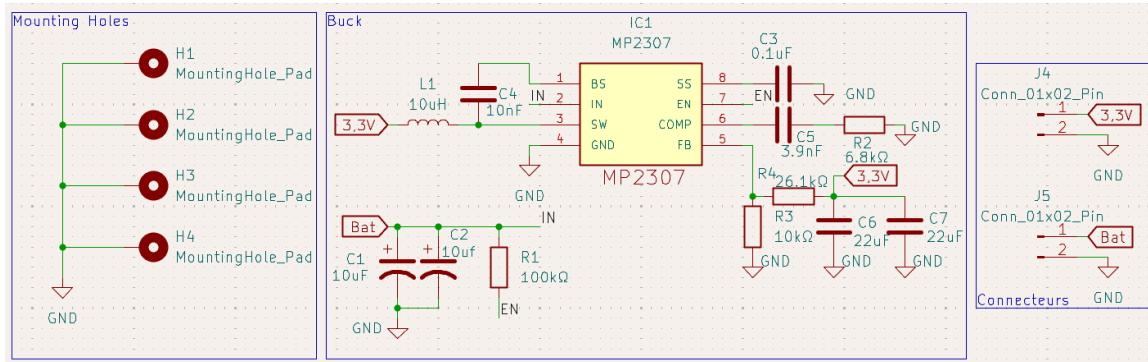
$$I_{esc} > 1,2 \times I_{moteur\ max}$$

Selon les spécifications techniques du moteur, le courant maximal est 18 A , donc  $I_{esc}$  doit être supérieure à 21.6, on a adopté pour un ESC de 30A.

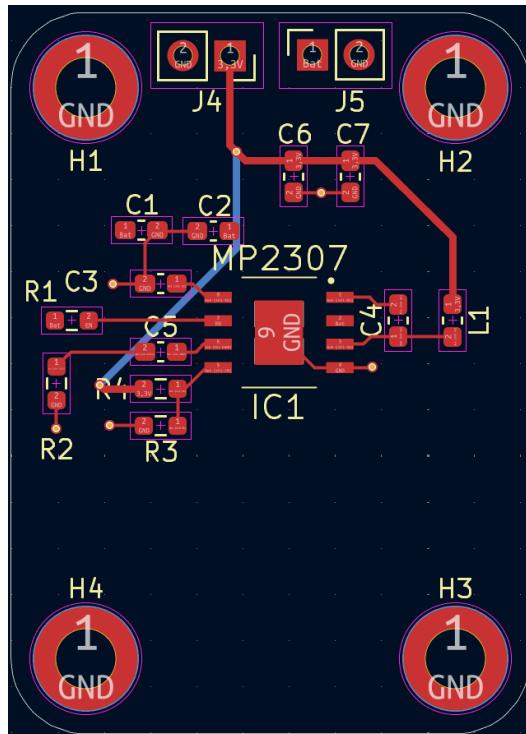
### PCBs Drone :

Pour le PCB du drone, nous avons choisi d'en faire deux, une carte pour la partie puissance, et une autre pour la partie commande. Ce choix a été fait afin d'optimiser la place, et pour pouvoir séparer les "hautes tensions" du drone de la partie qui fonctionne en basse tension. Ainsi, si nous avons un problème sur la partie puissance, qui est plus susceptible que d'en avoir que la carte de commande, nous n'avons qu'une carte plus simple à faire plutôt que l'ensemble des PCB du drone. Afin d'optimiser l'espace au maximum, nous avons décidé de faire les deux cartes de même taille pour pouvoir les placer l'une au-dessus de l'autre avec une petite séparation.

Nous avons vu précédemment que nous avons des batteries délivrant 11,1 V, ce qui est nécessaire pour les moteurs, mais qui n'est pas adapté pour le microcontrôleur. Pour cela, nous avons donc dû prendre un BUCK (MP2307), qui abaisse la tension à 3,3 V (conformément à la documentation du composant). Voici le schéma électrique et le routage de cette première carte :



*Figure 8 : Schéma électrique de la carte puissance du drone*



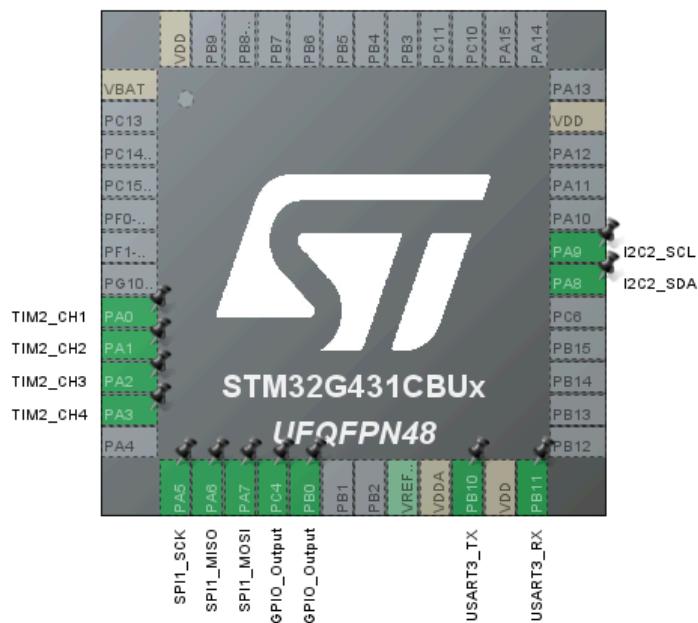
*Figure 9 : Routage de la carte puissance du drone*

La carte de commande du drone est le cerveau principal de notre projet, c'est là où le donne recevra les commandes de la manette, qu'elle les traitera, qu'elle fera toutes les mesures des capteurs pour l'asservissement, les calculs pour ce dernier ainsi que les commandes des moteurs.

Nous devons donc faire une carte assez complexes, sur laquelle nous allons mettre les composants suivants :

- Microcontrôleur STM32G431CBU6
  - Gyroscope + Accéléromètre MPU6050
  - Module RF NRF24
  - STLink

Nous avons choisi ce microcontrôleur puisqu'il possède une bonne puissance de calcul pour notre projet, assez de timer pour nos 4 PWM (une par moteur) ainsi que les communications SPI et I2C dont nous avons besoin. Voici les branchements que nous avons fait pour tester la compatibilité du modèle, ainsi que le schéma électrique et le routage de la carte de commande :



*Figure 10 : Test branchements du STM32*

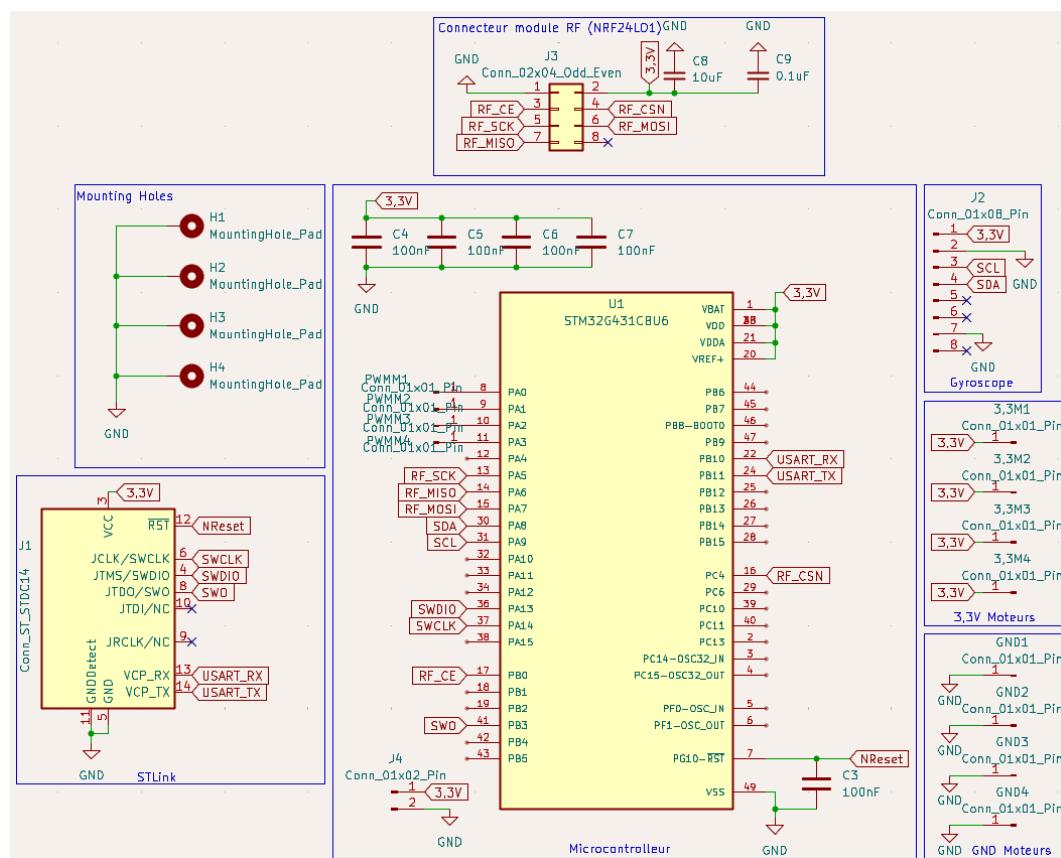


Figure 11 : Schéma électrique de la carte commande du drone

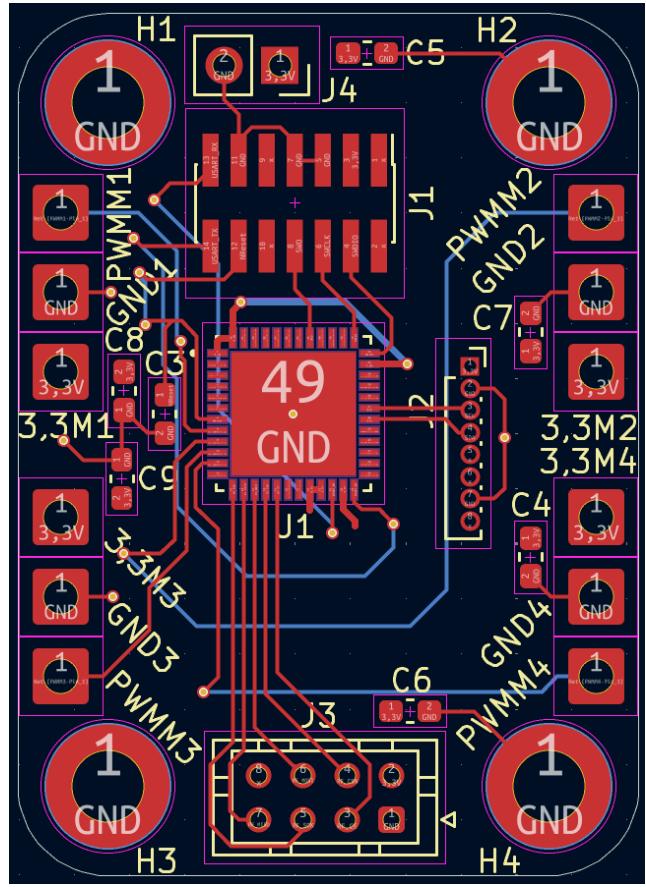


Figure 12 : Routage de la carte commande du drone

Sur notre carte de commande, nous pouvons notamment retrouver les branchements de chaque moteur. En effet, nous pouvons observer sur les côtés des cartes des branchements appelés PWMMX, GNDX et 3,3MX avec X correspondant au numéro d'un moteur. Chaque moteur doit être branché à une PWM, au GND et au 3,3 V, c'est la raison pour laquelle cela a été disposé de cette manière.

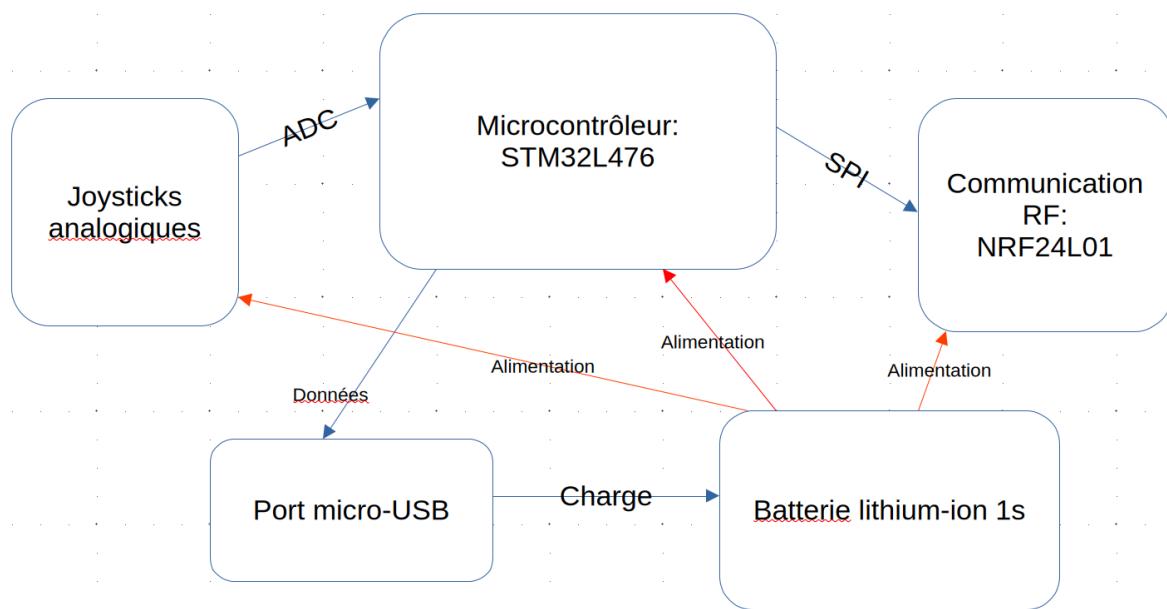
## **II) La manette du drone**

Dans cette partie, on traitera de la conception de la manette de drone.

### *a. Architecture électronique de la manette*

Tout d'abord, le but de cette conception était de créer une manette simple d'utilisation et adaptée à notre usage. En effet seules les fonctionnalités standards nous suffisaient et les manettes en général sont truffées de fonctionnalités dont nous n'avons pas besoin dans notre cas.

Pour le choix des composants, nous avons privilégié des composants faciles d'accès et des circuits courts (en général trouvables à l'ENSEA).



*Figure 13: Organigramme de la conception de la manette*

Dans un premier temps, on va s'intéresser aux microcontrôleurs. On utilise un **STM32L046** car c'est celle que nous utilisons pour la carte de développement, donc en termes de facilité il est plus simple de garder ce microcontrôleur. De plus, il fait partie de la gamme L des STM32 donc peu énergivore, et possède beaucoup de fonctionnalités dont on se servira notamment plus tard. Pour programmer le STM32, nous utilisons un **ST-LINK** que nous avons soigneusement routé.



Pour la partie communication, nous reprenons le **NRF24L01** qui est donc cette fois-ci utilisé en émetteur. Pour communiquer avec ce composant, il faut utiliser un protocole SPI. Sur le PCB, il est prévu un connecteur pour le connecter aux STM32.

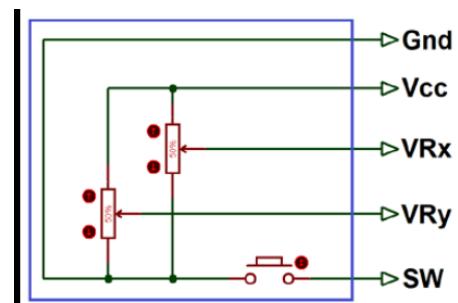
*Figure 14: Module NRF24L01 avec antenne*

Pour commander le drone, on utilise **2 joysticks analogiques**. On veut dans notre cas reproduire la manière standard pour contrôler un drone décrit précédemment. Le joystick analogique renvoie deux valeurs de tension **Vx** et **Vy**. En effet, en bougeant le stick dans une direction, on modifie la valeur de résistance d'un potentiomètre ce qui permet de faire varier sa tension en sortie. On lit ces valeurs de tensions avec l'ADC du STM32.

De plus ce joystick dispose d'un bouton que l'on pourra utiliser pour une fonctionnalité future.



*Figure 15: Joystick analogique*



*Figure 16: Schéma fonctionnement joystick analogique*

Ensuite la partie alimentation, on utilise une **batterie lithium-ion 1s** (3.7V), 600mAh ce qui est largement suffisant pour notre utilisation. On utilise le composant **TP4057** pour la gestion de la charge de la batterie. La charge notamment se fait par un port **micro-USB**, qui servira

plus tard pour le débogage de la manette (on compte utiliser la manette comme une vraie manette sur un simulateur de drone, pour pouvoir la tester).

Après, la majorité des composants utilisent une alimentation 3.3V (STM32,NRF24L01), donc on utilise un **BUCK-BOOST (TPS63060DSCR)** pour convertir le 3.7V de la batterie en 3.3V. L'utilisation d'un BUCK-BOOST sert à garder un 3.3V constant, car avec un LDO on aurait pas eu 3.3V constant, en fonction de si la batterie est chargée ou déchargée.

Finalement, on place quelques **boutons poussoires** pour différentes fonctionnalités futures qui seraient imprévues.

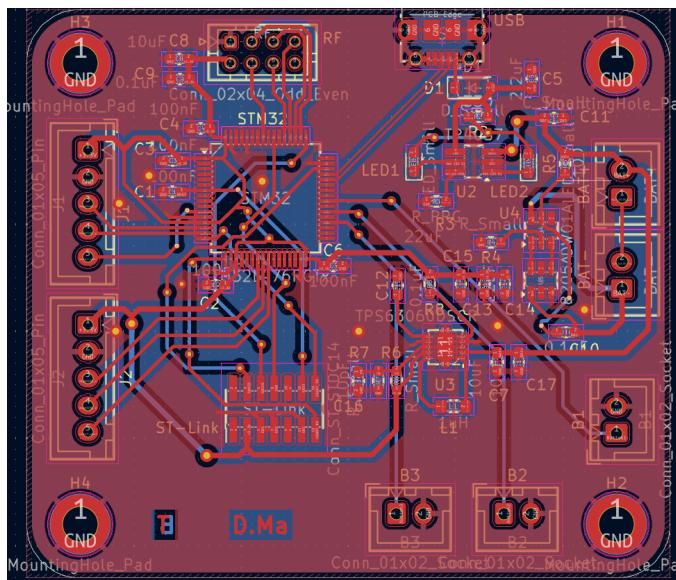


Figure 16: PCB de la manette sur Kicad

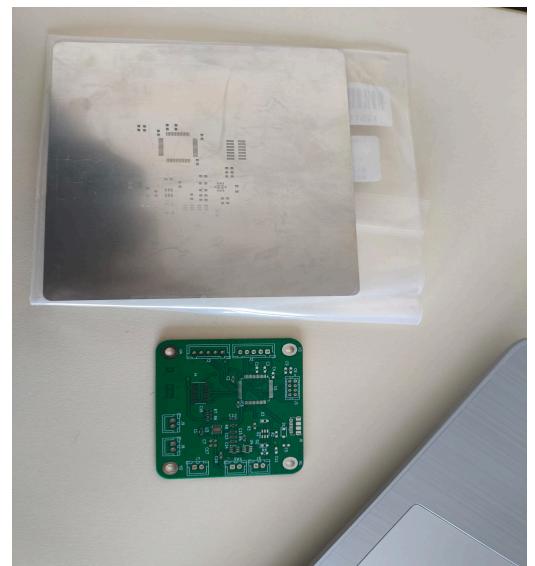


Figure 17: PCB de la manette et son stencil

Nous avons reçu le PCB mais pas encore certains composants. Donc malheureusement, à cette étape du projet, nous n'avons pas encore pu souder la carte ni tester la carte.

#### *b. Programmation pour les composants*

En attente des composants, nous avons essayé d'abord de programmer avec les composants que l'on a déjà.

Pour le **joystick analogique**, on lit les coordonnées x et y avec les tensions Vx et Vy. On utilise donc l'ADC du STM32 pour pouvoir interagir avec ce composant.

Lors des tests, nous avons constaté que le joystick que l'on utilisait avec un léger décalage par rapport à sa position de départ. C'est-à-dire que sa position de repos est décalé du centre (toute à droite).

Pour le module **NRF24L01**, on a démarré la phase de test de configuration en commençant l'écriture de la librairie en C.

Toute cette partie a été réalisée en attente des composants à cette étape du projet. Cette partie programmation va très sûrement être délaissé au profit du soudage du PCB lors de la réception des composants.

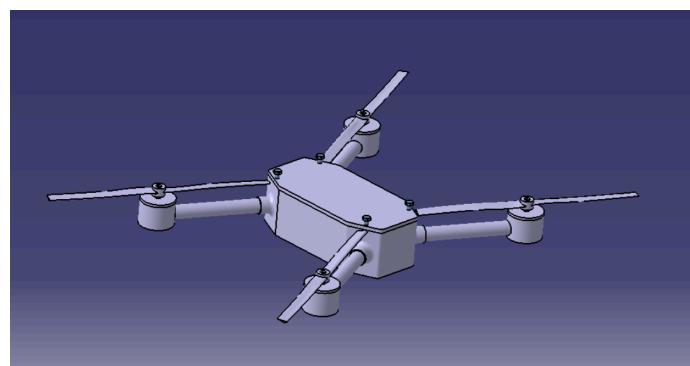
### **III)Modélisation 3D :**

#### *a.CAO du drone*

Pour l'architecture du drone, nous avons choisi une configuration de **châssis en "X"**. C'est une structure qui a fait ses preuves : elle offre le meilleur compromis entre stabilité en vol et maniabilité.

Concernant la fabrication, le bloc central ainsi que les quatre boîtiers moteurs ont été réalisés en **impression 3D (ABS)**. Nous avons choisi l'ABS pour sa solidité, mais surtout pour sa capacité à créer des pièces hermétiques : les boîtiers aux extrémités ne sont pas juste des supports, ils assurent **l'étanchéité** nécessaire pour protéger les connexions des moteurs en milieu aquatique.

Enfin, pour relier le tout, nous avons utilisé des **tubes en aluminium**. Ce choix s'explique par la rigidité du métal, qui empêche les bras de vibrer ou de plier, tout en restant suffisamment léger pour ne pas pénaliser l'autonomie.



*Figure 18:* Modèle 3D du drone

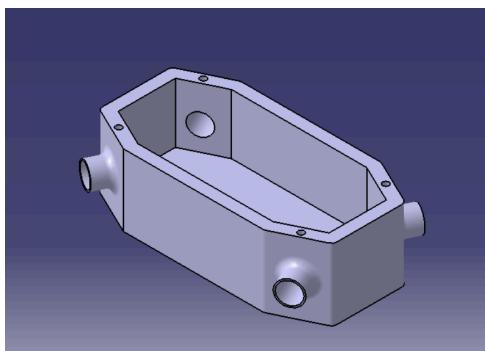


Figure 18: Bloc centrale du drone

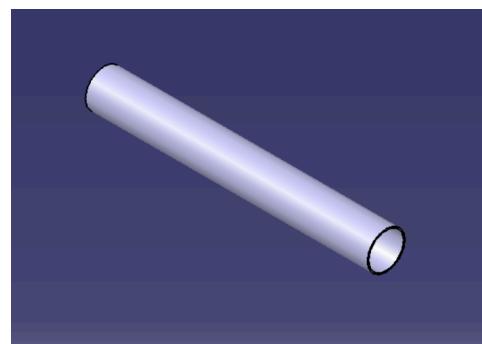


Figure 19: Tube du drone

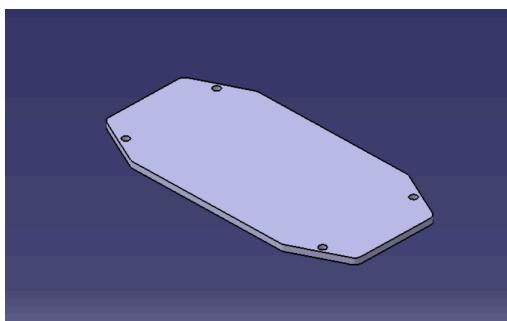


Figure 20: Couvercle du drone

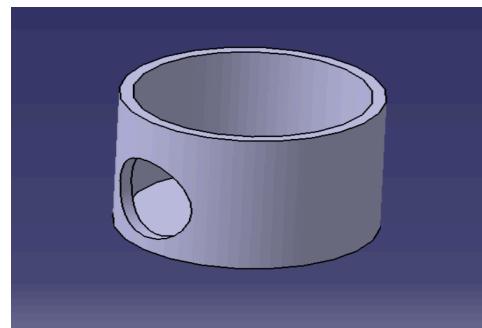


Figure 21: Boîtier moteur du drone

#### b. CAO de la manette

Pour piloter le drone, nous avons également conçu et modélisé notre propre **manette de commande**.

Contrairement au châssis du drone, le boîtier de la manette est imprimé en **PLA**. Comme la manette reste au sec, nous n'avions pas les mêmes contraintes thermiques ou d'étanchéité que pour le drone. Le PLA offre une rigidité parfaite pour la prise en main et permet une impression rapide et propre.

La modélisation a été pensée pour être fonctionnelle et ergonomique : elle intègre parfaitement les deux joysticks de contrôle, les boutons d'action, et prévoit un emplacement dégagé pour l'antenne externe afin de garantir une transmission radio optimale.



Figure 21: Modèle 3D de la manette

## **Conclusion**

Jusqu'à cette étape du projet, nous avons plusieurs éléments du drone tels que les modèles 3D du drone et du boîtier de la manette. Il nous manque toujours certains composants pour souder les PCB et donc pouvoir commencer la phase de test.

Pour les étapes d'après il faudra faire les tests notamment pour le vol du drone, la puissance des moteurs, l'ergonomie du châssis, et surtout envisager d'autre solutions notamment en termes d'étanchéité sous l'eau.

Plusieurs pistes sont en cours de réflexion comme mettre du vernis pour les cartes électroniques, mettre un isolant (joints) autour des points sensibles qui pourraient être susceptible de faire pénétrer l'eau.