

**ECOLE NATIONALE  
SUPERIEURE D'INGENIEURS**  
**LP CTT GM S6**



**UNIVERSITE DE LOME**

**BP : 1515 Lomé-TOGO**

**Travail-Liberté-Patrie**

**Septembre 2021**



## **ETUDE DE LA CONCEPTION D'UN DRONE AGRICOLE (PFE)**

### **Réalisé par :**

SEKPONA Kokou Sitsope

SENAYA Serge

SOKPO Mathias

LEGUEZIM Kossel

### **Dirigé par :**

Dr AFIO

Dr DROVOU

## *REMERCIEMENTS*

Nous tenons tout d' abord à remercier Dieu pour la grâce, la santé et les biens faits qu' il nous accorde au quotidien. Nous lui devons toutes nos sources d' inspiration pour l' accomplissement de ce projet.

Nous adressons toute notre reconnaissance aux Directeurs de Projet Dr AFIO et Dr DROVOU pour leurs patiences, leurs disponibilités et surtout leurs judicieux conseils, qui ont contribué à alimenter nos réflexions

Nous désirons également remercier les professeurs du département de Génie mécanique, qui nous ont fourni les outils nécessaires à la réussite de ce projet

En guise de reconnaissance, nous témoignons toutes notre gratitude aux personnes qui ont contribués de près ou de loin au bon déroulement de ce projet de spécialité et à l' élaboration de ce modeste travail.

## *RESUME*

Ce document est une étude détaillée de la conception d' un drone agricole faite pour pulvériser les grandes surfaces cultivables. Il nous revient dans un premier temps d' établir le cahier de charge ensuite d' étudier en dynamique et en cinématique le drone dans ses mouvements de vol puis d' étudier els éléments et la programmation des microcontrôleurs qui contrôle le mécanisme de vol. Y serra inclus également le calcul des surfaces pulvérisés pendant son vol et le cout total de la conception afin de comparer ce dernier aux systèmes manuels d' arrosage. Une modélisation 3D et des vues de mises en plan seront réalisés pour une meilleure compréhension du système.

## Table des matières

I. Introduction .....	5
ETUDE DU MARCHE .....	6
1. Les clients potentiels : .....	7
2. Les cibles visées : .....	7
III. CAHIER DES CHARGES .....	7
Énoncé du besoin .....	7
Mise en situation .....	7
IV. Dynamique du drone .....	11
1. Dimensionnement des hélices .....	11
2. Les différents mouvements : .....	14
3. Calcul de l'angle d'inclinaison théorique .....	18
4. Consommation en puissance du microcontrôleur (Arduino) .....	20
5. Calcul de la surface pulvérisé .....	22
6. Force et Moments .....	22
7. Vecteur d'état et ses dérivés .....	32

8.	Accélération linéaire .....	34
9.	Accélération angulaire .....	35
V.	Electronique .....	38
	Description du hardware utilis : é .....	38
1.	Arduino Uno : .....	39
2.	Gyroscope MPU-6050 : .....	40
3.	Electronique Speed Controller « ESC » : .....	44
4.	Batterie LIPO : .....	45
5.	Les hélices : .....	46
6.	Arduino Nano : .....	48
7.	Module radio NRF24L01 2.4GHz : .....	48
8.	Les joysticks : .....	50
9.	Code: .....	52

## I. Introduction

L’agriculture a toujours été une ressource indispensable au développement de tout pays. Et elle doit croître avec la population pour arriver à pallier à ses besoins alimentaires. Or cette dernière ne cesse de croître. Pour suivre ce rythme, cette agriculture doit augmenter sa production donc ses plantations et donc sa surface cultivable. Bref se mécaniser.

Cette solution a été adopté par des pays développé depuis les années 1900 surtout les grandes puissances qui sont arrivé ainsi à s’ autosuffire et à produire pour les Etats voisins.

Bon nombre de techniques ont été appliquées à l'agriculture pour l'aider à se passer des problèmes de la grandeur des surfaces cultivables comme les moissonneuses, semeuses, batteuses... Les drones ont été aussi impliqués.

Autrefois utilisés à des fins militaires et photographiques, ces engins trouvent une importante application dans le domaine agricole. Superviser, arroser des grandes surfaces cultivables, réaliser une inspection visuelle des terres afin de repérer les adventices ou les dégâts nuisibles mais aussi dotés de divers capteurs qui facilitent l'analyse de toute une panoplie de données : niveau d'azote, de chlorophylle, biomasse, taux d'humidité, stress hydrique, etc. De plus, le vol à basse altitude et la haute résolution permettent de récolter des vues aériennes plus pertinentes qu'avec une image satellite, aussi bien en viticulture qu'en arboriculture ou pour le maraîchage, l'horticulture et les grandes cultures de céréales.

Grâce à toutes ces informations et les préconisations sur les intrants nécessaires, les exploitants peuvent adapter précisément le niveau d'engrais ou d'autres pesticides à appliquer, et les zones concernées. Pour les agriculteurs qui utilisent régulièrement les drones, on remarque que la quantité de liquides épandus pour traiter les cultures a significativement diminué, facilitant ainsi l'agriculture durable et raisonnée. Les rendements sont également améliorés, ce qui permet un retour sur investissement relativement rapide, malgré un coût d'acquisition atteignant plusieurs milliers d'euros pour les drones les plus sophistiqués.

## **II. ETUDE DU MARCHE**

La principale activité de la population togolaise vit de l' agriculture ce qui explique un fort pourcentage de jeunes actifs dans ce domaine. Les drones agricoles constituent un besoin réel dans la réalisation d'une agriculture extensive. Superviser leurs cultures en temps réel, arroser et pulvériser aux insecticides, tels sont les problèmes majeurs auxquels ils sont parfois confrontés. Pour cela, la demande de ces drones est très forte, cette population désirant vivement accroître son rendement et dépasser vite le marché régional pour produire à l'échelle nationale et plus tard international.

### **1. Les clients potentiels :**

Les firmes agricoles et les entreprises exploitant les terres agricoles.

### **2. Les cibles visées :**

L' objectif de ce projet est d' aider les agriculteurs ou groupements d' agriculteurs à exploiter de très grandes surfaces

### **III. CAHIER DES CHARGES**

#### **Énoncé du besoin**

##### **Mise en situation**

Après le succès phénoménal de l' agriculture mécanisée, depuis quelques décennies grâce aux satellites, plusieurs exploitations agricoles dans les pays développés ont commencé par pratiquer la télédétection en se tournant ainsi vers une agriculture plus précise. Ce qui nous démontre que l' agriculture malgré sa forte mécanisation a besoin de plus en plus d' outils d' aide à la décision qui lui permettent de devenir une « agriculture de précision », soucieuse d' économie et de respect de l' environnement. Cependant, l' acquisition d' un satellite par un agriculteur moyen pour la mise en place d' une télédétection lui conduirait inévitablement à sa perte, qui plus est, un agriculteur d' un pays en voie de développement. Ce projet intervient donc pour apporter une solution simple adaptée pour tous, des grandes firmes agricoles aux petits paysans du tiers-monde.

##### **Problématique**

La charge consiste à concevoir un drone agricole capable d' assurer plusieurs fonctions simultanément en un temps record de façon plus économique dont le prix serait à la portée des agriculteurs afin d' affiner leur culture tout en garantissant

une rentabilité des investissements.

## Exigences de base du drone

- Mécanisme permettant au drone de survoler des hectares de domaine agricole assurant ainsi les fonctions d'épandage et de la télédétection
- Système d'alimentation autonome à base de l'énergie solaire assurant une autonomie parfaite du drone qui pourrait être utilisé dans des zones démunies d'énergie électrique
- Drone équipé de système de caméra de surveillance des cultures et massifs forestiers
- Possibilité d'adapter les buses de pulvérisation en fonction des produits à répandre sur les terres agricoles et être du type de cultures (maïs, blé, orge ou toute autre plante céréalière, légumes, etc.)

## Objectifs du projet

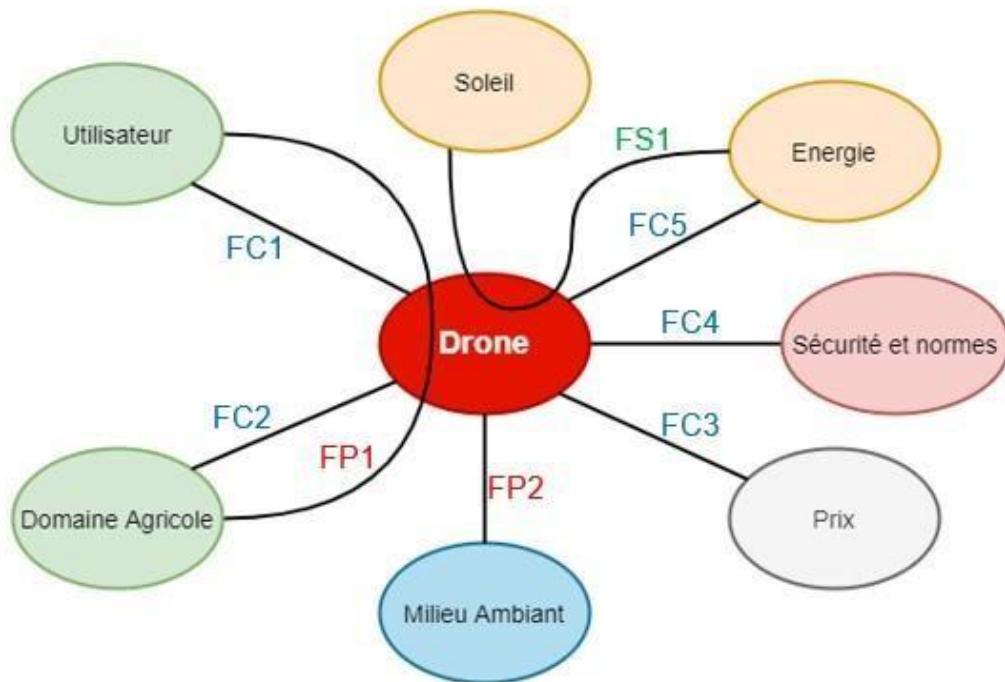
Les objectifs que nous espérons atteindre avec la réalisation de ce projet sont les suivants :

Ce projet a pour but de :

- Accroître le rendement des cultures pour générer d'énorme profit
- Réduire la pauvreté dans les pays du tiers monde en proposant sur le marché local et mondial des produits agricoles de qualités supérieurs
- Mettre en place la surveillance des cultures et massifs forestiers à moindre coup (croissance, attaques parasites, stress hydrique, adventices, matière organique, estimation des productions, dégâts aux cultures ...)

- Faciliter le traitement des zones précises et optimiser la quantité de produits phytosanitaires (produit de protection des plantes) utilisés en les répandant seulement où cela est nécessaire.
- Réduire le temps de travail en effectuant un survol et une dispersion automatique des fertilisants et des herbicides et d'épandre de façon régulière tout en gérant la distance de pulvérisation.
- Protéger les cultures en luttant contre les espèces animales envahissantes

## Fonction de service



- Fonctions Principales
  - FP1 : Surveiller le domaine agricole
  - FP2 : Voler dans la nature
- Fonctions Secondaires

- FS1 : Fournir de l' énergie

- Fonctions de Contraintes

- FC1 : Être facile d' utilisation
- FC2 : Avoir accès au domaine agricole
- FC3 : Avoir un prix d' achat accessible
- FC4 : Respecter les normes de sécurité ○
- FC5 : Être autonome

## Contraintes

### Contraintes en termes de délais

À partir de la livraison du cahier des charges, il faudra pour la réalisation du projet, un délai maximum d'un mois et demi.

### Contraintes de sécurité

La gestion de la sécurité est la principale contrainte du système. Il faudra implémenter un système robuste de log permettant de retracer tout l'itinéraire du drone à un instant t précis afin de pouvoir le retrouver en cas de perte.

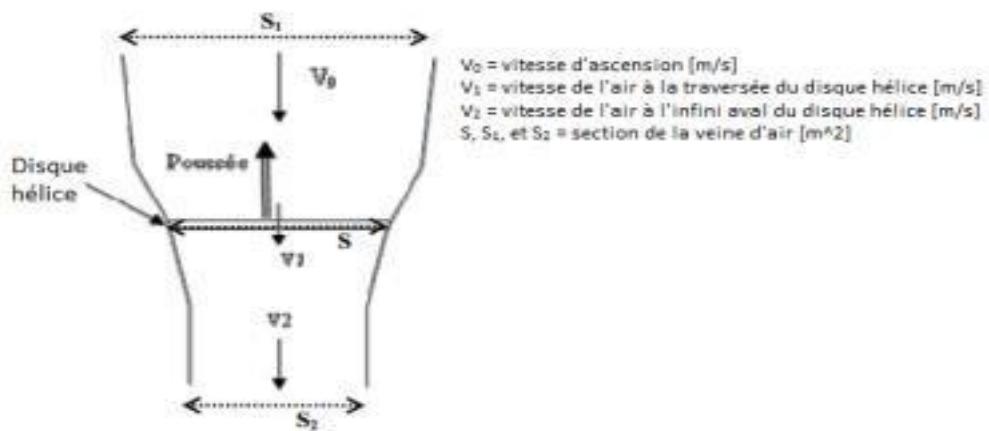
Perte qui peut être due à un dysfonctionnement ou au manque d'énergie.

### Contraintes économiques et environnementales

Dans l'optique de rendre le drone agricole accessible à tous, le coût de réalisation doit être abordable. La plupart des solutions technologiques seront choisies afin de pouvoir les réaliser au pays. D'autres pourront être commandées à l'extérieur. Les matériaux doivent être de bonne qualité et ne pas nuire à l'environnement. Le taux de pollution du drone doit être quasi inexistant à travers l'utilisation de l'énergie solaire comme source d'énergie qui constitue une énergie renouvelable. Un budget total de 5 millions de francs CFA devra être alloué à ce projet en vue de sa réalisation.

## IV. Dynamique du drone

### 1. Dimensionnement des hélices



Pour tout appareil volant, il faut un mécanisme de propulsion ou un générateur de force s'exerçant s'opposant à son poids pour ainsi le maintenir en altitude. Cette fonction est assurée par les hélices.

L' hélice est considérée comme un disque. Pour éviter les discontinuités de chaque côté du disque, on considère le disque comme infiniment fin. Deux forces s'exercent sur une hélice lorsqu'il est en rotation : La portance et la trainée.

La portance est l'opposé de son poids et la trainée sert à varier la force totale pour faire avancer le drone dans les directions voulues.

La Portance est donnée par :

$$P = Cz \times 1/2 \rho V^2 \times S$$

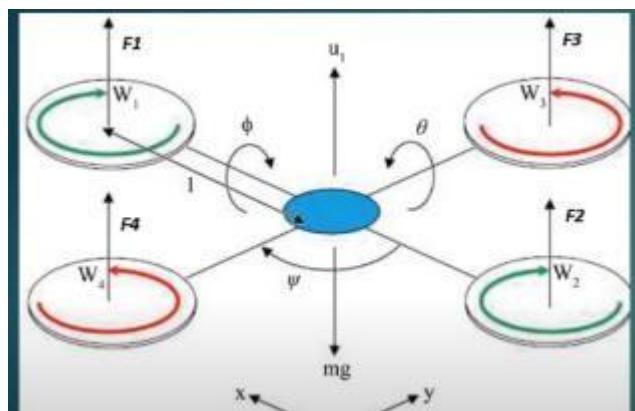
Avec : P : valeur de la portance en N (Newtons)

Cz : coefficient de portance -sans échelle-

$\rho$  : masse volumique du fluide en kg/m<sup>3</sup>

V : vitesse relative en m/s

S : surface projetée à l'horizontale en m<sup>2</sup>



Chaque hélice (i) génère une portance notée  $F_i$  sur la figure ; alors, on aura

$$F_i = P \text{ et la portance totale générée est } T = 4P = \sum_{i=1}^4 F_i$$

La projection de toutes les forces sur l' axe z donne

$$F_1 + F_2 + F_3 + F_4 = Mg$$

M la somme de la masse utile (masse de la charge) et de la masse du drone elle-même.

La condition nécessaire au vol est :

$$Mg \leq T$$

Pour le vol stationnaire, on a  $T=Mg$ ,

alors :

$$Mg = 4P \text{ donc } \frac{Mg}{4} = \frac{(RN)^2 \pi^3 \varphi C_z}{2 * 30^2} \text{ avec } V = 2\pi RN$$

Avec R le rayon de l' hélice (ce dernier en rotation étant considéré comme un disque tournant à une vitesse de rotation N) ;

$C_z$  le coefficient de portance.

En prenant  $C_z$  égale à 1 sur l' abaque (fonction de l' angle d' inclinaison de l' hélice) on a alors :

$$R = \left( \frac{2}{2 * N^2 \pi^3 C_z} \right)^{\frac{1}{4}} \quad \mathbf{30 * Mg}$$

- Le drone étant censé transporter 15 litres d' insecticides donc 15 Kg et la masse des accessoires estimé à 5 Kg, la masse totale sera de 20Kg
- La vitesse de rotation est prise à 1000 trs/min pour le vol stationnaire, le rayon maximal est alors :

$$R = \left( \frac{30^2 * 20 * 10}{2 * 1000^2 \pi^3 * 1} \right)^{\frac{1}{4}}$$

$$\underline{\underline{R=0,221 \text{ m}}}$$

## Calcul de la puissance mécanique à fournir pour pouvoir sustenter la masse du drone

$$\underline{\underline{P=T*V}}$$

$$P = \frac{MgN \pi R}{30}$$

$$\text{Donc } P = \frac{20 * 10 * 1000 * \pi * 0.221}{4 * 30}$$

$$\underline{\underline{P= 256,62 W}}$$

Donc chaque hélice est censée fournir une puissance de 256,62 W

Etant donné les pertes lors de la conversion de l'énergie électrique en énergie cinétique, on choisit les moteurs Brushless ayant un rendement égal à 0.9

Alors la puissance de chaque moteur sera :

$$Pel = \frac{P}{\eta}$$

$$\text{Alors } Pel = \frac{256.62}{0.9}$$

$$\underline{\underline{Pel= 285.13 W}}$$

Donc les moteurs doivent disposer au minimum d'une puissance de

$$Pel= 285.13 W$$

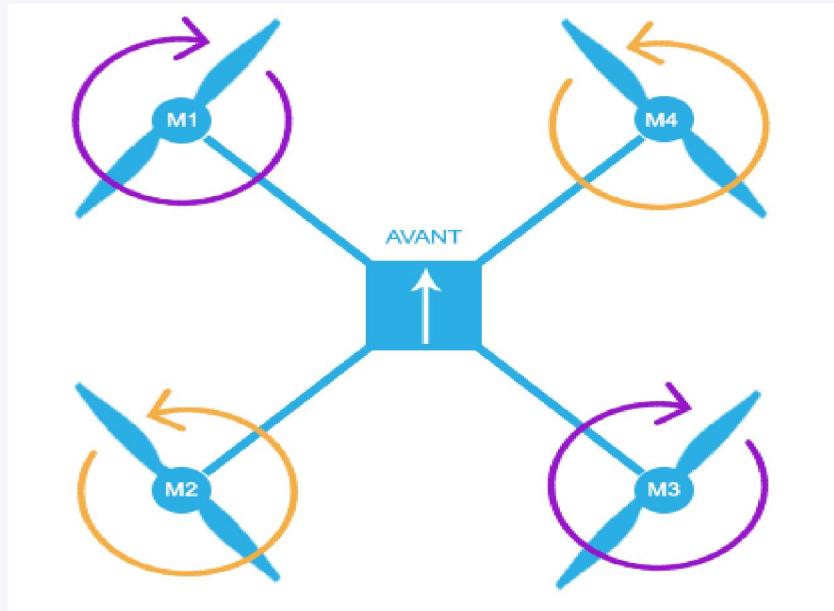
Il existe deux types de moteurs :

Les moteurs CW sur le même axe tournant dans le sens horaire et deux autres CCW tournant dans le sens anti-horaire.

## 2. Les différents mouvements :

### *Vol stationnaire :*

En tournant, les hélices vont créer une force de sustentation (effet d' une force qui maintient un corps à faible distance au-dessus d' une surface et sans contact avec elle) qui vont compenser le poids de l' engin. Lorsque cette force est supérieure au poids du multi copter, il s' élève dans les airs. Le sens de rotation des hélices est très important, sur le schéma ci-dessous, on constate que les hélices situées sur le même axe tournent dans le même sens. En d' autres termes, M1 et M3 tournent dans le sens horaire et M2 et M4 dans le sens antihoraire. Pourquoi ? Cela annule le couple induit par l' effort sur chaque moteur pour faire tourner les hélices.



### *Monter/descendre :*

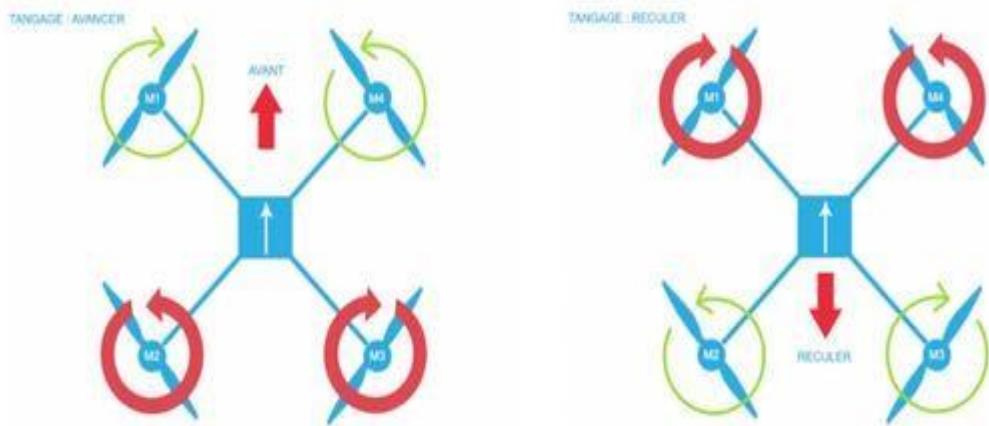
La portance étant proportionnelle au carré de la vitesse, plus elle augmente, plus la poussée totale que subit le drone augmente par rapport au poids et plus il monte ( $Mg \leq T$ ). D' où l' augmentation de la vitesse simultanée de tous les moteurs fait

monter le drone, de même que la diminution des vitesses la fait descendre, le poids de ce dernier devenant plus grande que la poussée totale ( $Mg \geq T$ )



Avancer/reculer :

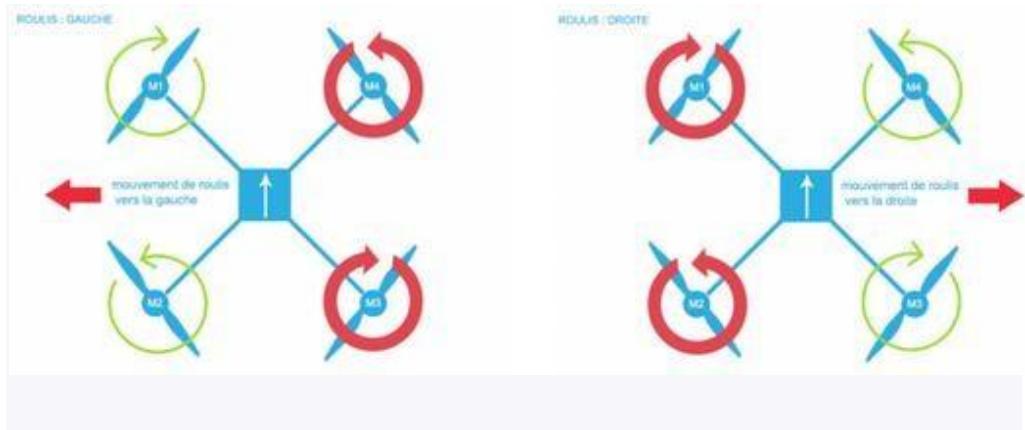
Pour avancer, on va diminuer la vitesse des moteurs avant et augmenter la vitesse des moteurs arrière et inversement pour reculer. On appelle cette action le tangage.



Gauche/Droite :

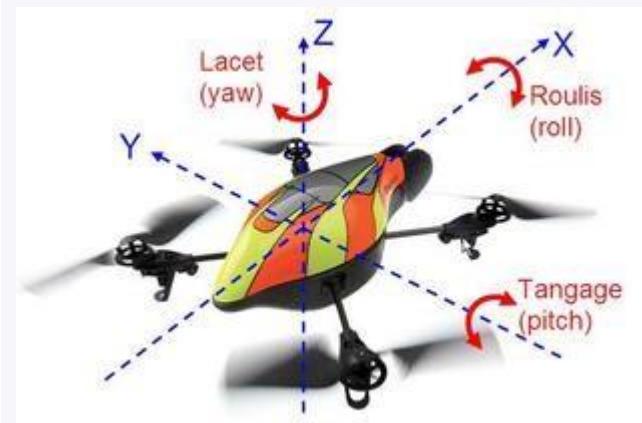
Pour incliner vers la gauche, on va diminuer les moteurs de gauche M1 et M2 et augmenter ceux de droite M3 et M4.

Inversement pour incliner vers la droite. Cette action s' appelle le Roulie.



### *Rotation :*

Pour un mouvement de rotation, on va augmenter la vitesse d'une paire de moteurs sur le même axe et inversement. Ceci est un mouvement de lacet.



3 gyroscopes sont indispensables pour le bon fonctionnement d'un drone :

- Lacet pour l'axe Z
- Roulis pour l'axe x
- Tangage pour l'axe y

On parle de "centrale inertielles" qui est l'association de l'ensemble des capteurs avec le logiciel embarqué et le hard pour le faire tourner. Il suffit de réfléchir 2 secondes pour s'apercevoir qu'il y a 6 mouvements à stabiliser et à contrôler :

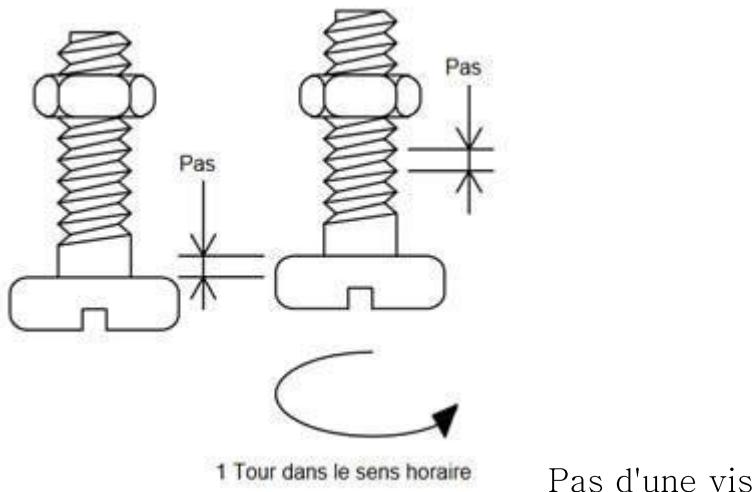
- A) les X, Y et Z linéaires
- B) les rotations autour de ces 3 axes.

### 3. Calcul de l'angle d'inclinaison théorique

#### Le pas

Le pas d'une hélice défini la distance qu'elle parcours lorsqu'elle fait un tour complet.

L'exemple le plus simple que tout le monde connaît est le boulon (vis + écrou). Lorsque la vis fait un tour complet dans l'écrou dans le sens horaire, elle avance d'une distance  $xx$  égale à son pas.



Pour une hélice c'est exactement la même chose sauf que là, l'écrou, c'est de l'air.

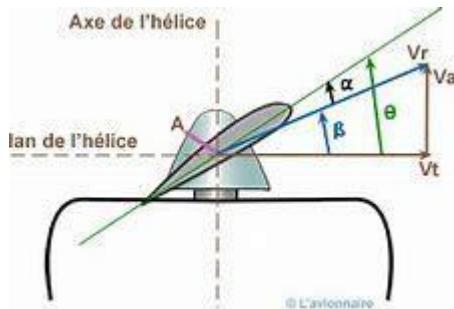
Si on sait maintenant de quoi on parle, reste à savoir ce que ça implique.

Un pas élevé implique une faible portance mais une vitesse de rotation maximale élevée. Un pas moindre implique une meilleure portance mais une vitesse de rotation max limitée (par compensation).

L' hélice a deux types de pas : Le pas effectif et le pas

géométrique.

Le pas géométrique est l' inclinaison de l' hélice et le pas effectif est le l' angle d' inclinaison de l' hélice à 0.7R



Pour le vol normal, le pas géométrique  $H=4.5$  en moyenne

$$\text{Le pas effectif est } h = \frac{H}{D} \text{ donc } h = \frac{4.5}{2.2}$$

$$h=2.045$$

Si  $\varphi$  est l' angle du pas géométrique et  $\beta$  celui du pas effectif.

La formule décrivant le pas effectif et géométrique est :

$$H=2\pi r \tan \varphi \text{ et } h=2\pi r \tan \beta$$

L' angle  $\alpha$  d' inclinaison de l' hélice est alors

$$\alpha = |\varphi - \beta|$$

Alors on a :

$$\alpha = \left| \tan^{-1}\left(\frac{H}{2\pi R}\right) - \tan^{-1}\left(\frac{h}{2\pi R}\right) \right|$$

Par calcul,

$$\underline{\alpha = |-9, 37|} \text{ donc } \alpha \text{ est proche de } 10^\circ$$

### Calcul du temps de vol.

La tension de la batterie

Nous utilisons les moteurs Brushless de 1000 kV (1kv)

=trs/min\*volt).

Le nombre de tours maximale par minutes est 10.000trs donc la tension  $U = \frac{N}{Kv}$

Le moteur doit avoir un Kv compris entre 900 et 1000kv

La valeur maximale de la tension sera obtenu pour le plus petit Kv possible donc pour 900kv

$$U = \frac{10000}{900}$$

$$U = 11.1V$$

#### 4. Consommation en puissance du microcontrôleur (Arduino)

I=500mA et U=7V donc la puissance consommée est  $P = U*I$

Donc  $P_{el} = 3.5W$

#### Calcul de la puissance des moteurs électriques

La puissance électrique nécessaire est  $P_{el} = \frac{P}{\eta}$

Les moteurs Brushless ont un rendement pouvant atteindre 0.9

$$\text{Alors } P_{el} = \frac{4*256.62}{0.9}$$

$$P_{el} = 1140.53W$$

#### Calcul de la puissance du moteur de la pompe

Les pompes insecticides ont en moyenne un débit

$$Q = 0.5GPM = 0.5 * 3.7855l/min$$

$$Q = 1.5l/min$$

Moteur de la pompe.  $P = F \cdot V = F \cdot \frac{Q}{s} = P \cdot Q$

Pour les pulvérisateurs la pression est comprise entre

$$P = 12 \cdot 10^5 \cdot 1.5 \cdot 10^{-3} / 60 \quad P = 30 \text{ W}$$

$$P_{el} = \frac{P}{\eta} \quad \eta = 0.9 \quad \text{donc } \underline{P_{el} = 33.33 \text{ W}}$$

$$P_{el \text{ Totale}} = 1140,53 + 33,33 + 3,5 \text{ W}$$

$$\underline{P_{el \text{ Totale}} = 1177,36 \text{ W}}$$

Indications de la Batterie sont : 16Ah 11,1 v

On a alors

$$\underline{E = P \cdot t}$$

$$t = T/P$$

$$\text{et } t = \frac{16 \cdot 3600 \cdot 11,1}{1177,36}$$

$$t = 543,04 \text{ S ou } \underline{t = 9 \text{ min } 05 \text{ s}}$$

Pour augmenter la durée du vol, on mettra en série 2 batteries dont l'une est de 16000mAh et l'autre 600mAh

Alors si on calcul à nouveau le temps de vol, on aura :

$$E = P \cdot t$$

$$t = T/P$$

$$\text{et } t = \frac{(16+6) \cdot 3600 \cdot 11,1}{1177,36}$$

$$\underline{t = 12 \text{ min } 44 \text{ s}}$$

## 5. Calcul de la surface pulvérisé

Pour une bonne pulvérisation, il faut une vitesse d' avance de  
 $v=12\text{m/s}$  (valeur pratique)

Alors :  $l=v*t$

$$=12,44*60*12\text{m/s}$$

$$l = 8,960 \text{ km}$$

La surface  $S=l*d$

Largeur de la surface que couvre le drone simultanément en vol :  $d=1\text{m}$

$$S=8960,25 \text{ m}^2 * 1\text{m}$$

$$\underline{S=8960,25\text{m}^2}$$

Les buses de pulvérisation

Buses à jet plat compact et injecter d' air IDK 90

Angle de pulvérisation de  $90^\circ$

Diamètre : 8mm

Taille des gouttelettes : Extrême grosses

## 6. Force et Moments

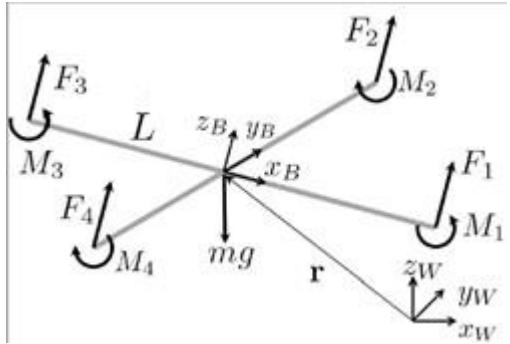


Figure III-A : Forces et moments

Chaque hélice en rotation produit deux types de forces. Quand un rotor tourne, cette hélice produit une poussée vers le haut étant donné par  $F = K_f * \omega^2$  (représenté par les forces F1, F2, F3 et F4 sur la Figure III-A) où  $\omega$  (oméga) est le taux de rotation du rotor mesurée en rad/s. Les valeurs de  $K_f$  et  $K_m$  (mentionnées ci-dessous) sont généralement trouvées empiriquement. Nous montons le moteur et l'hélice sur une cellule de charge et mesurons la force et le moment pour différentes vitesses de moteur.

La poussée totale vers le haut générée par les 4 hélices est donnée en additionnant toutes les poussées individuelles générées, pour  $i = 1$  à 4, elle est donnée par  $F_i = K_f * \omega^2$

Outre la force ascendante, une hélice en rotation génère également une rotation de rotation opposée appelée couple ou moment (représenté par les moments M1, M2, M3 et M4). Par exemple, un rotor tournant dans la direction CW produira un couple qui fera tourner le corps du drone dans la direction CCW.

Ce couple de rotation est donné par  $M = K_m * \omega^2$ . Le moment généré par un moteur est dans le sens opposé à sa rotation, donc les moteurs de rotation CW et CCW génèrent des moments opposés. Et c'est la raison pour laquelle nous avons des moteurs rotatifs CW et CCW de sorte qu'en état de vol stationnaire, les moments de 2 rotors rotatifs CW et 2 CCW s'annulent et le drone ne continue pas à tourner autour de son axe corporel (également appelé lacet). Les moments / couples M1, M2, M3 et M4 sont des moments générés par des moteurs individuels. Le moment global généré autour de l'axe z du drone ( $Z_b$  sur la Figure III-A) est donné par la sommation des 4 moments.

$Moment_z = M1 + M2 + M3 + M4$ , les moments CW et CCW auront des signes opposés et donc dans des conditions idéales

(ou chaque fois que nous ne voulons pas de mouvement de *lacet* (rotation autour de l'axe z)) , le *moment\_z* sera proche de 0.

Contrairement à *moment\_z*, le moment/couple global générée autour des calculs des axes x et y est peu différent. En regardant la Figure III–A, nous pouvons voir que les moteurs 1 et 3 se trouvent sur l'axe x du drone. Ils ne contribueront donc à aucun moment/couple autour de l'axe x. Cependant, nous pouvons voir que la différence de forces générées par les moteurs 2 et 4 fera basculer le corps du drone autour de son axe x et c'est ce qui constitue le moment / couple global autour de l'axe x, qui est donné par le moment

$\text{moment\_x} = (F2 - F4) * L$  , où L est la distance entre l'axe de rotation des rotors et le centre du quadra copter. Par la même logique,

$M_y = (F3 - F1) * L$ .  
En résumé, le moment autour des 3 axes peut être désigné par le vecteur

$$M = [M_x, M_y, M_z]^T, \quad (^T \text{ pour Transposé})$$

### Orientation et position



Figure III-B: Orientation et position

Un drone a des attributs de position et d'orientation, ce qui signifie qu'il peut être n'importe quelle position ( coordonnées x, y, z ) et peut faire certains angles (thêta ( $\theta$ ), phi ( $\varphi$ ) et psi ( $\psi$ )) par rapport à la terre / Cadre inertiel. La figure ci-dessus montre plus clairement thêta ( $\theta$ ), phi ( $\varphi$ ) et psi ( $\psi$ ).

### Se déplacer dans les directions z et x & y

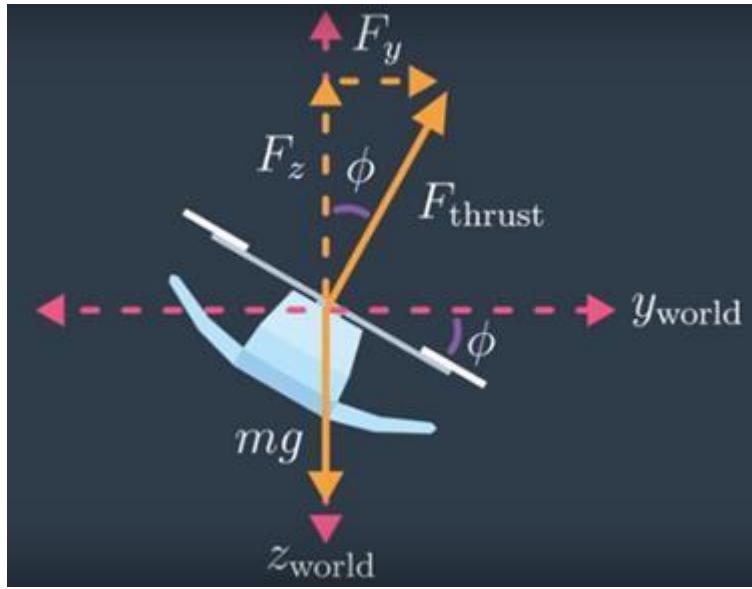


Figure 4 : Déplacement dans les directions z et x & y

Chaque fois qu'un drone est stationnaire, il est aligné avec le référentiel terrestre, ce qui signifie que son axe Z est dans la même direction que le champ gravitationnel de la terre. Dans un tel cas, si un drone veut se déplacer vers le haut, il lui suffit de régler la vitesse de rotation de l'hélice appropriée et il peut commencer à se déplacer dans la direction z en fonction de l'équation *force totale générée – la gravité*. Cependant, s'il veut se déplacer dans la direction x ou y, il doit d'abord s'orienter (en faisant l'angle theta ou phi requis). Lorsque cela se produit, la poussée totale générée par quatre hélices ( $F_{\text{thrust}}$ ) a une

composante dans la direction z et dans la direction x/y comme indiqué dans la figure 2D ci-dessus. Pour l'exemple ci-dessus,

en utilisant la trigonométrie de base, nous pouvons trouver la

force directionnelle z et y en suivant l'équation, où phi est

l'angle formé par l'axe z du corps du drone avec le cadre

mondial.

$$F_y = F_{\text{poussée}} * \sin \phi$$

$$F_z = F_{\text{poussée}} * \cos \phi$$

## Référentiel terrestre et Corps

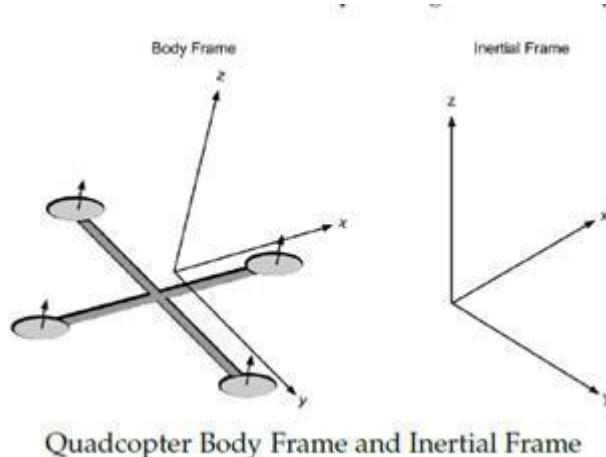


Figure 5 : Référentiel terrestre et celle du corps

Pour mesurer les angles theta, phi et psi indiqués ci-dessus, le

capteur IMU embarqué du drone est généralement utilisé. Ce capteur mesure la vitesse à laquelle le corps du drone tourne autour de son châssis et fournit cette vitesse angulaire en sortie. Lors du traitement de ces sorties IMU, nous devons être prudents et comprendre que les vitesses angulaires envoyées par celle-ci ne sont pas par rapport au Body frame, mais par rapport au référentiel du châssis. Le diagramme ci-dessus montre ces deux cadres à titre de référence.

### Matrice de rotation

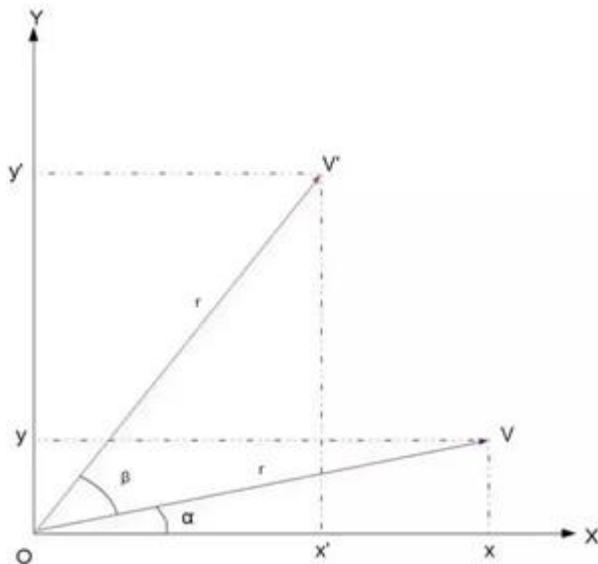


Figure 6 : Matrice de rotation

Pour convertir les coordonnées de Body Frame en World Frame et vice versa, nous utilisons une matrice  $3 \times 3$  appelée Rotation

Matrix. C'est-à-dire que si  $V$  est un vecteur dans les coordonnées terrestre et que  $V'$  est le même vecteur exprimé dans les coordonnées fixes du corps, alors les relations suivantes sont vérifiées :

$$V' = R * V \text{ et}$$

$$V = R^T * V' \text{ où } R \text{ est la matrice de rotation et } R^T \text{ est sa transposée.}$$

Pour bien comprendre cette relation, commençons par comprendre la rotation en 2D.  $V$  subit une rotation d'un angle  $\beta$  pour obtenir le nouveau vecteur  $V'$ . Soit  $r=|V|$ . Alors, on a les relations ci-dessous :

$$V_x = r \cos \alpha \text{ et } V_y = r \sin \alpha$$

$$V'_x = r \cos(\alpha + \beta) \text{ et } V'_y = r \sin(\alpha + \beta). \text{ En développant cela, nous obtenons}$$
$$V'_x = r * (\cos \alpha * \cos \beta - \sin \alpha * \sin \beta) \text{ et } V'_y = r * (\sin \alpha *$$

$$\cos \beta + \cos \alpha * \sin \beta)$$

$$V'x = vx * \cos \beta - vy * \sin \beta \text{ et } V'y = Vy * \cos \beta + Vx * \sin \beta$$

C'est exactement ce que nous voulons car le point souhaité  $V'$  est décrit en fonction du point d'origine  $V$  et de l'angle réel  $\beta$ .

Pour conclure, nous pouvons écrire ceci en notation matricielle sous la forme

$$\begin{bmatrix} v'_x & v'_y \end{bmatrix} = \begin{bmatrix} v_x & v_y \end{bmatrix} \begin{bmatrix} \cos \beta & \sin \beta \\ -\sin \beta & \cos \beta \end{bmatrix}$$

Passer de la 2D à la 3D est relativement simple dans le cas de Rotation Matrix. En fait, la matrice 2D que nous venons de dériver peut-être considérée comme une matrice de rotation 3D pour la rotation autour de l'axe z. Par conséquent, pour une rotation autour de l'axe z, la matrice de rotation serait

$$R_x(\beta) = \begin{bmatrix} v'_x & v'_y & v'_z \end{bmatrix} = \begin{bmatrix} v_x & v_y & v_z \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \beta & \sin \beta \\ 0 & -\sin \beta & \cos \beta \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} v'_x & v'_y & v'_z \end{bmatrix} = \begin{bmatrix} v_x & v_y & v_z \end{bmatrix} \begin{bmatrix} \cos \beta & 0 & -\sin \beta \\ 0 & 1 & 0 \\ \sin \beta & 0 & \cos \beta \end{bmatrix}$$

0, 0, 1 dans les valeurs de la dernière ligne et colonne indiquent que les coordonnées z du point pivoté ( $v'_z$ ) sont identiques à la coordonnée z du point d'origine ( $v_z$ ). Nous appellerons cette matrice de rotation de l'axe Z  $R_z(\beta)$ . En extrapolant la même logique aux rotations autour des axes x et y, nous pouvons obtenir des valeurs pour  $R_x(\beta)$  et  $R_y(\beta)$  comme

Et la valeur finale de la matrice de rotation de mouvement 3D sera simplement la multiplication croisée des trois matrices de rotation ci-dessus.

$R = R_z(\phi) \times R_y(\theta) \times R_x(\psi)$ , où psi ( $\phi$ ), phi ( $\psi$ ) et theta ( $\theta$ ) sont respectivement la rotation autour des axes z, y et x.

## 7. Vecteur d'état et ses dérivés

Comme notre drone a 6 degrés de liberté, nous le suivons généralement en surveillant ces six paramètres ainsi que leurs dérivés (comment ils changent avec le temps) pour obtenir une estimation précise de la position et de la vitesse de déplacement du drone. Nous le faisons en maintenant ce que l'on appelle souvent un vecteur d'état  $X = [x, y, z, \theta, \dot{x}, \dot{y}, \dot{z}, \dot{\theta}, \dot{\varphi}, \dot{\psi}, \ddot{x}, \ddot{y}, \ddot{z}, \ddot{\theta}, \ddot{\varphi}, \ddot{\psi}, p, q, r]$  et sa dérivée  $X_{dot} = [\dot{x}, \dot{y}, \dot{z}, \dot{\theta}, \dot{\varphi}, \dot{\psi}, \ddot{x}, \ddot{y}, \ddot{z}, \ddot{\theta}, \ddot{\varphi}, \ddot{\psi}, \ddot{p}, \ddot{q}, \ddot{r}]$  où  $x, y$  et  $z$  sont la position du drone dans le cadre mondial,  $\dot{x}, \dot{y}$  et  $\dot{z}$  sont les vitesses positionnelles/linéaires dans le cadre mondial.  $\varphi, \theta, \psi$  représentent l'attitude / l'orientation du drone dans le cadre mondial tandis que  $\dot{\varphi}, \dot{\theta}, \dot{\psi}$  représentent le taux de changement de cet angle (Euler).  $p, q, r$  sont des vitesses angulaires dans le cadre du corps tandis que  $\dot{p}, \dot{q}$  et  $\dot{r}$  sont son dérivé (dérivé = taux de changement) aka l'accélération angulaire dans le cadre du corps.  $\ddot{x}, \ddot{y}$  et  $\ddot{z}$  représentent les accélérations

linéaires dans World Frame.

## 8. Accélération linéaire

Comme indiqué précédemment, chaque fois que les hélices se déplacent, le drone commencera à se déplacer (accélérer) dans les directions x, y et z en fonction de la poussée totale générée par ses 4 hélices (représentées par  $F_{total}$  dans l'équation ci-dessous) et de l'orientation du drone (représentée par Matrice de rotation  $R$ ). Nous savons force = masse \* accélération. En ignorant la matrice de rotation  $R$ , si nous considérons uniquement l'accélération dans la direction z, elle serait donnée par

$$Z_{force} = (\text{masse} * \text{force gravitationnelle}) - (\text{masse} *$$

$$Z_{acceleration})$$

$$\text{masse} * Z_{acceleration} = \text{masse} * \text{force gravitationnelle} -$$

$$Z_{force}$$

$$\text{Et donc } Z_{acceleration} = \text{force gravitationnelle} - Z_{force} /$$

**masse**

En l'extrapolant aux directions x et y et en incluant la matrice

de rotation (pour les raisons décrites dans les sections 4 et 6),

l'équation décrivant l'accélération linéaire pour un drone est

donnée par l'équation ci-dessous, où  $m$  est la masse du drone et

$g$  est la force gravitationnelle. Le signe négatif dans  $F$  indique

que nous considérons la force gravitationnelle comme étant dans

la direction z positive.

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix} + \frac{1}{m} R \begin{pmatrix} 0 \\ 0 \\ -F_{total} \end{pmatrix}$$

Accélération linéaire

## 9. Accélération angulaire

En plus du mouvement linéaire, en raison des hélices rotatives

et de son orientation, le drone aura également un certain

mouvement de rotation. . Bien qu'il soit pratique d'avoir les

équations linéaires du mouvement dans le cadre inertiel / terre, les équations rotationnelles du mouvement nous sont utiles dans

le cadre du corps, afin que nous puissions exprimer des rotations autour du centre du quadricoptère au lieu d'environ

notre centre inertiel . Comme mentionné dans la section 4, nous utiliserons l'IMU du drone pour obtenir ses accélérations angulaires. Considérons que la sortie de l'IMU est  $p$ ,  $q$  et  $r$ , représentant les vitesses de rotation autour des axes  $x$ ,  $y$  et  $z$  du corps du drone.

Nous dérivons les équations de rotation du mouvement des équations d'Euler pour la dynamique des corps rigides. Exprimées sous forme vectorielle, les équations d'Euler s'écrivent sous la forme

$$I\dot{\omega} + \omega \times (I\omega) = \text{moment}$$

Équations d'Euler pour la dynamique des corps rigides où  $\omega = [p, q, r]^T$  est le vecteur vitesse angulaire,  $I$  est la matrice d'inertie et le *moment* est un vecteur de moment/couples externes développé dans la section 2. Nous nous en tiendrons à l'utilisation de  $\omega$  comme taux de rotation après cette section. Nous pouvons réécrire l'équation ci-dessus comme

$$\dot{\omega} = \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = (\text{moment} - \omega * (I * \omega)) / I$$

En remplaçant  $\omega$  par  $[p, q, r]^T$ , le vecteur de *moment en expansion et le remaniement au-dessus de l'équation, nous obtenons des accélérations angulaires dans le cadre du corps comme*

$$I \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} \text{moment\_x} \\ \text{moment\_y} \\ \text{moment\_z} \end{pmatrix} - \begin{pmatrix} p \\ q \\ r \end{pmatrix} \times I \begin{pmatrix} p \\ q \\ r \end{pmatrix}$$

Accélérations angulaires dans le châssis

## 10. Taux de changement des angles d'Euler

Bien que l'orientation du drone soit à l'origine observée dans Body frame, nous devons les convertir en World Frame. Encore une fois, nous utilisons la matrice de rotation selon la formule ci-dessous à cette fin. La dérivation de cette formule est peu allongée et est fournie dans la Référence [6]

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{pmatrix} \times \begin{pmatrix} p \\ q \\ r \end{pmatrix}$$

Taux de changement des angles d' Euler

Bien que l' orientation du drone soit observée à l' origine dans le cadre du corps, nous devons les convertir en cadre mondial. Encore une fois, nous utilisons la matrice de rotation selon la formule ci-dessous à cette fin.

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{pmatrix} \times \begin{pmatrix} p \\ q \\ r \end{pmatrix}$$

é

## V. Electronique

Description du hardware utilisé :

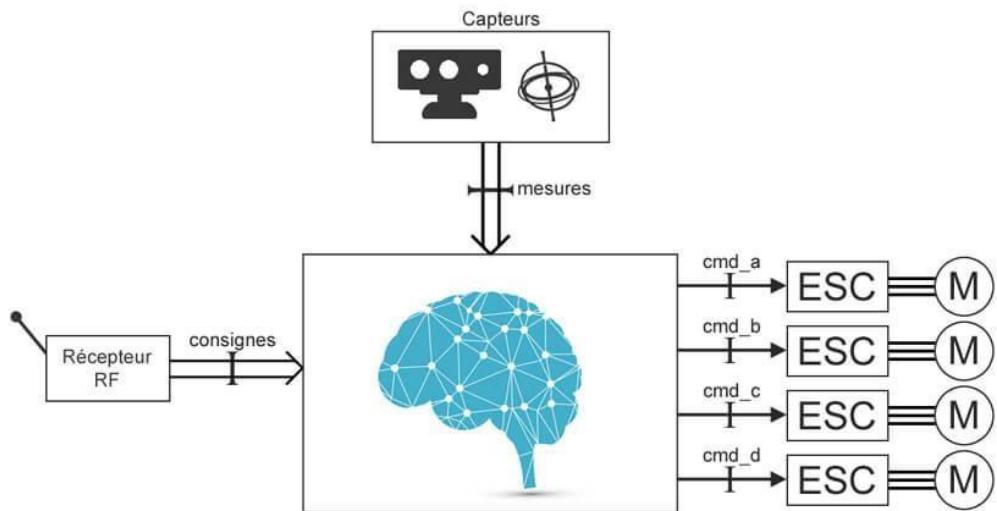
Les quadra copters sont des systèmes complexes, qui exigent des composants et une instrumentation variée qui doit être à la fois de haute performance et de poids allégés. Ici on va introduire les plus importants :

### 1. Arduino Uno :

Un Arduino Uno Figure 3.1, c'est le composant le plus simple pour commencer avec, il nous donne beaucoup de couvert pour

continuer à développer mon propre contrôleur de vol du quadra copter car il s'agit d'un PCB à double face professionnelle avec des faibles interférences et bruits, donc il convient bien pour le vol.

Il est le cerveau du drone.



Alors nous pourrons utiliser un Arduino pro mini pour cela, mais le problème c'est que l'alimentation sur un pro mini est basée d'un régulateur MIC5205, qui ne peut fournir qu'un 150mA avec un bon refroidissement.

Donc pour garder les choses sur le côté sûr, nous utilisons un Arduino Uno pour la réalisation du contrôleur de vol.



Figure 3.1 : la carte Arduino Uno.

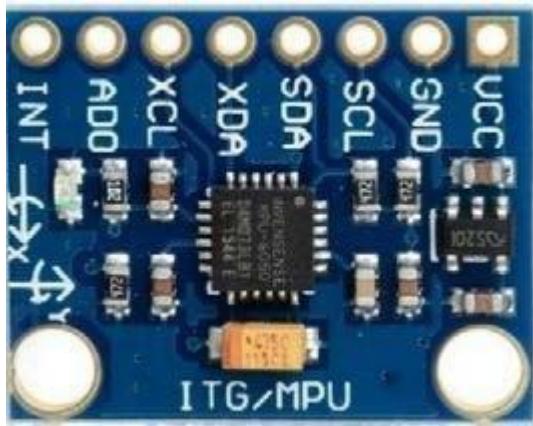
Il est basé sur un microcontrôleur ATmega en 8-bit et Il contient tout le nécessaire pour supporter le microcontrôleur ;

Il suffit de le connecter à un ordinateur avec un câble USB ou de l’ alimenter avec un adaptateur AC-DC ou une batterie pour le démarrer.

Dans le cadre de ce projet le traitement séquentiel des instructions de programme est important, on utilise souvent les interruptions des pates d’ entrées de l’ Arduino Uno pour lire les signaux de commande venant du récepteur radio aussi en utilise les sorties PWM de l’ Arduino Uno pour commander la rotation des moteurs Brushless via les ESC.

Finalement l’ utilisation du protocole de communication I2C permet de lire les données provenant de Gyro, de sorte que nous pouvons agir sur la stabilité du drone.

## 2. Gyroscope MPU-6050 :



Nous avons seulement besoin de corriger le mouvement indésirable du quadra copter, le Gyro c' est le capteur idéal de notre projet. Nous avons choisi le MPU-6050, car il est très facile à obtenir, il est également très bon marché et il détient dès un accéléromètre et un gyroscope la fois, ceci est très pratique à lors de la programmation d'une IMU.

La centrale inertielle ou l' IMU est un ensemble de capteurs qui peut être installé au bord de n'importe quel corps mobile pour lire les données des mouvements de ce corps.

Ce type de puce est en réalité ce qu'on appelle un micro **système électromécanique** ou un MEMS pour abrégé, ceci n'est rien d'autre que des capacités qui varient en fonction de l'accélération et de la vitesse angulaire.

Donc pour notre cas, nous pourrons déterminer et traiter les mouvements du quadra copter dans l'espace. L'interface du module MPU-6050 avec Arduino se fait aux broches (SCL et SDA) à l'intermédiaire d'une liaison I2C.

Le mouvement angulaire sur le quadra copter est partout le même. Donc, peu importe où se trouve le gyroscope. Mais c'

est toujours une bonne idée de monter le gyroscope au centre de gravité du quadra copter où les vibrations sont les plus faibles.

**Les moteurs sans balais « Brushless » :**

Dans le cadre de ce projet, nous avons besoin des moteurs de petite masse et à la fois de très haute vitesse avec un rapport couple-masse élevée. Pour cela nous sommes obligés de travailler avec les moteurs sans balais ou « Brushless BLDC motor ».

Les moteurs brushless sont en fait un type de moteur synchrone à aimants permanents tourne avec un courant alternatif triphasé AC.

Ils présentent de nombreux avantages par rapport aux moteurs DC classiques à savoir, leurs longues durées de fonctionnement et leurs insensibilités aux parasites en plus leurs réponses dynamiques élevées.

Ce type de moteur, c'est ce qui fait tourner et générer de la poussée pour permettre au drone de voler.

Dans le cas des multi rotors, les moteurs génèrent une poussée vers le haut qui maintient le drone comme un quadra copter volant. Autrement dit, un moteur sans balai contient un ensemble d'electroaimants (bobines) qui sont reliés ensemble dans des paires spécifiques.

Ces électroaimants sont connectés en trois sections principales, de sorte que le moteur brushless comporte trois fils sortants,

Figure 3.4.



Figure 3.3 :

Moteur Brushless EMAX XA2212 1400Kv.

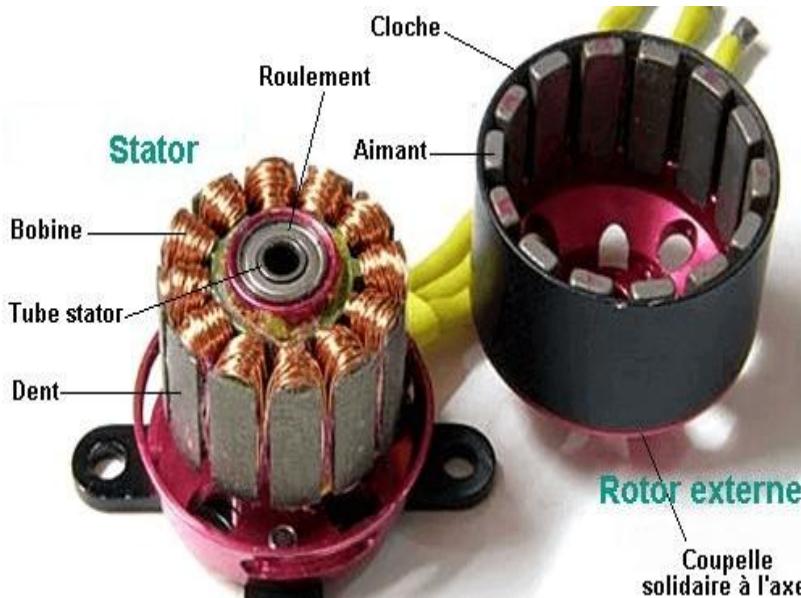


Figure 3.4 : Composantes interne du Moteur Brushless.

Le moteur sans balais se compose de deux sections principales :

Rotor : la partie qui tourne et a les aimants montés sur un motif radial.

Stator : la partie qui ne tourne pas, et dispose

d' électroaimants.

Nous utilisons pour ce projet quatre moteurs de la marque EMAX XA2212 1000Kv, ces moteurs ont 27.9 mm de diamètre et 43.16 mm de longueur, ils pèsent 100g et ils consomment une puissance de 300 Watts avec un courant max de 16.4A. Le KV veut dire la vitesse de rotation du moteur pour 1 volt. Il indique le nombre de tour/minute/volt du moteur à vide.

Dans notre cas, le circuit de puissance qui contient les moteurs fonctionne sur un voltage nominal de 11.1 volts, donc pour cela avec 1000 KV sous 11.1 volts chaque moteur peut effectuer jusqu'à 10000 tours/minute s'il est à vide c'est ce qui est appeler rotation par minute RPM.

### 3. Electronique Speed Controller « ESC » :

ESC signifie Electronique Speed Controller Figure 3.5, c'est un circuit électronique qui gère la vitesse des moteurs brushless. Sur un multi-rotor, chaque moteur obtient son propre ESC, dont chacun se connecte au contrôleur de vol.

Les ESC converties les séquences des signaux de commande PWM venant du contrôleur de vol ou du récepteur radio en un signal alternatif triphasé, et entraîne le moteur brushless en fournissant le niveau d'énergie électrique approprié.



Figure 3.5: variateur de vitesse « Hobbywing OPTO ESC 20A ».

Les ESC utilisés pour ce projet sont de la série hobby Kings 20A spécialement conçus pour les multi rotors, ils ont un large éventail de fonctions de programmation combinées avec une courbe d'accélération linéaire.

L'ESC doit supporter un ampérage au moins égal à l'ampérage max des moteurs. Généralement on prend un ESC d'ampérage un peu supérieur afin qu'il ne chauffe pas. Dans mon cas les moteurs brushless consomment 16.4A pour un max poussé « Thrust », nous choisissons un ESC de 20A afin de garder un peu de marge de sécurité.

#### 4. Batterie LIPO :

Notre *système* quadra copter besoin beaucoup d'énergie et a besoin d'une source d'alimentation puissante et en plus légère, pour cela nous sommes obligés de travailler avec la batterie du Lithium-Polymère Figure 3.6.

Les batteries Lithium-Polymère sont des batteries chimiques très puissantes et assez légère pour le bon vol du quadra copter, en plus elle inclut un circuit de protection (PCM) qui protège la

batterie contre les charges et décharges excessives et des court-circuits ou d'explosion [14].



Figure 3.6 : Batterie de Lithium-Polymère « LIPO » 11.1V 70C 2.2Ah.

Nous utilisons pour ce projet une batterie LIPO de 3 cellules et d'une capacité de 16000mAh à 11.1 volts avec un Taux de décharge « C Rating » de 70C et un poids de près de 1kg.

Le Taux de décharge est une façon informelle de décrire la quantité de courant que la batterie peut fournir en toute sécurité, Le C se réfère à la capacité de la batterie et à l'amplitude des heures. Dans notre cas cette batterie de 70C peut fournir jusqu'à  $70 \times 16\text{Ah} = 1120$  ampères dans une heure.

## 5. Les hélices :

Selon les calculs précédents, nous choisissons les DJI 8045 comme le montre la Figure 3.7. Ces hélices sont fabriquées avec la fibre de carbone qui est une matière résistante et légère, elles ont 8 pouces (21 cm) de rayon, un angle d' inclinaison de 10° , un pas de tangage de 4.5 pouces (un pas de tangage signifie combien de distance que le quadra copter va aller de l' avant pour une révolution à 360 degrés de l' hélice).



Figure 3.7 : Hélices DJI 8045.

Comme déjà précédemment, on prendra deux de sens horaire et deux de sens anti horaire en direction de la palme, ces hélices seront fixées aux moteurs et leur rôle sera de brassé l' aire pour générer la force de poussée. La taille de l' hélice varie avec la puissance du moteur, pour bien comprendre voici quelques règles à respecter pour le bon choix des différents accessoires :

Plus une hélice est petite, plus elle a besoin de tourner vite pour être efficace, plus elle est grande, moins elle a besoin de tourner vite.

Sur un gros modèle d' avion par exemple, une petite hélice même tournant très vite (beaucoup de KV), il ne suffira pas à le faire voler correctement (pas assez puissante). Donc il faudra l'

' équiper d' une grande hélice, qui aura besoin de tourner moins vite (moins de KV).

## 6. Arduino Nano :

L' Arduino Nano est une carte petite, complète et facile à utiliser basée sur l' ATmega328 Figure 3.8. Il a plus ou moins la même fonctionnalité de l' Arduino Uno, mais dans un package différent. Il ne dispose que d' une prise d' alimentation DC et fonctionne avec un câble USB Mini-B au lieu d' un câble standard pour se connecter au PC.

L' Arduino Nano peut être alimenté via une alimentation externe non régulée 6–20V (broche 30) ou Alimentation externe régulée 5V (broche 27).

La source d' alimentation est automatiquement sélectionnée pour la source de tension la plus élevée. Il est généralement destiné pour des réalisations qui nécessitent la miniaturisation et à l' installation semi-permanente dans des objets ou des expositions, on l' utilise souvent dans la réalisation du transmetteur RF.

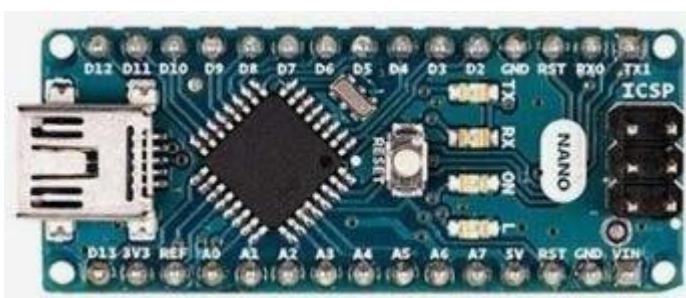


Figure 3.8 : Arduino Nano.

## 7. Module radio NRF24L01 2.4GHz :

Notre système quadra copter reçoit des données de contrôle en temps réel, il a donc besoin d' avoir un **système radio de bonne qualité**. Le transmetteur RF a besoin d' un module puissant pour envoyer le signal assez loin, mais le récepteur peut avoir un type normal d' antenne PCB. Le module radio que nous utilisons est celui avec l' antenne à puissance amplifiée,

Figure 3.9.



Figure 3.9 : NRF24L01 Radio module 2.4GHz.

L' NRF24L01 est un module émetteur-récepteur qui utilise la bande de 2,4 GHz et peut fonctionner avec un débit de 250 Kbps à 2 Mbps à l' air libre.

La consommation d' énergie de ce module est juste d' environ 12 mA au cours de la transmission et la tension de fonctionnement est de 1,9 Volts à 3,3 Volts. Le module NRF24L01 envoie et reçoit des données dans des «paquets» de plusieurs octets à la fois et la communication avec le microcontrôleur se fait via le protocole de communication SPI.

L' NRF24L01 peut utiliser jusqu' à 125 canaux différents, ce qui permet d' avoir un réseau de 125 modems indépendants en un seul module.

Ces unités ont beaucoup de complexité interne, mais certaines personnes talentueuses ont écrit des bibliothèques Arduino qui nous facilitent la tâche et nous donne beaucoup de liberté en programmation.

## 8. Les joysticks :

Les parties les plus importantes du contrôleur radio sont les joysticks, mieux dits, les potentiomètres.

Un joystick Figure 3.10, est un dispositif de contrôle du curseur utilisé dans les jeux informatiques et la technologie d' assistance basé sur des potentiomètres.

Le joystick qui a obtenu son nom du bâton de contrôle utilisé par un pilote pour contrôler les ailerons et les ascenseurs d' un avion, c' est un levier main qui pivote sur une extrémité et à transmet ses coordonnées par l' intermédiaire d' un microcontrôleur.

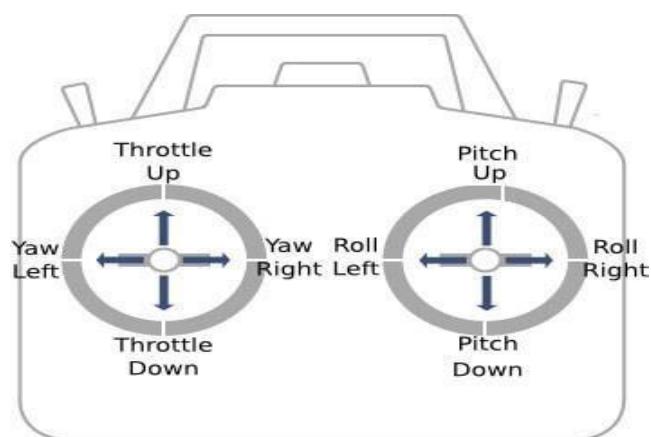


Figure 3 : Commandes de vol du drone.

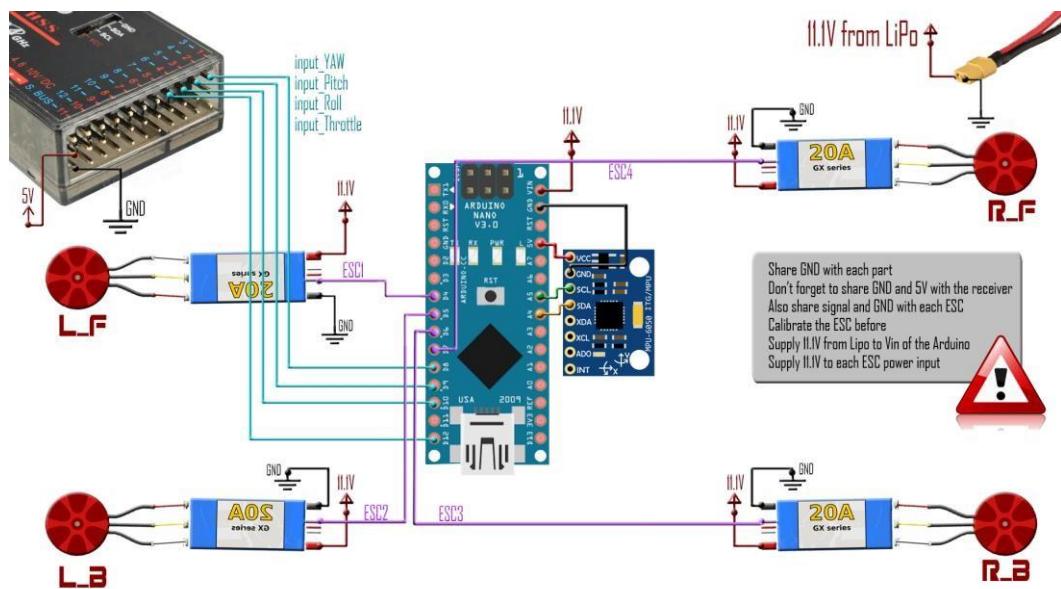
Les émetteurs radiocommande des quadras copter comportent deux potentiomètres de précision dans chaque joystick, dont ils varient la valeur de la résistance en fonction des accès de contrôle pour les mouvements Pitch, Roll, Yaw et Throttle,

Figure 3



Figure 3.10 : Joystick de haute qualité.

Dans ce contexte, on utilise l' Arduino Nano pour convertir le signal d' entrée de chaque potentiomètre en un signal numérique, puis le module radio NRF24101 permettra d' envoyer les variations de ces valeurs au récepteur du drone. En fait, ces **valeurs numériques** correspondent à la position du bâton de contrôle de chaque joystick.



## 9. Code:

```
#include <Servo.h>

bool firstRunSuccessful = false;

int ch1raw;
int ch1correct;
int ch2raw;
int ch2correct;
int ch3raw;
int ch3correct;
int ch4raw;
int ch4correct;
```

```
Servo esc9; // SILVER_B
```

```
Servo esc10; // BLACK_B  
Servo esc11; // BLACK_A  
Servo esc12; // SILVER_A  
  
int esc9Val;  
int esc10Val;  
int esc11Val;  
int esc12Val;  
  
void setup() {  
    // s' exécute une seule fois:  
  
    pinMode(4, INPUT); // Roll  
    pinMode(5, INPUT); // Throttle  
    pinMode(6, INPUT); // Pitch  
    pinMode(7, INPUT); // Yaw  
  
    esc9.attach(9);  
    esc10.attach(10);  
    esc11.attach(11);  
    esc12.attach(12);  
  
    esc9.writeMicroseconds(1000);  
    esc10.writeMicroseconds(1000);  
    esc11.writeMicroseconds(1000);
```

```
esc12.writeMicroseconds(1000);
delay(1000);

esc9.detach();
esc10.detach();

delay(1000);

esc9.attach(10);
esc10.attach(9);

esc9.writeMicroseconds(1000);
esc10.writeMicroseconds(1000);

Serial.begin(9600);
Serial.println("setup complete");

}

void loop() {
    // Boucle qui s' exécute indéfiniment

    delay(100);
    noInterrupts();
    ch1raw = pulseIn(4, HIGH); // Roll
    ch2raw = pulseIn(5, HIGH); // throttle
```

```
ch3raw = pulseIn(6, HIGH); // Pitch  
ch4raw = pulseIn(7, HIGH); // Yaw  
interrupts();  
  
//ch1correct = map(ch1raw, 1, 1, 1, 1);  
ch2correct = map(ch2raw, 1083, 1985, 2000, 1000);  
//ch3correct = map(ch3raw, 1, 1, 1, 1);  
//ch4correct = map(ch4raw, 1, 1, 1, 1);  
  
masterControl(ch1raw, ch2correct, ch3raw, ch4raw);  
}
```

```
void masterControl(int roll, int throttle, int pitch, int yaw)
{
    int minThrottle = 1180;
    if ((roll == 0) && (throttle == 3200) && (pitch == 0) &&
(yaw == 0)) {
        Serial.println("Turn on remote");
        return;
    }
    if ((roll < 1500) && (roll > 1496) && (throttle > 1538) &&
(throttle < 1544) && (pitch < 874)
&& (pitch > 870) && (yaw < 1500) && (yaw > 1496)) {

        if (firstRunSuccessful) {
            emergencyLanding();
        } else {
            Serial.println("Turn on remote");
        }
    }
}
```

```
    return;  
}  
  
if (throttle < minThrottle) {  
    esc9Val = 1000;  
    esc10Val = 1000;  
    esc11Val = 1000;  
    esc12Val = 1000;  
    esc9.writeMicroseconds(esc9Val);  
    esc10.writeMicroseconds(esc10Val);  
    esc11.writeMicroseconds(esc11Val);  
    esc12.writeMicroseconds(esc12Val);  
    return; //  
}  
  
setMotorSpeed(roll, throttle, pitch, yaw);
```

```
Serial.print(esc9Val); // Silver_B
Serial.print(", ");
Serial.print(esc10Val); // Black_B
Serial.print(", ");
Serial.print(esc11Val); // Black_A
Serial.print(", ");
Serial.println(esc12Val); // Silver_A
//*/
esc9.writeMicroseconds(esc9Val);
esc10.writeMicroseconds(esc10Val);
esc11.writeMicroseconds(esc11Val);
esc12.writeMicroseconds(esc12Val);

}
```

```

void setMotorSpeed(int roll, int throttle, int pitch, int yaw) {

    int maxMotorSpeed = 2000;

    // if everything is at rest, set values to the throttle (hover)
    if ((pitch > 1465) && (pitch < 1475) && (roll > 1505) &&
        (roll < 1515)) {

        esc9Val = throttle;
        esc10Val = throttle;
        esc11Val = throttle;
        esc12Val = throttle;
        return;
    }

    // Assign each motor value to its physical motor position. Done
    // for readability.

    int s_A = esc12Val;
    int b_A = esc11Val;
    int s_B = esc9Val;
    int b_B = esc10Val;

    // get 4 motor speeds based on throttle and pitch
    int pitchDifference;
    int frontMotorSpeed;
    int backMotorSpeed;
    if (pitch < 1465) {
        pitchDifference = map(pitch, 1000, 1469, 500, 0);

```

```
frontMotorSpeed = 2000;  
backMotorSpeed = backMotorSpeed - subtractionTotal;  
}  
  
}  
else if (pitch > 1475) {  
    pitchDifference = map(pitch, 1470, 2000, 0, 500);  
    // int forward = map(pitch, 1470, 2000, 0, 500);  
  
    backMotorSpeed = throttle + pitchDifference;  
    frontMotorSpeed = throttle;  
    if (backMotorSpeed > maxMotorSpeed) {  
        int subtractionTotal = backMotorSpeed - maxMotorSpeed;  
  
        backMotorSpeed = 2000;  
        frontMotorSpeed = frontMotorSpeed - subtractionTotal;  
    }  
}
```

```
s_A = frontMotorSpeed;  
b_A = frontMotorSpeed;  
s_B = backMotorSpeed;  
b_B = backMotorSpeed;  
  
// update 4 motor speeds to account for roll  
int rollDifference;  
int leftMotorSpeed;  
int rightMotorSpeed;  
if (roll < 1505) { // Left  
    rollDifference = map(roll, 1000, 1509, 500, 0);  
  
    //rightMotorSpeed is high  
    rightMotorSpeed = throttle + rollDifference;  
    leftMotorSpeed = throttle;  
    if (rightMotorSpeed > maxMotorSpeed) {  
        int subtractionTotal = rightMotorSpeed - maxMotorSpeed;
```

```
leftMotorSpeed = 2000;  
rightMotorSpeed = rightMotorSpeed - subtractionTotal;  
}  
  
}  
  
// XXX  
// Consider frontMotorSpeed, backMotorSpeed,  
leftMotorSpeed, rightMotorSpeed, to assign values.
```

```

//s_A | frontMotorSpeed, leftMotorSpeed
//b_A | frontMotorSpeed, rightMotorSpeed
//s_B | backMotorSpeed, rightMotorSpeed
//b_B | backMotorSPEED, leftMotorSpeed

s_A = (frontMotorSpeed + leftMotorSpeed) / 2;
b_A = (frontMotorSpeed + rightMotorSpeed) / 2;
s_B = (backMotorSpeed + rightMotorSpeed) / 2;
b_B = (backMotorSpeed + leftMotorSpeed) / 2;

// Assign each motor speed to its respective esc.
esc9Val = s_B;
esc10Val = b_B;
esc11Val = b_A;
esc12Val = s_A;

}

int roll(int rollInput) {
    return 0;
}

int throttle(int throttleInput) {
    return 0;
}

```



```
}
```

```
int yaw(int yawInput) {  
    return 0;  
}  
  
// XXX  
void emergencyLanding() {  
    Serial.println("Emergency Landing");  
  
    int landingSpeedList[] = {  
        2000, 1870, 1755, 1652, 1572, 1510, 1480, 1460, 1420,  
        1380, 1340, 1300, 1280, 1260, 1245, 1180,  
    };  
    int emergencyMotorSpeed = 2000;  
    esc9.writeMicroseconds(emergencyMotorSpeed);  
    esc10.writeMicroseconds(emergencyMotorSpeed);  
    esc11.writeMicroseconds(emergencyMotorSpeed);  
    esc12.writeMicroseconds(emergencyMotorSpeed);
```

```
int i = 0;  
while (i < 16) {  
    delay(1000);  
    Serial.println(landingSpeedList[i]);  
  
    esc9.writeMicroseconds(landingSpeedList[i]);  
    esc10.writeMicroseconds(landingSpeedList[i]);  
    esc12.writeMicroseconds(landingSpeedList[i]);  
  
    i++;  
}  
  
delay(5000);  
  
esc9.writeMicroseconds(1000);  
esc10.writeMicroseconds(1000);  
esc11.writeMicroseconds(1000);  
esc12.writeMicroseconds(1000);  
  
Serial.println("Emergency Landing Successful");  
}
```

comme de la récolte et du suivi des cultures (Monitoring and Yield Prediction), analyser et

## **CONCLUSION**

Aujourd’ hui, l’ agriculture se modernise pour répondre aux défis de son temps. Elle devient une agriculture de plus en plus traiter les cultures.

La technologie des drones pourrait aider les agriculteurs du monde entier à surveiller la croissance de leurs cultures, lutter contre les nuisibles, améliorer leur régime foncier, et bien plus encore, détecter les arbres infectés, traiter les images et extraction des informations pour prédire les récoltes, effectuer la préparation et visualisation des produits géospatiaux.

Les drones sont apparus comme une solution d’ aide à la décision, d’ analyse, de surveillance et de traitement des cultures.

En effet, les capteurs apportent des informations plus précises et donnent une panoplie de données : niveau d’ azote, de chlorophylle, biomasse, taux d’ humidité, stress hydrique….

Pour faire de cette agriculture une agriculture de précision, elle a besoin d’ utiliser ces technologies que sont les drones.