

RAPPORT DU PROJET DE FIN DE SEMESTRE (S5)

**Filière d'Ingénieur : Génie Électrique Option Systèmes
Embarqués et Informatique Industrielle**

Sous le thème :

Réalisation d'une carte de commande d'un drone

Réalisé par :

EL Amri Aymane

ELFATAOUI Abdelkodouss

Encadré par :

Pr. Abdessamad KLILOU

Résumé

Ce rapport se concentre sur la réalisation d'une carte de commande pour un drone, essentielle pour contrôler ses mouvements et assurer sa stabilité. Nous avons défini les objectifs, les spécifications techniques et les étapes clés du projet, incluant les exigences de performance et de sécurité ainsi que les outils et méthodes utilisés.

Le quadrirotor, choisi pour son élégance et sa dynamique, a été modélisé à l'aide du formalisme de Newton-Euler, avec un schéma de commande PID appliqué pour stabiliser le vol stationnaire. Les premiers essais ont démontré un potentiel prometteur, bien que des ajustements soient nécessaires. Une approche méthodique nous a permis d'identifier et d'optimiser les paramètres critiques.

En conclusion, le projet avance conformément aux prévisions, chaque étape renforçant notre compréhension et notre capacité à atteindre les objectifs fixés. Les bases solides posées permettront d'optimiser et de fiabiliser le drone, ouvrant la voie à des applications aériennes innovantes et efficaces.

Listes des abréviations

MLI	Modulation de Largeur d'Impulsion)
BLDC	Brushless Direct Current
PWM	Pulse Width Modulation
P	Proportional
PI	Proportional Integral
PID	Proportional Integral Derivative

Liste des figures

Figure 1.1 : les composants d'un drone	6
Figure 2.1 : Chaine de traction	11
Figure 2.2 : Montage de fonctionnement d'un brushless.....	12
Figure 2.3 : Schéma de câblage des sorties avec des ports AND	13
Figure 2.4 : Exemple de situation de commutation.....	14
Figure 2.5 : Modèle du moteur brushless	14
Figure 2.6: le modèle de BLDC pour la zone 1 ou Uab	14
Figure 2.7 : modèle de BLDC pour la 2ème zone.....	15
Figure 2.8: mouvements d'un quadrirotor	18
Figure 2.9 : Mouvement de roulis (ϕ)	18
Figure 2.10 : Mouvement de tangage (θ)	19
Figure 2.11: Mouvement de lacet (ψ).....	19
Figure 2.12 : Les forces qui agissent sur le drone	20
Figure 2.13 : signaux de commande de l'onduleur diphasé avec 120 °	22
Figure 2.14:schéma de bloc de moteur BLDC sur Matlab/simulink.....	22
Figure 2.15: Simulation de moteur brushless en boucle ouverte dans le sens inverse.....	23
Figure 2.16 : Simulation de moteur brushless en boucle ouvert	23
Figure 2.17 : schéma de bloc de BLDC moteur avec régulateur P	23
Figure 2.18 : simulation de BLDC moteur avec régulateur P	24
Figure 2.19: simulation de BLDC moteur avec régulateur P avec perturbation	24
Figure 2.20 : schéma de bloc de BLDC moteur avec régulateur P avec perturbation	24
Figure 2.21: schéma de bloc de BLDC moteur avec régulateur Pi	25
Figure 2.22: simulation de BLDC moteur avec régulateur Pi	25
Figure 2.23 : schéma de bloc de BLDC moteur avec régulateur Pi avec perturbation	25
Figure 2.24 : simulation de BLDC moteur avec régulateur Pi avec perturbation	26
Figure 2.25 : bloc de schéma de moteur bldc avec pid	26
Figure 2.26 : simulation de moteur bldc avec pid	26
Figure 2.27 : simulation de moteur bldc avec pid avec perturbation	27
Figure 2.28 : bloc de schéma de moteur bldc avec pid avec perturbation	27
Figure 2.29 : signaux de commande d'onduleur.....	28
Figure 2.30 : bldc avec onduleur(transistors MOSFET) sur proteus	28
Figure 2.31 : bldc avec onduleur (transistors BC647) sur proteus	28
Figure 2.32 : simulation de 4 moteurs bldc sur Proteus	29
Figure 2.33 : simulation de bldc avec onduleur (transistors BC647) sur proteus	29
Figure 2.34 : simulation de bldc avec onduleur (transistors MOSFET) sur proteus.....	29
Figure 2.35 : Schéma pour le calcul de la vitesse du moteur	30
Figure 2.36 : Résultat de simulation de Schéma pour le calcul de la vitesse du moteur.....	30
Figure 2.37 : Simulation de moteur Brushless avec Régulateur P	31
Figure 2.38 : Simulation de moteur Brushless avec Régulateur Pi	31
Figure 2.39 : imulation de moteur Brushless avec Régulateur PID	32
Figure 2.40 : l'interfaçage du module MPU6050 avec Arduino	32
Figure 2.41 : Schéma pour la vérification de batterie	33
Figure 3.1 : Moteur sans balais	37
Figure 3.2 : Contrôleur électronique de la vitesse.....	38
Figure 3.3 : Module d'alimentation gm v1.0	39
Figure 3.4 : Batterie Li-Po 11.1V 2200 mAh	39
Figure 3.5 : ArduPilot APM 2.8	40

Figure 3.6 : Transmetteur-récepteur Fly-sky	41
Figure 3.7 : les hélices des moteurs.....	42
Figure 3.8 : Arduino nano et MPU6050	43
Figure 3.9 : L'assemblage du châssis	44
Figure 3.10 : Soudage des ESC	45
Figure 3.11 : Schéma de câblage (Ardupilot).....	46
Figure 3.12 : Configuration d'Ardupilot	47
Figure 3.13 : Interface de connexion pour Ardupilot	48
Figure 3.14 : Control d'un moteur par le transmetteur	49
Figure 3.15 : Schéma de câblage (Arduino nano)	50
Figure 3.16 : la réalisation de la carte de commande	50
Figure 3.17 : Le drone après l'assemblage.....	51
Figure 3.18 : Essai du drone en conditions réelles.....	52

Liste des tableaux

Tableau 2.1 : les séquences de commutation pour le moteur BLDC	12
Tableau 3.1 : Caractéristiques	37
Tableau 3.2 : Caractéristiques du Dispositif RF	41

Table des matières

Résumé	i
Liste des abréviations	ii
Liste des tableaux	iii
Liste des figures	iv
Table des matières	v
Introduction générale.....	1
Chapitre 1 : Présentation de Projet et son cahier des charges	2
Introduction	4
1.1 Contexte de l'étude	4
1.2 Généralité sur les drones.....	4
1.2.1 Définition	4
1.2.2 Définition d'un quad-copter	5
1.2.3 Types des drones	5
1.2.4 Composants d'un drone.....	6
1.2.5 Fonctionnement.....	7
1.2.6 Stabilité d'un drone	7
1.3 Cahier des charges	8
Conclusion.....	8
Chapitre 2 : Modélisation, Études Théoriques et Simulations.....	9
Introduction	10
2.1 Chaine de traction	10
2.2 Moteurs Brushless	11
2.2.1 Schémas électriques	11
2.2.2 Modélisation du BLDC	14
2.3 Etudes théoriques.....	17
2.3.1 Principe d'un MLI.....	17
2.3.2 Fonctionnement d'un quadrirotor :	17
2.3.3 Mouvements d'un quadrirotor.....	18
2.3.3.1 Mouvement de roulis (ϕ)	18
2.3.3.2 Mouvement de tangage (θ)	18
2.3.3.3 Mouvement de lacet (ψ).....	19
2.3.3.4 Le vol stationnaire.....	19

2.3.3.5	Les translations verticales	20
2.3.3.6	Les translations horizontales	20
2.4	Effets physiques agissants sur le quadrirotor.....	20
2.4.1	Les forces	20
2.4.2	Le poids	21
2.4.3	La portance	21
2.4.4	La trainée.....	21
2.4.5	La traction ou la poussé.....	21
2.5	Partie puissance	22
2.5.1	Simulation de moteur brushless en boucle ouverte.....	22
2.5.2	Simulation de moteur brushless en boucle fermée.....	23
2.5.2.1	Régulateur proportionnel	23
2.5.2.2	Régulateur pi	25
2.5.2.3	Régulateur PID.....	26
2.6	Partie commande	27
2.6.1	Moteur brushless	27
2.6.1.1	Simulation de moteur Brushless en boucle ouverte	27
2.6.1.2	Simulation de moteur Brushless en boucle Fermée	30
2.6.1.3	Le Module MPU6050	32
2.6.1.4	Vérification de niveau de batterie	33
Conclusion.....		34
Chapitre 3 : Outils Matériel et Réalisation		35
Introduction		36
3.1	Outils matériels.....	36
3.1.1	Moteur sans balais	36
3.1.2	Contrôleur électronique de la vitesse	38
3.1.3	Module d'alimentation gm v1.0	38
3.1.4	Batterie Li-Po 11.1V 2200 mAh	39
3.1.5	ArduPilot APM 2.8.....	39
3.1.6	Transmetteur-recepteur Flay-sky.....	40
3.1.7	Les hélices 1045	41
3.1.8	Arduino nano et MPU6050	42
3.2	Assemblage du drone.....	43
3.3	L'assemblage du châssis	44
3.4	Soudage des composants	44

3.5	Schéma de câblage (en utilisant Ardupilot).....	45
3.6	Réalisation de la carte de commande	49
Conclusion.....		52
Conclusion générale.....		53
Références.....		
Annexe		

Introduction générale

Dans l'évolution constante de la technologie aéronautique, les drones, ou véhicules aériens sans pilote, jouent un rôle crucial dans divers domaines tels que la surveillance, la cartographie et la photographie aérienne. Au cœur de ces engins, la carte de commande est essentielle pour orchestrer les mouvements, assurer la stabilité et garantir la sécurité des opérations.

Ce rapport se concentre sur le processus de réalisation d'une carte de commande pour un drone, un élément clé influençant directement son comportement en vol. De la conception initiale à l'intégration des composants électroniques et à la programmation des algorithmes de contrôle, cette étude explore les aspects techniques et conceptuels nécessaires à sa mise en œuvre.

Nous aborderons les défis techniques, les choix de conception et les innovations nécessaires pour créer une carte de commande performante et fiable. Ce rapport fournit un aperçu complet du processus, mettant en lumière les étapes cruciales pour la réalisation réussie d'une carte de commande pour un drone, ouvrant ainsi la voie à des applications aériennes innovantes et efficaces.

Le premier chapitre, Présentation du projet et cahier des charges, établit le contexte de l'étude, présente la problématique et détaille les spécifications techniques et les contraintes du projet. Le deuxième chapitre, Outils de conception et matériel, présente les outils et composants nécessaires pour la conception et la réalisation de la carte de commande, incluant la description des chaînes de traction, des composantes matérielles et des outils logiciels utilisés.

Le troisième chapitre, Modélisation et études théoriques, couvre les étapes de modélisation et les études théoriques nécessaires à la conception de la carte de commande, incluant organigrammes, schémas électriques et analyse des effets physiques sur le drone quadrirotor. Le dernier chapitre, Réalisation, est consacré à la réalisation pratique de la carte de commande, décrivant les étapes de fabrication, les tests effectués et les ajustements finaux pour garantir le bon fonctionnement du système.

Ce projet explore les aspects techniques et conceptuels nécessaires à la mise en œuvre d'une carte de commande performante et fiable, ouvrant ainsi la voie à des applications aériennes innovantes et efficaces.

Chapitre 1 : Présentation de Projet et son cahier des charges

Introduction

Dans le cadre de notre dernière année à la FST de Béni Mellal, dans la filière d'Ingénieur en Génie Électrique, option Systèmes Embarqués et Informatique Industrielle, et dans le but d'exploiter nos apprentissages théoriques dans les différents aspects reliés à notre domaine d'étude, nous avons choisi de travailler sur le projet de la modélisation et la réalisation d'une carte de commande pour un drone. Ce projet inclut la conception du circuit embarqué du drone ainsi que la modélisation avec le logiciel de conception MATLAB et son extension SIMULINK.

Comme précisé dans notre introduction, nous travaillons sur la commande automatique et le contrôle d'un drone quadrirotor. Notre travail consiste à réaliser le circuit électronique embarqué ainsi que la modélisation.

1.1 Contexte de l'étude

La technologie et l'innovation ont conjointement permis l'apparition d'un nombre croissant de drones. Utilisés dans de nombreux domaines, les drones sont aujourd'hui principalement employés pour les prises de vue aériennes, bien que leurs fonctions soient très variées. Parmi celles-ci, on trouve notamment la régulation du drone.

Pour cette raison, le projet de modélisation et de conception d'un prototype miniature de régulation du drone intervient pour répondre à cette fonction en se basant sur la conception d'un circuit électronique embarqué et la modélisation de ce système. La régulation du drone a pour rôle principal de donner aux utilisateurs la possibilité de faire un vol autonome.

1.2 Généralité sur les drones

1.2.1 Définition

Un drone désigne un aéronef sans pilote à bord, généralement télécommandé depuis le sol. Il peut être programmé pour voler de façon autonome ou être contrôlé en Wi-Fi via un smartphone ou une tablette. Il existe deux types principaux de drones : les multirotors et les ailes volantes. Les multirotors, de formes, tailles et poids variés, sont propulsés par plusieurs moteurs, généralement électriques, mais parfois à essence, faisant tourner des hélices. Les ailes volantes, plus proches des avions, sont généralement propulsées par un moteur à hélices.

1.2.2 Définition d'un quad-copter

Un quad-copter est un aéronef à voilure tournante comportant quatre rotors pour sa sustentation. Les rotors sont généralement placés aux extrémités d'une croix. Afin d'éviter à l'appareil de tourner sur lui-même sur son axe de lacet, il est nécessaire que deux hélices tournent dans un sens et les deux autres dans l'autre sens. Pour pouvoir diriger l'appareil, il est nécessaire que chaque couple d'hélices tournant dans le même sens soit placé aux extrémités opposées d'une branche de la croix.

1.2.3 Types des drones

Il existe en effet plusieurs modèles de drones pour répondre à des besoins très divers :

- Le drone loisir ou mini drone : il prend de plus en plus de place dans le rayon jouet des enfants, mais correspond également aux adultes débutants souhaitant s'amuser. Certains modèles loisirs sont dotés d'une caméra pour la prise de vue aérienne. Ce type de drone a l'avantage d'être peu onéreux et se pilote facilement. On retrouve de plus en plus de mini drones pouvant être pilotés en intérieur et parfois même en extérieur.
- Le drone avec caméra : ce type de drone s'adresse aux personnes souhaitant réaliser de l'imagerie aérienne. Les caméras sont montées sur des axes pour une grande stabilité et selon les modèles, les caméras embarquées peuvent filmer en HD, full HD, 4 K voire davantage pour les drones destinés au cinéma !
- Le drone de course : le drone de course permet de pratiquer le FPV (First Person View) Racing, c'est-à-dire une course d'obstacles en totale immersion avec des figures acrobatiques. Les images capturées par la caméra du drone diffusent en effet en temps réel le parcours au pilote muni de lunettes FPV ou d'un masque FPV. Le vol en immersion demande de nombreuses heures d'entraînement et se pratique généralement sur des terrains privés.
- Le drone professionnel : depuis le 1er janvier 2018, tout drone de plus de 800 g est considéré comme un drone professionnel et doit être enregistré à la DGAC (Direction Générale de l'Aviation Civile). Les drones professionnels ont de multiples fonctions et sont plus complexes à piloter. Les pilotes professionnels peuvent survoler de nuit ou dans les zones urbaines s'ils en font la demande et obtiennent des autorisations. Les drones professionnels servent dans de nombreux domaines : thermographie, cartographie, surveillance, contrôle, recherche, etc.

➤ Le drone militaire : Le drone militaire est un drone entièrement différent et n'est utilisé que par l'armée, soit pour des missions de surveillance, de renseignements ou s'ils sont armés pour des missions d'attaques.

1.2.4 Composants d'un drone

La figure suivante représente les composants essentiels d'un quad-copter on détaillera par la suite les trois composantes principales.



Figure 1.1 : les composants d'un drone

- D'un châssis : c'est la base du drone. Selon les modèles, le châssis peut être différent et être composé de 3, 4, 6 ou 8 bras. Les bras d'un drone peuvent être simples ou doubles. Le châssis d'un drone doit être ergonomique afin de permettre à l'objet de s'envoler facilement. Les châssis peuvent être composés de bois, d'aluminium, de plastique ou encore de fibre de carbone.
- D'un système de propulsion : c'est ce qui va permettre au châssis de s'envoler. Pour cela, le système de propulsion comprend des moteurs, des hélices, des batteries et des contrôleurs de vitesse électroniques (ESC). Selon ces différents éléments, le temps de vol ainsi que la charge maximale supportée par le drone pourront différer.
- Un contrôleur de vol : grâce à un microprocesseur et des capteurs, il permet de faire le lien entre le pilote et le drone.

1.2.5 Fonctionnement

Le fonctionnement d'un quad-copter est assez particulier. On distingue quatre mouvements possibles : vertical, le lacet, le roulis et le tangage. Pour garder le contrôle du lacet, cela implique que deux hélices tournent dans le sens horaire (hélice à pas normal) et les deux autres dans le sens antihoraire (hélices à pas inversé).

- Le vertical

Le mouvement de vertical correspond tout simplement à la montée/descente du quadcopter. La montée est obtenue en augmentant la vitesse des quatre moteurs. La descente, qui elle est plus difficile à doser, s'obtient par la réduction de la vitesse des moteurs.

- Le lacet

Le mouvement de lacet sert à faire tourner le quad-copter sur lui-même. Il est obtenu en augmentant la vitesse des hélices à pas normal et en diminuant proportionnellement la vitesse des hélices à pas inversé.

- Le roulis et le tangage

Le roulis et le tangage sont des mouvements assez similaires visant à pencher le quad-copter sur un axe ou sur un autre. Ce mouvement est obtenu en augmentant la vitesse d'une hélice et en abaissant proportionnellement la vitesse de l'hélice opposée (hélice du même couple).

1.2.6 Stabilité d'un drone

Même quand il vente, les drones peuvent voler sur place et rester stable. Ils y parviennent parce qu'ils sont munis de gyroscopes. Ce dispositif est composé d'une roue qui tourne rapidement sur un axe. Si on tente de changer sa direction, la roue tendra à résister à ce changement de direction ; c'est l'effet gyroscopique, qui repose sur la conservation du moment angulaire. Il y a normalement trois gyroscopes dans un drone : Un sur l'axe des X empêche l'appareil de s'incliner sur l'avant ou l'arrière. Un sur l'axe des Y qui gère le tangage de gauche à droite puis un sur l'axe Z pour la rotation sur lui-même.

1.3 Cahier des charges

Le cahier des charges de notre projet de modélisation et de réalisation d'une carte de commande pour un drone quadrirotor inclut plusieurs aspects clés. Premièrement, il est essentiel de concevoir le circuit électronique embarqué du drone. Cette étape inclut la sélection des composants adéquats, la conception du schéma électronique, et la réalisation du circuit imprimé. Deuxièmement, notre projet nécessite la modélisation de ce circuit en utilisant le logiciel MATLAB et son extension SIMULINK. La modélisation doit permettre de simuler les différents aspects de la commande automatique et du contrôle du drone, en prenant en compte les dynamiques complexes d'un quadrirotor. Ce travail implique également la mise en place d'algorithmes de contrôle, tels que les régulateurs PID, pour stabiliser et diriger le drone. Le cahier des charges doit donc intégrer des spécifications techniques précises pour chaque composant, des critères de performance pour la modélisation, ainsi qu'un plan de test pour vérifier la fonctionnalité et la robustesse du système embarqué une fois réalisé. En résumé, notre projet couvre à la fois la partie matérielle, avec la conception du circuit embarqué, et la partie logicielle, avec la modélisation et la simulation des systèmes de commande.

Conclusion

En résumé, ce chapitre a présenté les objectifs, les spécifications techniques et les étapes clés de notre projet de modélisation et de réalisation d'une carte de commande pour un drone. Nous avons défini les exigences de performance et de sécurité, ainsi que les outils et méthodes que nous utiliserons. Ce cahier des charges nous fournira une feuille de route claire pour mener à bien ce projet, en assurant une approche structurée et méthodique pour atteindre nos objectifs.

**Chapitre 2 : Modélisation, Études Théoriques et
Simulations**

Introduction

Dans ce chapitre, nous présentons le schéma de traction ainsi que les outils de simulation adoptés tout au long de ce projet. La modélisation occupe une part importante pour l'analyse de tous systèmes, quels qu'ils soient, car il est indispensable de bien connaître le comportement du système physique et de son environnement. La modélisation des véhicules aériens à voilures tournantes est relativement complexe et requiert la connaissance de l'aérodynamique du véhicule étudié, de ses caractéristiques mécaniques ainsi que de la dynamique des actionneurs. Les difficultés sont amplifiées pour les systèmes de petite échelle en raison de la complexité des phénomènes aérodynamiques qui entrent en jeu, de leur plus forte sensibilité aux perturbations, et des limitations sur la charge utile qui génèrent des contraintes de dimensionnement et de poids pour les capteurs embarqués.

2.1 Chaine de traction

La chaîne de traction d'un drone est un ensemble de composants qui travaillent ensemble pour fournir la propulsion nécessaire au vol. Cette chaîne de traction, illustrée dans l'image fournie, comprend les éléments suivants :

- Hélices : Les hélices sont les dispositifs qui convertissent l'énergie mécanique des moteurs en poussée, permettant au drone de s'élever, de se stabiliser et de se déplacer dans différentes directions.
- Moteurs Brushless : Ces moteurs sans balais (brushless) sont utilisés pour entraîner les hélices. Ils sont choisis pour leur efficacité et leur durabilité. Chaque moteur est connecté à une hélice, fournissant la puissance nécessaire pour la propulsion.
- Contrôleurs de Vitesse Électroniques (ESC) : Les ESC (Electronic Speed Controllers) régulent la vitesse des moteurs brushless en ajustant la quantité de courant électrique fournie. Ils reçoivent des signaux de commande de la carte Arduino pour moduler la vitesse des moteurs en temps réel, ce qui permet de contrôler la stabilité et le mouvement du drone.
- Carte Arduino : L'Arduino agit comme le cerveau du système, recevant les commandes de l'application et les transmettant aux ESC. Il intègre les données des capteurs pour ajuster les commandes de vol et maintenir la stabilité du drone.
- Batterie : La batterie fournit l'énergie nécessaire au fonctionnement des moteurs et des autres composants électroniques.

- Accéléromètre et Gyroscope : Ces capteurs mesurent les mouvements et les orientations du drone, fournissant des données cruciales pour la stabilisation et la navigation.
- Application de Contrôle : Une application sur un smartphone ou une tablette envoie des commandes au drone via WiFi. Elle permet de piloter le drone et de surveiller son état en temps réel.
- Ordinateur Portable : Utilisé pour la programmation et le monitoring avancé. Il peut être utilisé pour ajuster les paramètres de vol et analyser les données recueillies par les capteurs.

Cette chaîne de traction intégrée permet au drone de voler de manière stable et contrôlée, en utilisant des moteurs puissants et des contrôleurs précis, soutenus par une multitude de capteurs pour assurer une navigation sécurisée et efficace.

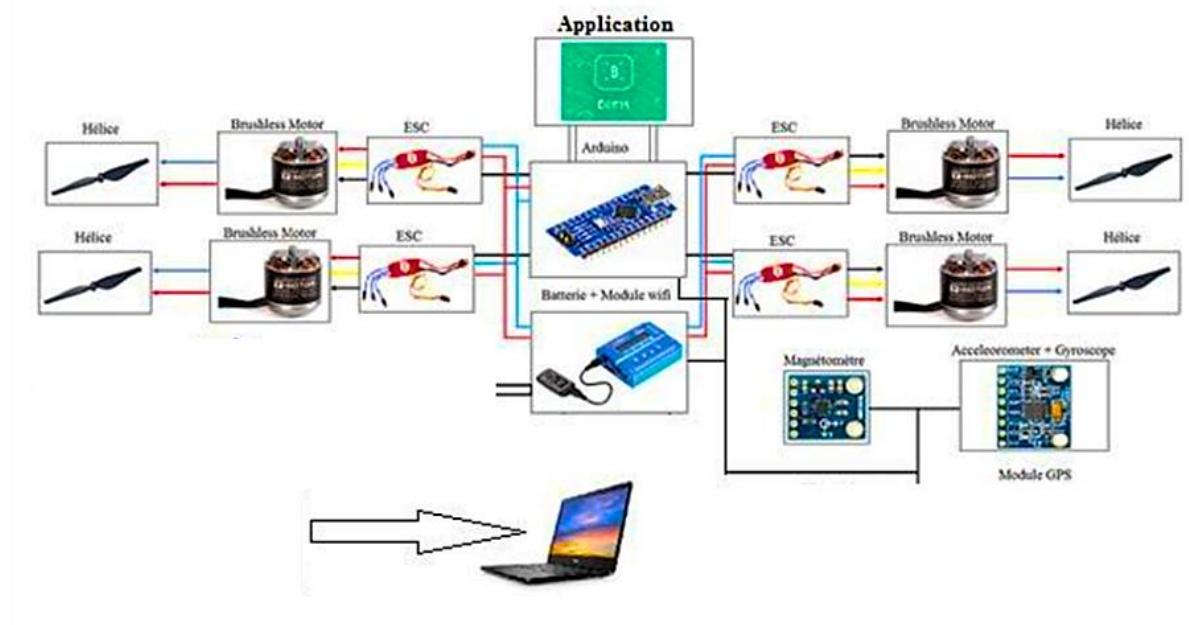


Figure 2.1 : Chaîne de traction

2.2 Moteurs Brushless

2.2.1 Schémas électriques

Les moteurs brushless sont constitués de 3 éléments principaux :

- Une partie fixe, le stator, muni de trois groupes de bobines, appelées les trois phases du moteur.
- Une partie tournante, le rotor, muni d'aimants permanents.

- Trois capteurs magnétiques à " effet Hall ".

Ce montage contient deux blocs :

1. Circuit de puissance :

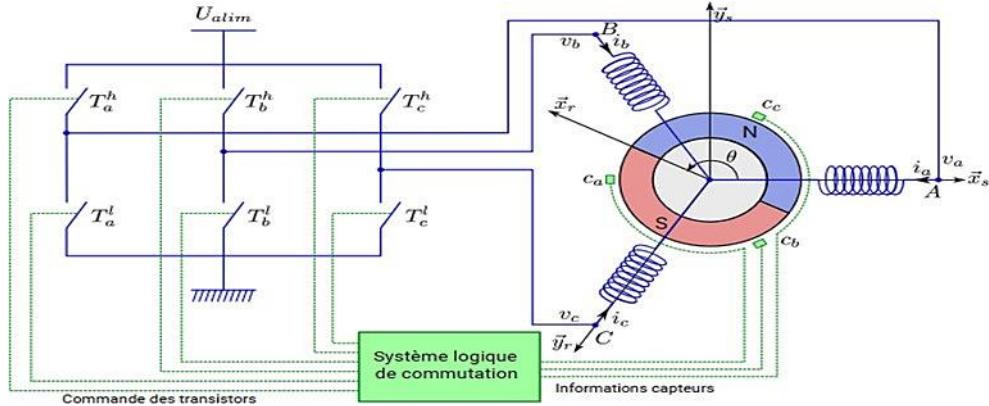


Figure 2.2 : Montage de fonctionnement d'un brushless

L'onduleur ou le contrôleur transforme le courant continu en courant triphasé à fréquence variable et va alimenter successivement les bobines de stator ; Ces bobines fonctionnent comme des électroaimants et permettent de générer un champ magnétique tournant qui va créer des pôles fictifs qui vont attirer les pôles réels de rotor.

De ce fait, l'aimant permanent de rotor tend à s'aligner sur le champs magnétique tournant.

2. La logique de commande ou bien circuit de commande :

La logique de commande détermine la position du rotor à partir des capteurs à effet Hall. Elle en déduit l'orientation à donner au champ magnétique du stator. Au cours de la rotation, elle commande les trois bobines pour ajuster régulièrement l'orientation du champ à la position du rotor. La plupart des moteurs BLDC ont trois capteurs de Hall intégrés dans le stator sur le côté opposé à l'entraînement du moteur. Chaque fois que les pôles magnétiques du rotor passent près des capteurs à effet Hall, ils donnent un signal haut ou bas, indiquant le pôle N ou le pôle S passant à proximité des capteurs. Sur la base de la combinaison de ces trois signaux de capteurs de hall, la séquence exacte de commutation peut être déterminée.

La logique de commande permet de contrôler les trois phases du moteur BLDC à travers six transistors en exploitant les sorties des capteurs à effet de Ha, Hb et Hc.

Le tableau suivant montre les séquences de commutation pour le moteur BLDC :

Tableau 2.1 : les séquences de commutation pour le moteur BLDC

Zone	Alim	H1	H2	H3	Sens trigonométrique					Sens inverse						
					S1	S2	S3	S1'	S2'	S3'	S1	S2	S3	S1'	S2'	S3'
1	a+b-	1	0	1	1	0	0	0	1	0	1	0	0	0	1	0
2	a+c-	1	0	0	1	0	0	0	0	1	0	0	1	0	1	0
3	b+c-	0	1	0	0	1	0	0	0	1	0	0	1	1	0	0
4	b+a-	0	1	0	0	1	0	1	0	0	0	1	0	1	0	0
5	c+a-	0	1	1	0	0	1	1	0	0	0	1	0	0	0	1
6	c+b-	0	0	1	0	0	1	0	1	0	1	0	0	0	0	1

On trouve les signaux de commande suivants :

$S1=Ha \cdot Hb'$	$S2=Hb \cdot Hc'$	$S3=Ha' \cdot Hc$
$S1'=Ha' \cdot Hb$	$S2'=Hb' \cdot Hc$	$S3'=Ha \cdot Hc'$

En câblant ces sorties par des portes AND on trouve le schéma suivant :

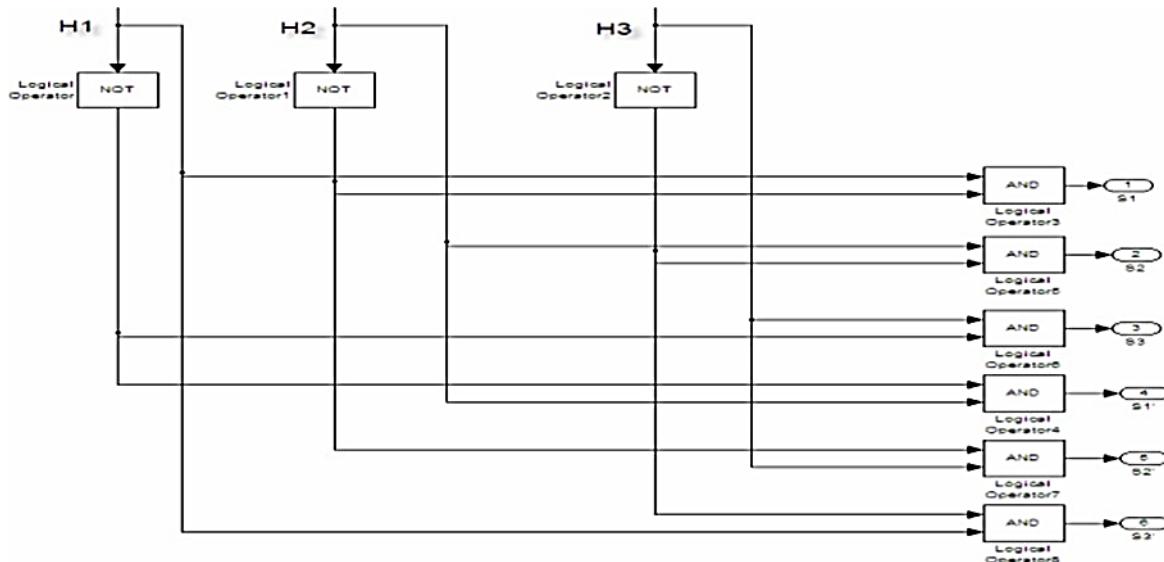


Figure 2.3 : Schéma de câblage des sorties avec des ports AND

Dans le cas simple du moteur BLDC, à chaque commutation, deux phases sont reliées respectivement à la tension d'alimentation et à la masse, et une phase n'est pas connectée. Prenons l'exemple suivant :

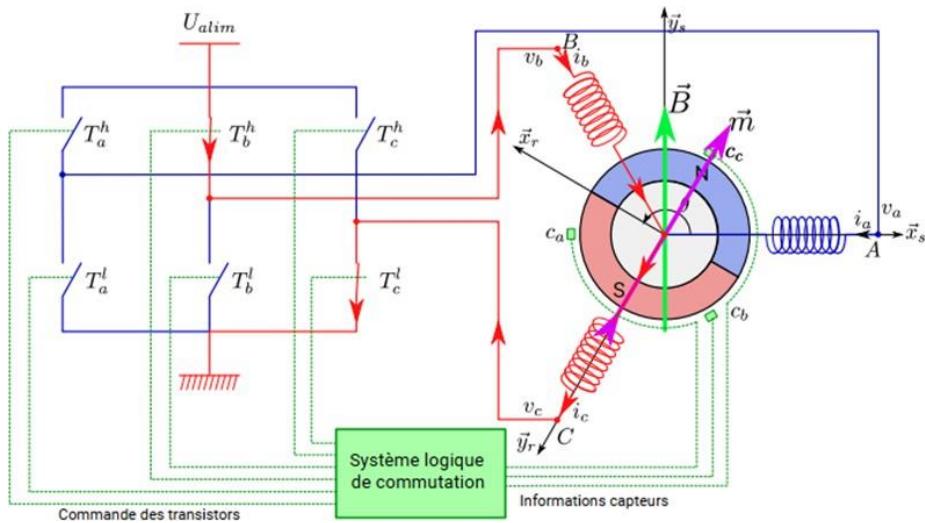


Figure 2.4 : Exemple de situation de commutation

2.2.2 Modélisation du BLDC

Le moteur brushless peut être modélisé ainsi :

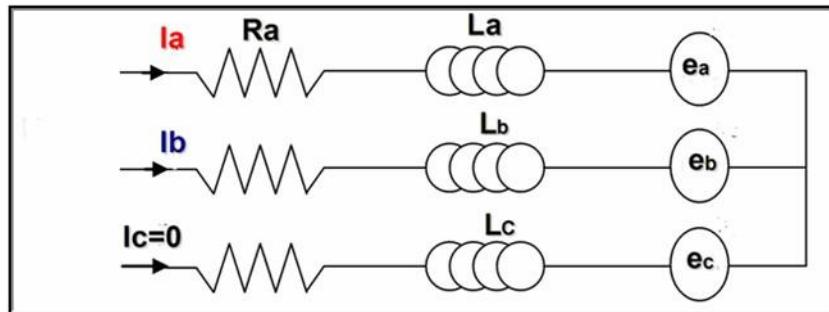
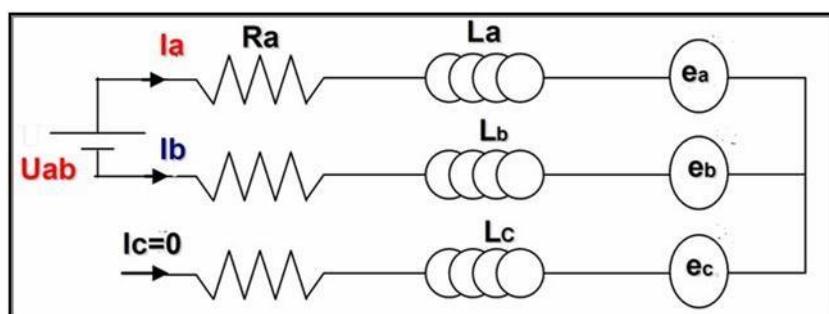


Figure 2.5 : Modèle du moteur brushless

a) Étude zone 1 :

Cette zone est caractérisée par : $I_a = I$, $I_b = -I$ et $I_c = 0$ avec la tension $U = U_{ab} > 0$

Figure 2.6: le modèle de BLDC pour la zone 1 ou U_{ab}

On a : $V_a = R_a I_a + \frac{d\Phi_a}{dt}$ Φ_a est le flux traversant la bobine a.

$$\Phi_a = \Phi_{aa} + \Phi_{ba} + \Phi_{ca} + \Phi_r$$

$$\Phi_a = L_a I_a + M I_b + M I_c + \Phi_r$$

Ainsi l'expression V_a devient :

$$V_a = R_a I_a + L_a \frac{dI_a}{dt} + M \frac{dI_b}{dt} + M \frac{dI_c}{dt} + E_a$$

Compte tenu de : $R_a = R_b$ et $L_a = L_b$ l'expression V_a devient :

$$V_a = R_a I_a + L_a \frac{dI_a}{dt} - M \frac{dI_a}{dt} + E_a$$

Donc :

$$V_a = R_a I_a + (L_a - M) \frac{dI_a}{dt} + E_a$$

De la même façon on démontre que :

$$V_b = -R_a I_a - (L_a - M) \frac{dI_a}{dt} + E_b$$

On a : $E_a = -E_b = k\theta\Omega = -E_{max}$ donc l'expression U_{ab} est :

$$U_{ab} = V_a - V_b = 2R_a I_a + 2(L_a - M) \frac{dI_a}{dt} + 2E_{max}$$

En posant :

$$R = 2R_a, L = 2(L_a - M) \text{ et } E = 2E_{max}$$

On obtient une expression qui est bien l'équation électrique de la MCC :

$$U_{ab} = RI_a + L \frac{dI_a}{dt} + E$$

a) Etude zone 2 :

b) Etude zone 2 :

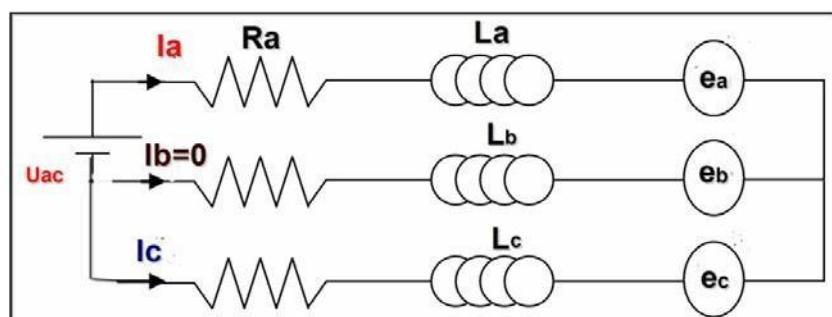


Figure 2.7 : modèle de BLDC pour la 2ème zone

Cette zone est caractérisée par : $I_a = I$, $I_c = -I$ et $I_b = 0$ avec la tension $U = U_{ac} > 0$

$$\text{On a : } V_a = R_a I_a + \frac{d\Phi_a}{dt}$$

$$\Phi_a = L_a I_a + M I_b + M I_c + \Phi_r$$

$$\text{Et : } V_c = R_c I_c + \frac{d\Phi_c}{dt}$$

$$\Phi_c = L_c I_c + M I_b + M I_a + \Phi_r$$

En tenant Compte de : $R_a = R_c$ et $L_a = L_c$ l'expression V_a devient :

$$V_a = R_a I + (L_a - M) \frac{dI}{dt} + E_a$$

De la même façon on trouve que :

$$V_c = -R_a I - (L_a - M) \frac{dI}{dt} + E_c$$

On a : $E_a = -E_c = k\Omega = -E_{max}$ donc l'expression U_{ac} est :

$$U_{ac} = V_a - V_c = 2R_a I + 2(L_a - M) \frac{dI}{dt} + 2E_{max}$$

En posant :

$$R = 2R_a, L = 2(L_a - M) \text{ et } E = 2E_{max}$$

On obtient une expression qui est bien l'équation électrique de la MCC :

$$U_{ac} = RI + L \frac{dI}{dt} + E$$

D'après l'étude des deux zones 1 et 2, on peut déduire que la tension aux bornes du moteur brushless garde la même forme sur toutes les zones de fonctionnement.

Le couple de moteur :

$$T = \frac{I_a E_a + I_b E_b}{\Omega}$$

Dans la zone 1, on a :

$$\text{Or dans cette zone on a : } I_a = -I_b = I \text{ et } E_a = -E_b = E_{max}$$

$$\text{Donc : } T = 2I \left(\frac{E_{max}}{\Omega} \right) = KI \quad \text{avec : } K = 2 \left(\frac{E_{max}}{\Omega} \right)$$

$$\text{L'équation mécanique est : } j \frac{d\Omega_{mec}}{dt} = T_e - T_i - f\Omega_{mec}$$

2.3 Etudes théoriques

2.3.1 Principe d'un MLI

Pour commander le moteur de façon plus progressive, par exemple pour réaliser un asservissement, il faut pouvoir régler le niveau de tension. Cette opération est classiquement réalisée par un onduleur commandé par une modulation à largeur d'impulsion (MLI ou PWM). La commutation utilisant déjà trois demi-points, Chaque interrupteur électronique est fermé pendant une demi-période, les commandes des demi-points étant décalées de 120. Il est assez naturel d'associer une MLI à la commutation, la MLI consiste à comparer la tension de référence V_{ref} avec un signal triangulaire V_{tri} d'une haute fréquence dont les extrémités sont définies entre $-V_{dc}$ et $+V_{dc}$ tel que :

- Si $V_{ref} > V_{tri}$: le signal de commande $S=1$, l'interrupteur est à l'état fermé. La tension aux bornes de la charge est égale à $+V_{dc}$.
- Si $V_{ref} < V_{tri}$: le signal de commande $S=0$, l'interrupteur est à l'état ouvert. La tension aux bornes de la charge est égale à $-V_{dc}$.

2.3.2 Fonctionnement d'un quadrirotor :

Un quadrirotor est un robot mobile aérien à quatre rotors définit dans l'espace par 6 DDL. Ces 4 rotors sont généralement placés aux extrémités d'une croix, et l'électronique de contrôle est habituellement placée au centre de la croix. Afin d'éviter à l'appareil de tourner sur lui-même sur son axe de lacet, il est nécessaire que deux hélices tournent dans un sens, et les deux autres dans l'autre sens. Pour pouvoir diriger l'appareil, il est nécessaire que chaque couple d'hélice tournant dans le même sens soit placé aux extrémités opposées d'une branche de la croix.

Le fonctionnement d'un quadrirotor est assez particulier. En faisant varier astucieusement la puissance des moteurs, il est possible de le faire monter/descendre, de l'incliner à gauche/droite (roulis) ou en avant/arrière (tangage) ou encore de le faire pivoter sur lui-même (lacet), le quadrirotor a six degrés de libertés, trois mouvements de rotation et trois mouvements de translation.

2.3.3 Mouvements d'un quadrirotor

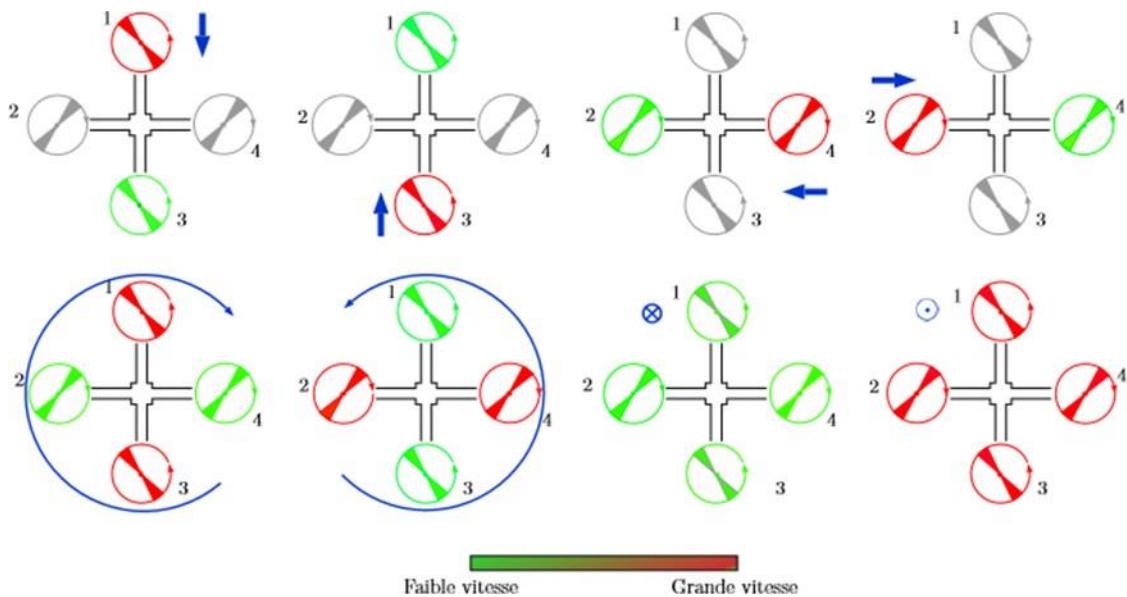


Figure 2.8: mouvements d'un quadrirotor

2.3.3.1 Mouvement de roulis (ϕ)

En aéronautique, la rotation autour de l'axe x est appelée roulis. Ce mouvement est assuré par la différence des forces (F_2, F_4) produites par les actionneurs droit et gauche. Cette différence de forces produit un couple $\tau\phi$ autour de l'axe x.

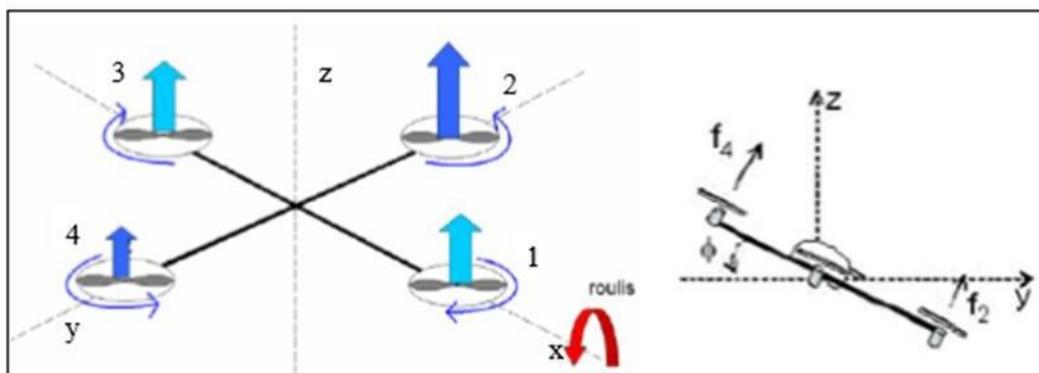
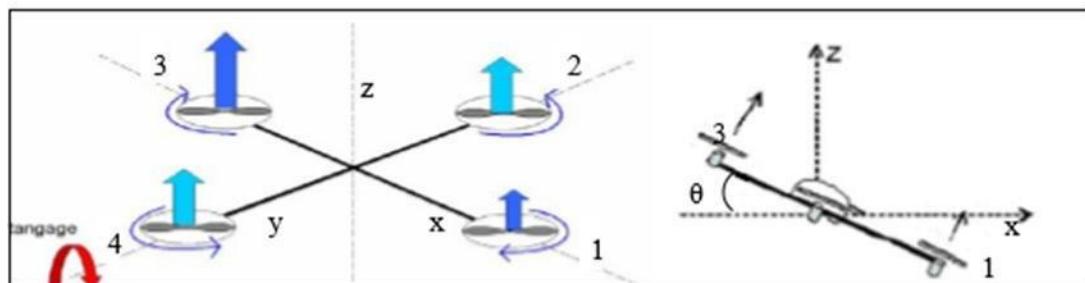


Figure 2.9 : Mouvement de roulis (ϕ)

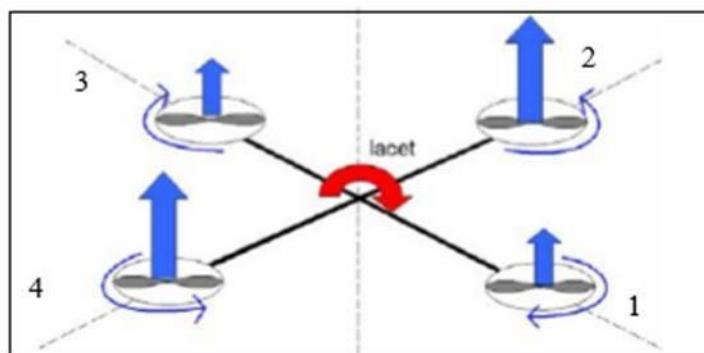
2.3.3.2 Mouvement de tangage (θ)

La rotation autour de l'axe y est appelée tangage en aéronautique. Ce mouvement est assuré par la différence des forces (F_1, F_3) produites par les actionneurs avant et arrière. Cette différence de forces produit un couple $\tau\theta$ autour de l'axe y.

Figure 2.10 : Mouvement de tangage (θ)

2.3.3.3 Mouvement de lacet (ψ)

La rotation autour de l'axe z est appelée lacet en aéronautique. Ce mouvement est assuré par la somme des couples de traînée (Q_i) produits par les quatre actionneurs. Étant donné que les sens de rotation des actionneurs (M1, M3) et (M2, M4) sont opposés, nous pouvons régler la somme des quatre couples résistants. Quand les quatre rotors tournent à la même vitesse, ils sont soumis au même couple résistant, donc la somme est nulle. Par conséquent, il n'y a pas de rotation autour de l'axe z. Cependant, si nous provoquons une différence de vitesse entre les moteurs tournant en sens opposé, les couples résistants provoquent un couple $\tau\psi$ autour de l'axe z, provoquant ainsi la rotation de l'engin.

Figure 2.11: Mouvement de lacet (ψ)

2.3.3.4 Le vol stationnaire

En montée verticale et après avoir franchi le seuil définissant l'effet du sol, le quadrirotor peut rester en vol stationnaire à une certaine hauteur constante par rapport au sol en ayant une vitesse de translation nulle. La force de sustentation doit alors équilibrer le poids mg du quadrirotor. Dans ce mode de vol, le quadrirotor a la liberté de faire des rotations autour de l'axe z qui, dans ce cas, coïncide avec l'axe z_n du système de coordonnées inertiel.

2.3.3.5 Les translations verticales

Afin de planer, toute la force de portance devrait seulement être le long de l'axe z avec une grandeur exactement opposée à la force de pesanteur. Par conséquent, la poussée produite par chaque rotor doit être identique. Les mouvements ascendants et descendant sont obtenus par la variation de la vitesse de rotation des moteurs par conséquence la poussée produite, si la force de portance est supérieure au poids du quad-copter le mouvement est ascendant, et si la force de portance est inférieure au poids du quad-copter le mouvement est descendant.

2.3.3.6 Les translations horizontales

Elles sont définies de façon similaire aux translations verticales mais cette fois-ci dans le plan xnyn. Lorsqu'une translation est effectuée suivant la direction xn et la force de portance maintient le système à une hauteur constante par rapport au sol. Le système effectue un vol connu dans la littérature sous le nom de "vol en palier".

2.4 Effets physiques agissants sur le quadrirotor

2.4.1 Les forces

Les forces qui agissent sur le système sont :

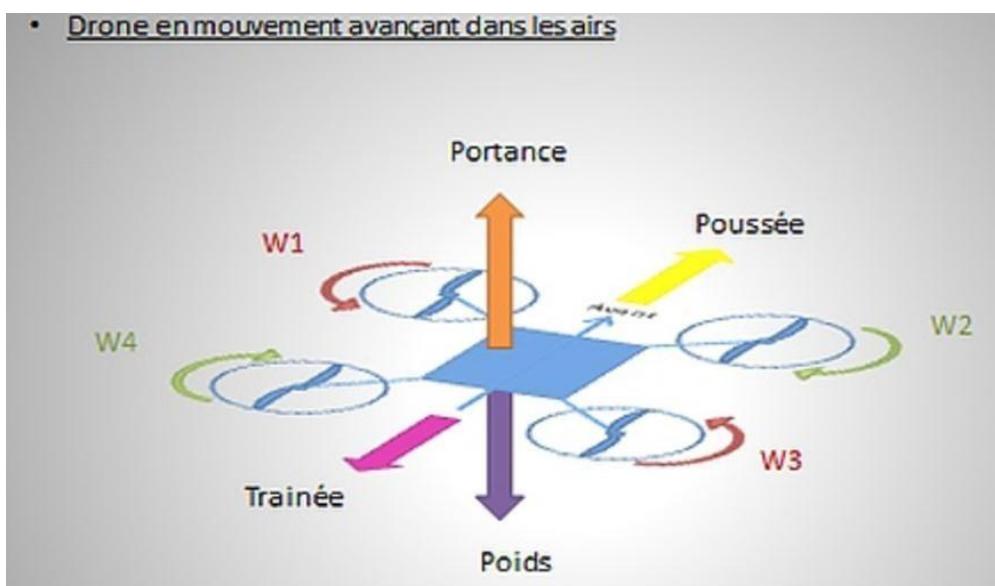


Figure 2.12 : Les forces qui agissent sur le drone

2.4.2 Le poids

Comme tous les corps possédants une masse le drone subit une attraction gravitationnelle exercée par la terre. Plus sa masse sera importante plus son poids sera élevé. Cette force maintient le drone sur le sol.

Elle s'exprime par la formule :

Poids= Masse. Intensité de pesanteur le poids en Newton la masse en Kg l'intensité de pesanteur en N/Kg

2.4.3 La portance

C'est une force qui permet au drone de décoller lorsqu'elle devient supérieure à son poids. Elle s'exerce sur un corps en mouvement dans un fluide, qui s'exerce perpendiculairement à la direction du mouvement. Dans le cas du drone la portance s'exerce sur ses hélices, le vent rencontre l'hélice (la partie supérieure se nomme extrados et la partie inférieure intrados). L'aire passant au niveau de l'extrados doit parcourir un chemin plus long ainsi la vitesse est plus importante et la pression y regnant est plus faible. (Dépression) L'aire passant au niveau de l'intrados a une vitesse plus faible et ainsi la pression qui y règne sera plus importante. (Surpression) L'addition de la surpression et de la dépression donne la portance qui aspire l'hélice vers le haut.

C'est une force multifactorielle qui dépend de : -la masse volumique de l'air ρ en kg/m³. La densité de l'air affecte les performances de l'avion. La densité dépend de la pression, de la température et de l'humidité donc de l'altitude. -la vitesse V du vent en m/s -la surface de l'aile en m² -le coefficient de portance Cp (qui dépend de l'angle d'incidence , de la forme de l'aile de l'état de sa surface) La formule permettant de déterminer la force de la portance Fz d'un drone est : $F_p = \frac{1}{2} \rho \times V^2 \times S \times C_p$

2.4.4 La trainée

C'est une force de résistance exercée par le vent sur un corps s'opposant à son déplacement. Elle permet ainsi de faire reculer le drone en augmentant la vitesse des moteurs avant . Elle est parallèle au vent qui s'oppose au déplacement. La formule qui permet de calculer cette force est la suivante : $F_t = \frac{1}{2} \rho \times V^2 \times S \times C_t$ Ft Force de trainée en N Ct Coefficient de trainée V vitesse du vent

2.4.5 La traction ou la poussé

Cette force est créée par le moteur. Elle résulte de l'accélération des deux moteurs arrières permettant lorsqu'elle est supérieure à la force de trainée de faire avancer le drone, elle s'oppose

donc à la force de trainée. Elle peut être déterminée par la formule suivante : $F_t = \rho \times C_t \times n^2 \times D^4$
 n^2 vitesse de rotation de l'hélice en tours / seconde D diamètre de l'hélice C_t coefficient de traction

2.5 Partie puissance

2.5.1 Simulation de moteur brushless en boucle ouverte

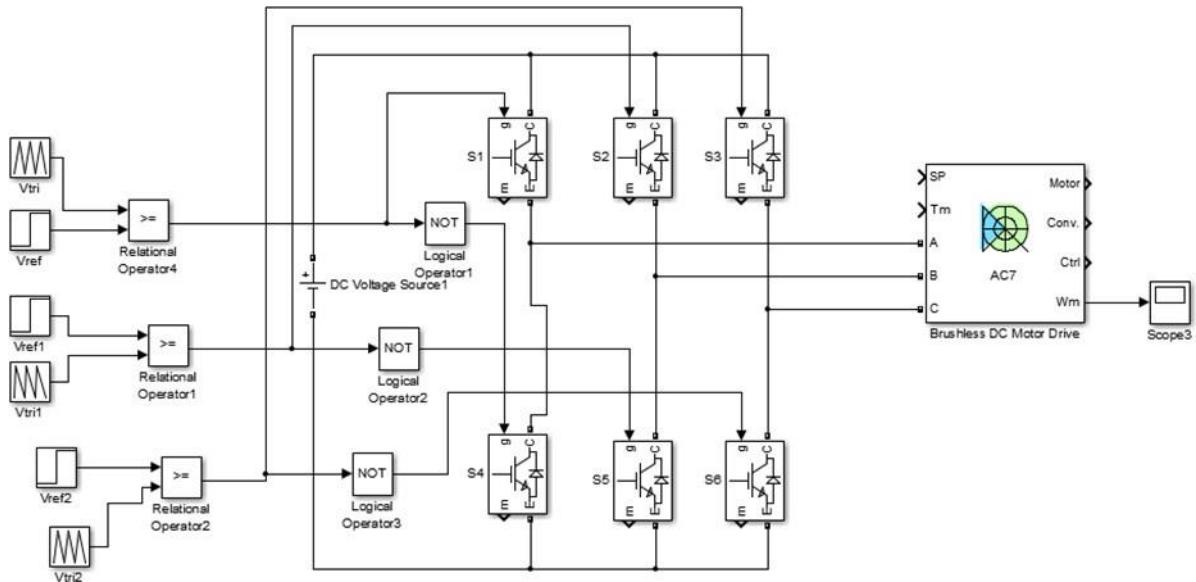


Figure 2.14:schéma de bloc de moteur BLDC sur Matlab/simulink

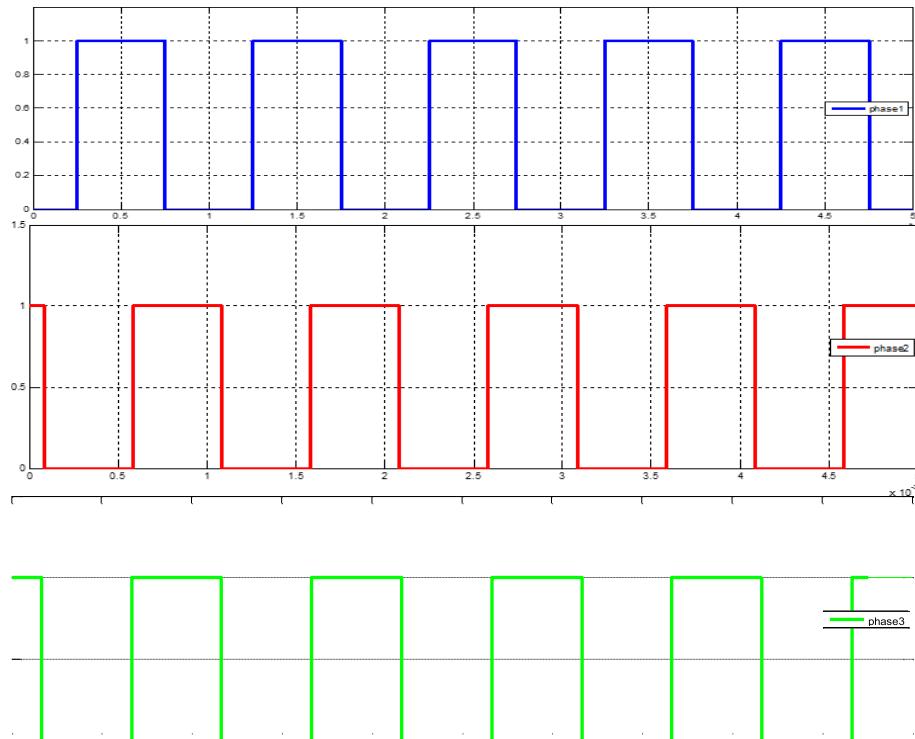


Figure 2.13 : signaux de commande de l'onduleur diphasé avec 120 °

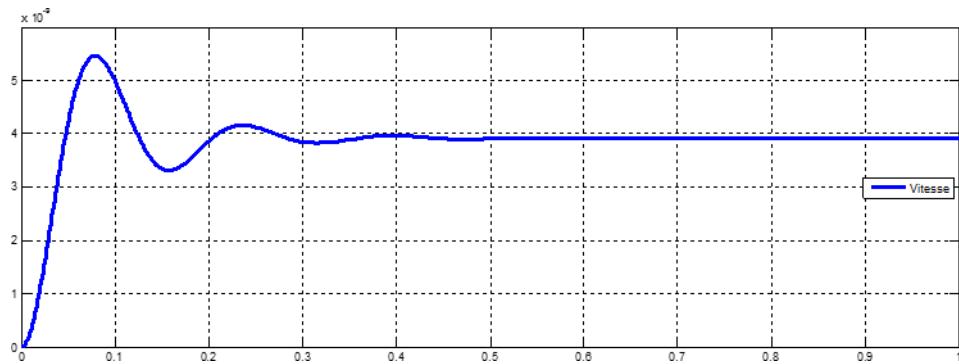


Figure 2.16 : Simulation de moteur brushless en boucle ouvert

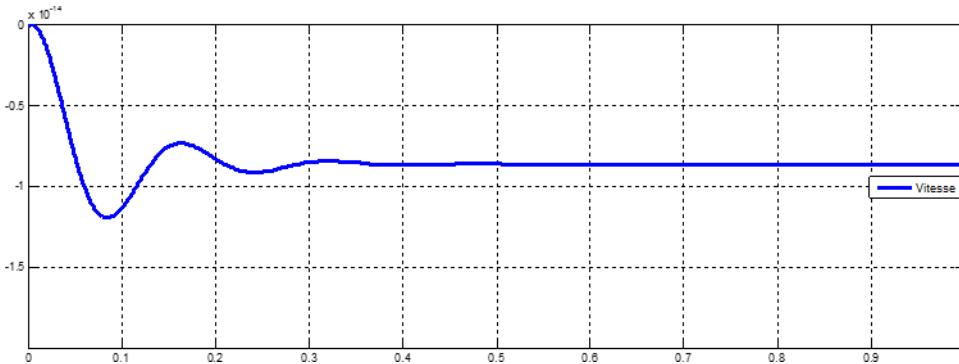


Figure 2.15: Simulation de moteur brushless en boucle ouverte dans le sens inverse

2.5.2 Simulation de moteur brushless en boucle fermée

2.5.2.1 Régulateur proportionnel

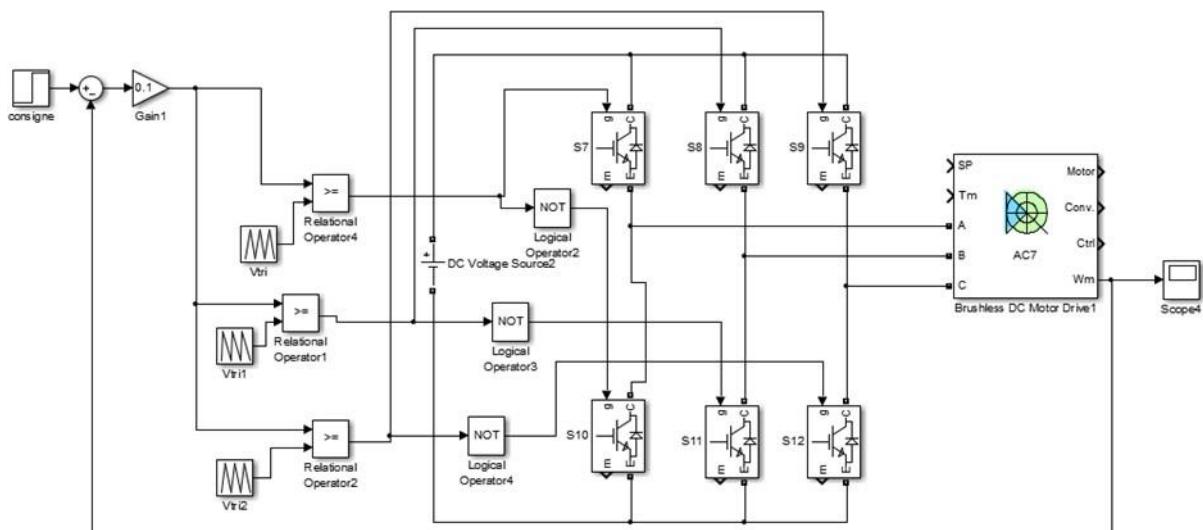


Figure 2.17 : schéma de bloc de BLDC moteur avec régulateur P

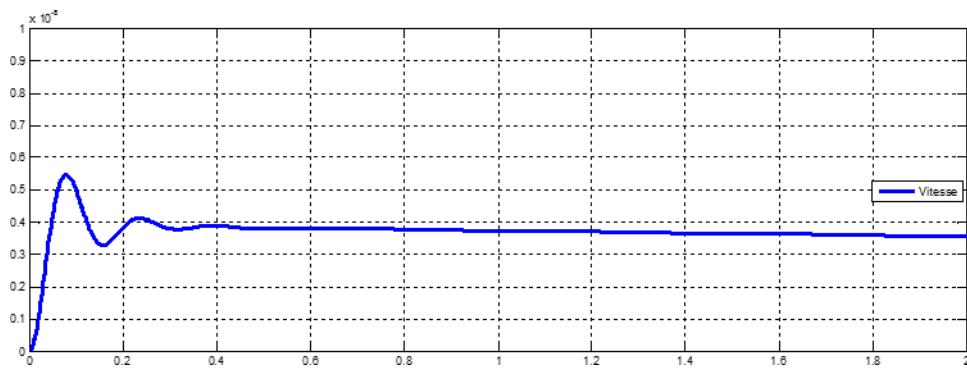


Figure 2.18 : simulation de BLDC moteur avec régulateur P

- ✚ Avec perturbation : de 10 en $t = 0 . 2$

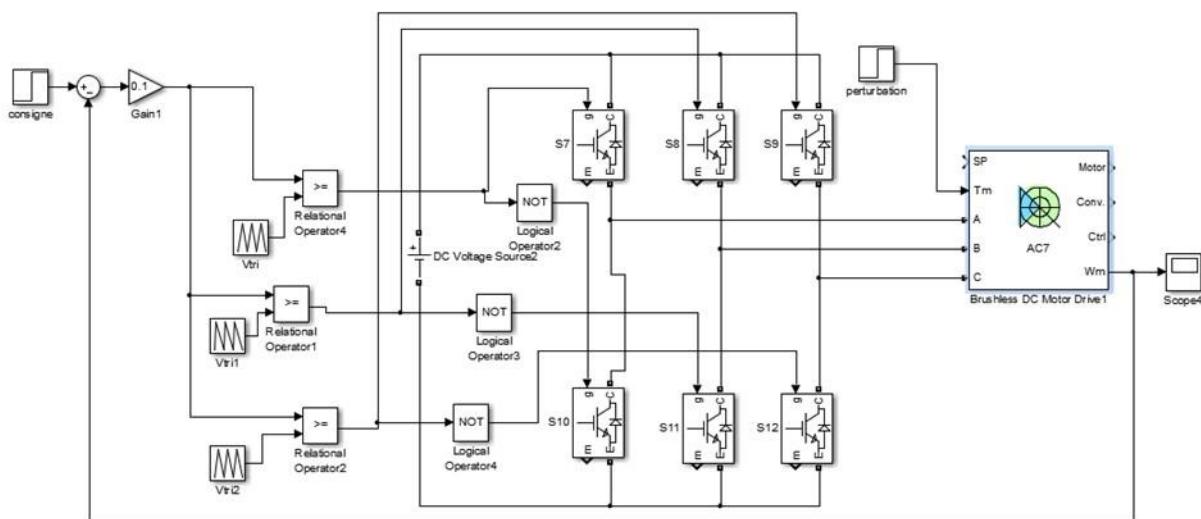


Figure 2.20 : schéma de bloc de BLDC moteur avec régulateur P avec perturbation

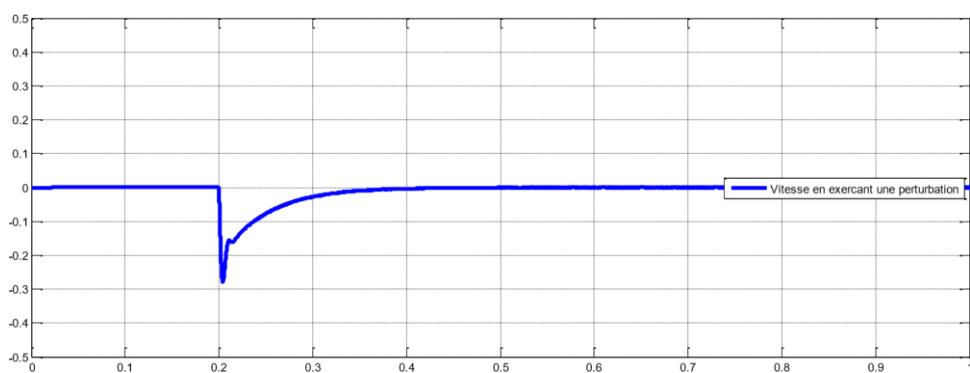


Figure 2.19: simulation de BLDC moteur avec régulateur P avec perturbation

2.5.2.2 Régulateur pi

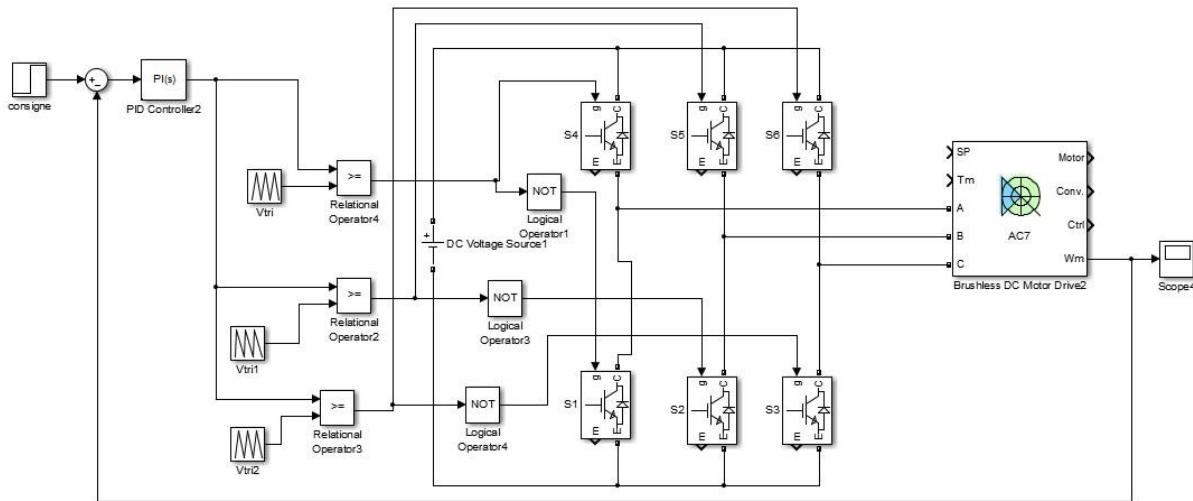


Figure 2.21: schéma de bloc de BLDC moteur avec régulateur Pi

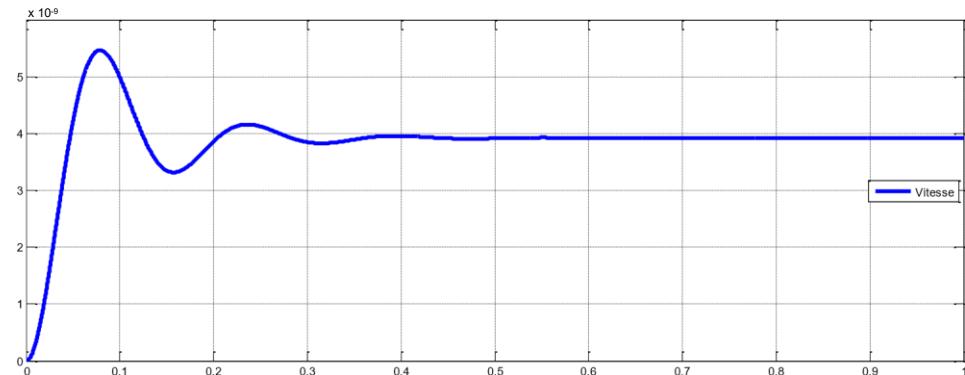


Figure 2.22: simulation de BLDC moteur avec régulateur Pi

✚ Avec perturbation : de 10 en t=0 .2

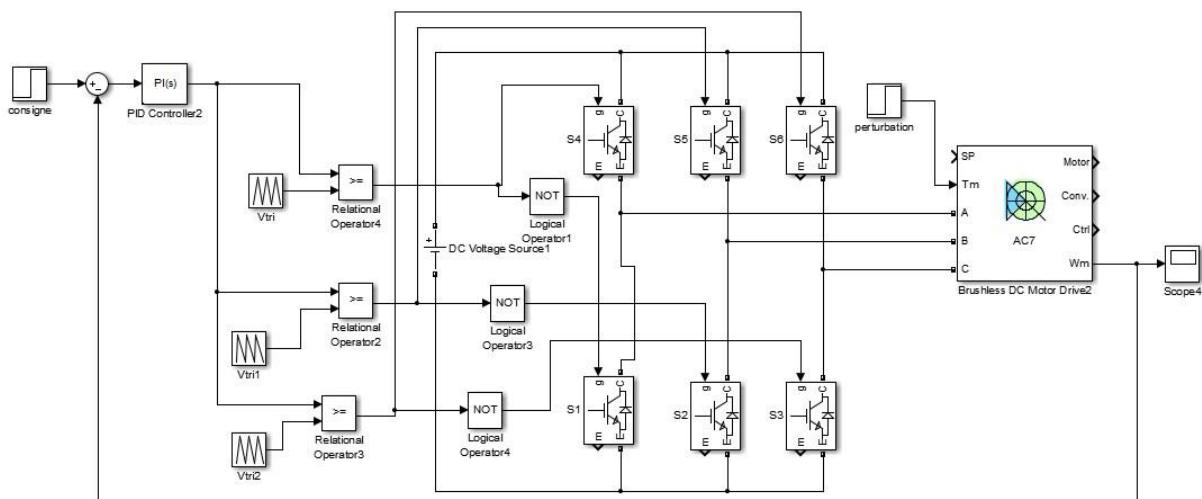


Figure 2.23 : schéma de bloc de BLDC moteur avec régulateur Pi avec perturbation

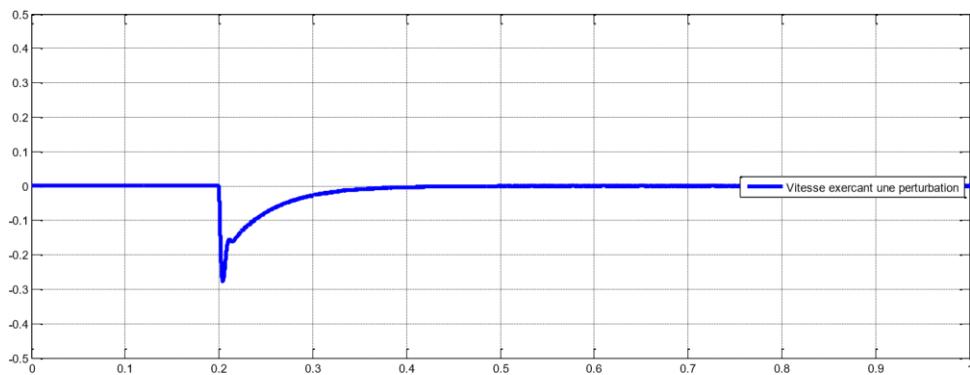


Figure 2.24 : simulation de BLDC moteur avec régulateur Pi avec perturbation

2.5.2.3 Régulateur PID

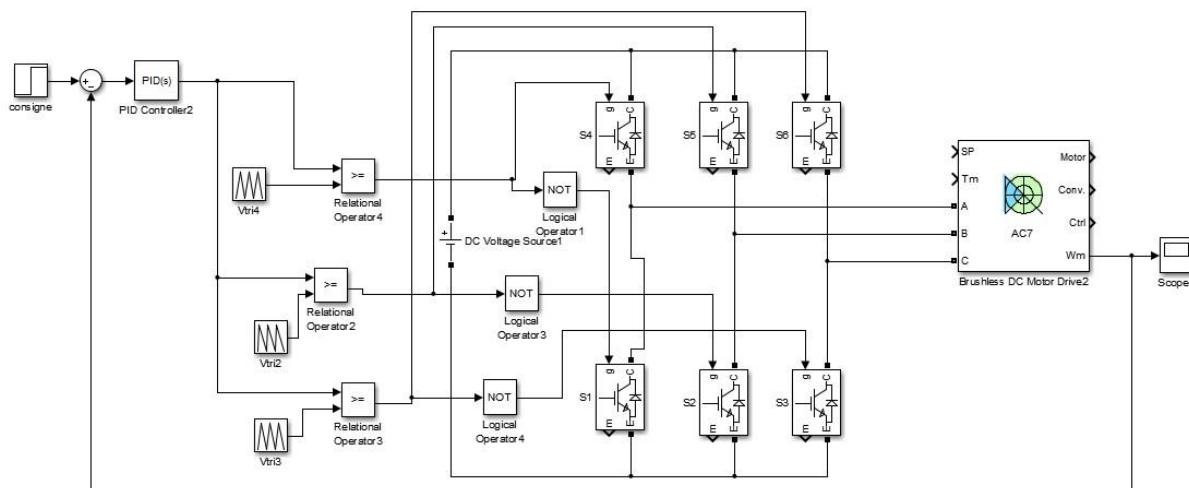


Figure 2.25 : bloc de schéma de moteur bldc avec pid

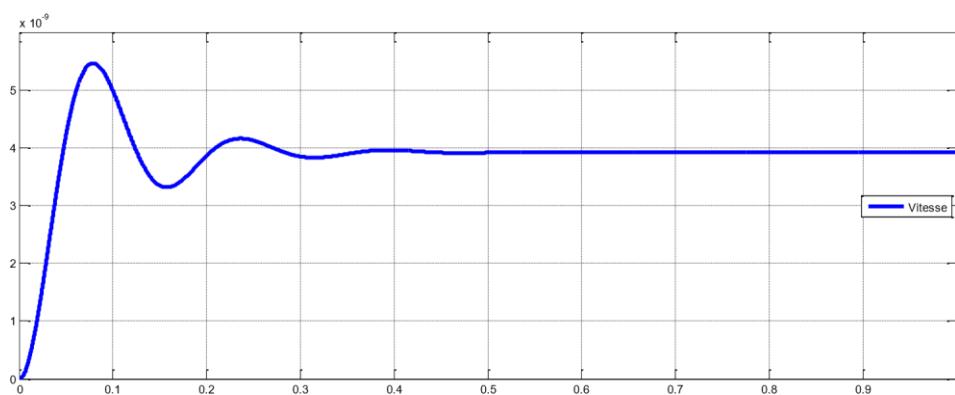


Figure 2.26 : simulation de moteur bldc avec pid

✚ Avec perturbation : de 10 en t=0 .2

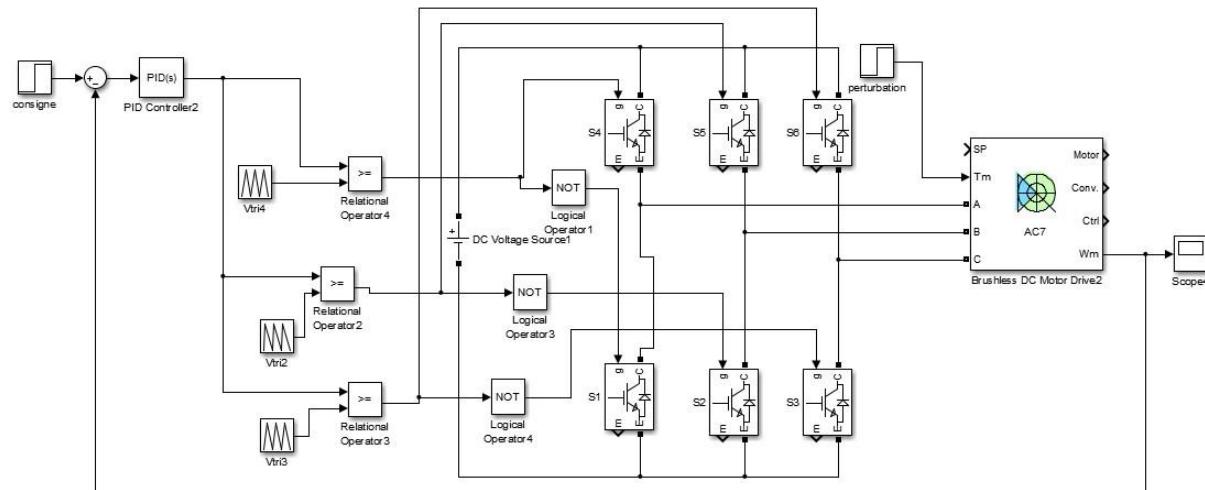


Figure 2.28 : bloc de schéma de moteur bldc avec pid avec perturbation

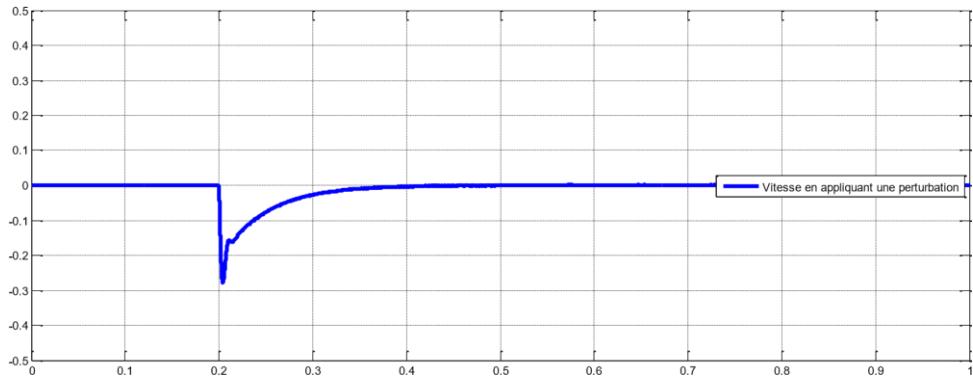


Figure 2.27 : simulation de moteur bldc avec pid avec perturbation

2.6 Partie commande

2.6.1 Moteur brushless

2.6.1.1 Simulation de moteur Brushless en boucle ouverte

La simulation d'un moteur brushless en boucle ouverte analyse le comportement du moteur sans rétroaction pour ajuster les paramètres de contrôle. Les signaux de commande sont directement appliqués au moteur, permettant d'évaluer ses performances de base, de comprendre son comportement dynamique, et d'identifier des caractéristiques telles que la vitesse, le couple et les courants de phase. Cette étape est essentielle avant d'adopter des stratégies de contrôle plus avancées, comme le contrôle en boucle fermée.

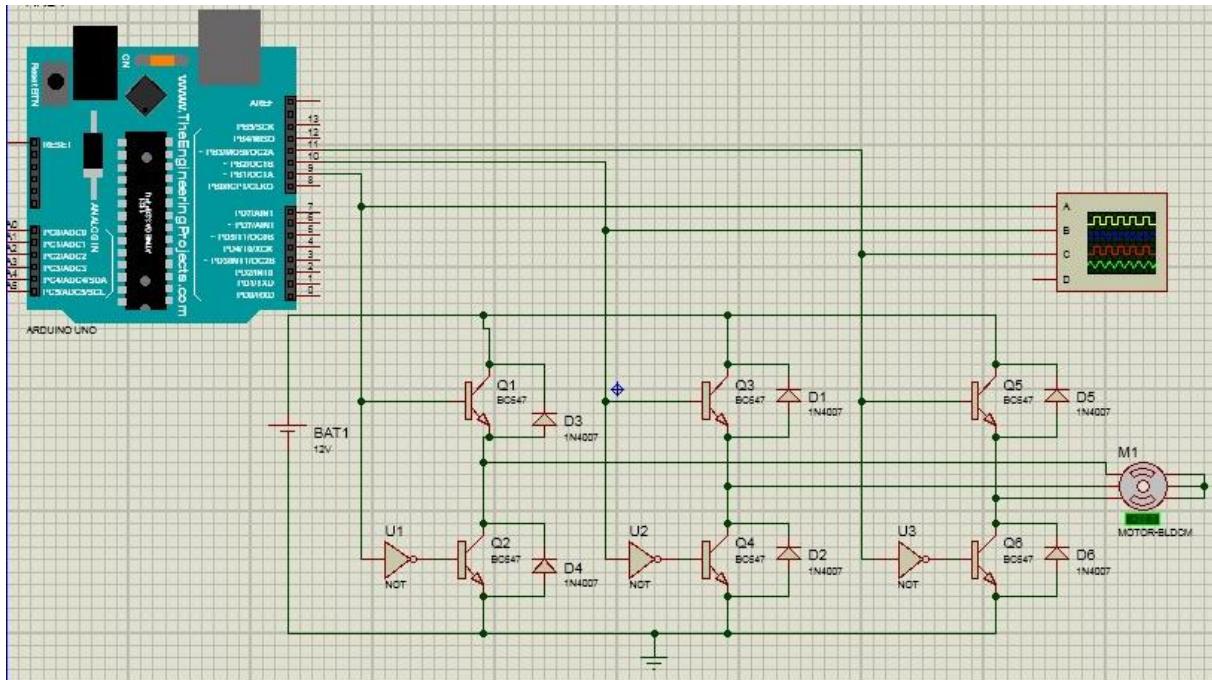


Figure 2.31 : bldc avec onduleur (transistors BC647) sur proteus

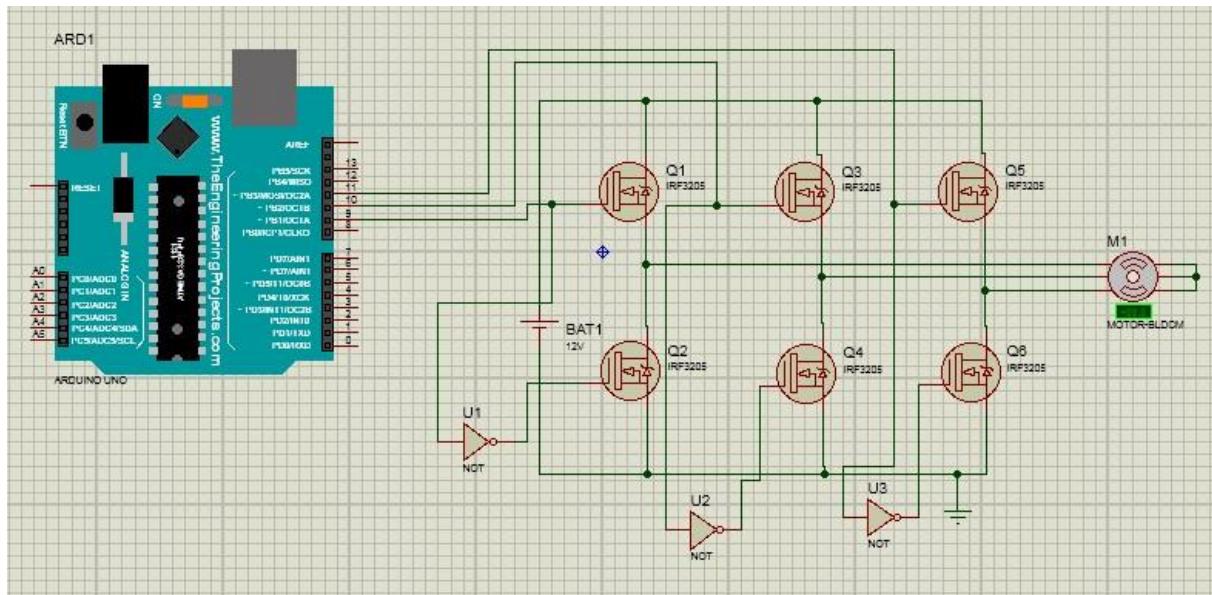


Figure 2.30 : bldc avec onduleur(transistors MOSFET) sur proteus

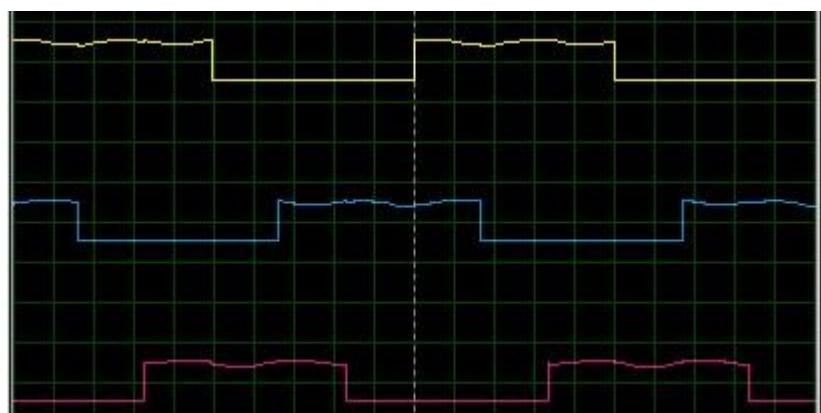


Figure 2.29 : signaux de commande d'onduleur

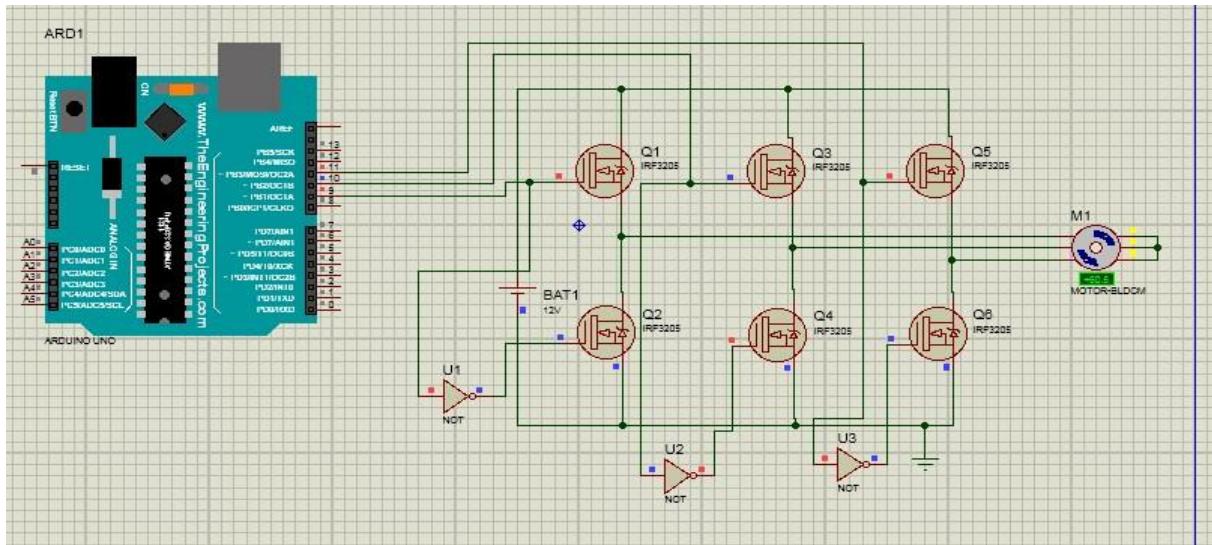


Figure 2.34 : simulation de bldc avec onduleur (transistors MOSFET) sur proteus

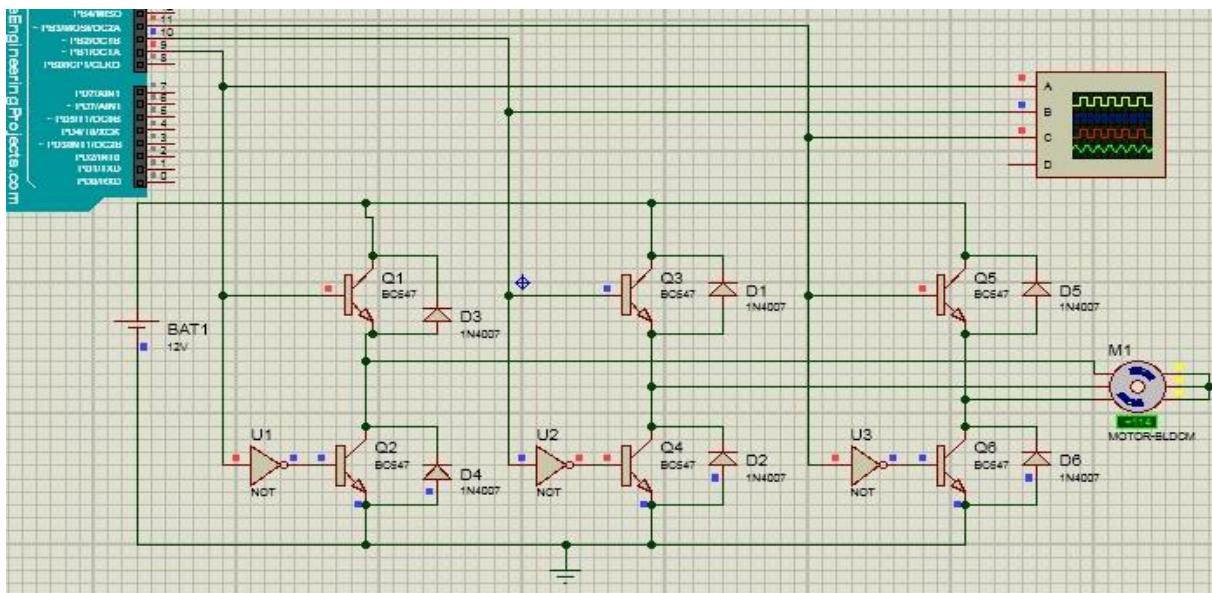


Figure 2.33 : simulation de bldc avec onduleur (transistors BC647) sur proteus

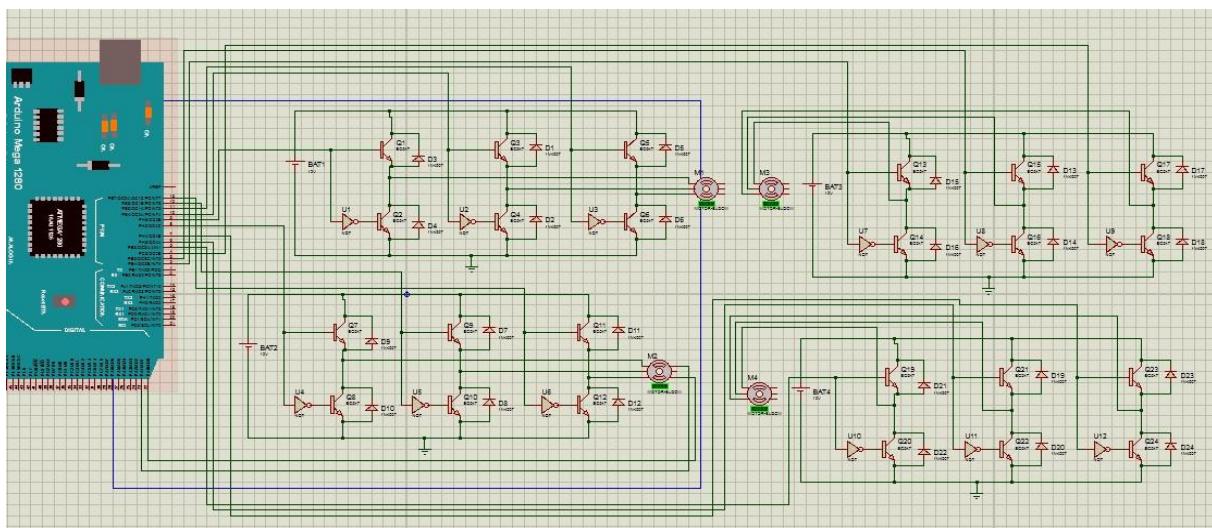


Figure 2.32 : simulation de 4 moteurs bldc sur Proteus

2.6.1.2 Simulation de moteur Brushless en boucle Fermée

- Calcule de la vitesse de moteur brushless

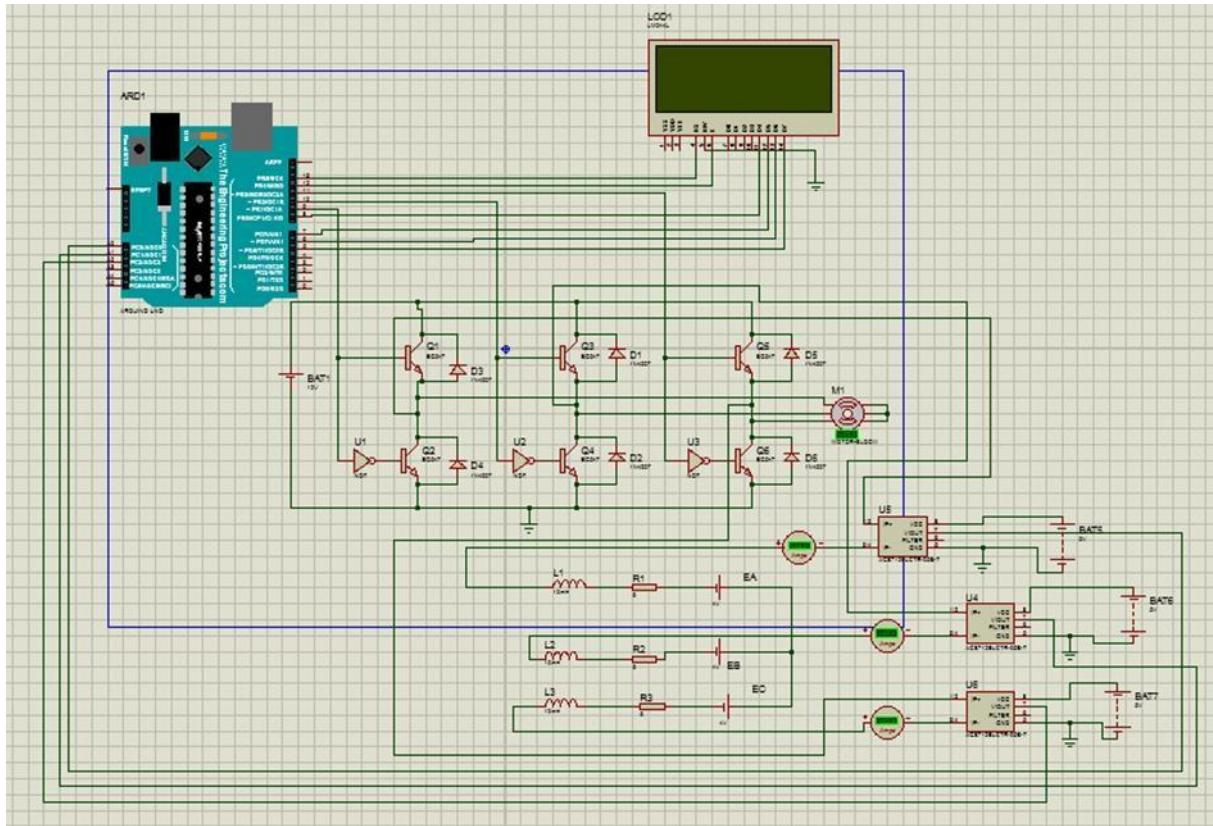


Figure 2.35 : Schéma pour le calcul de la vitesse du moteur

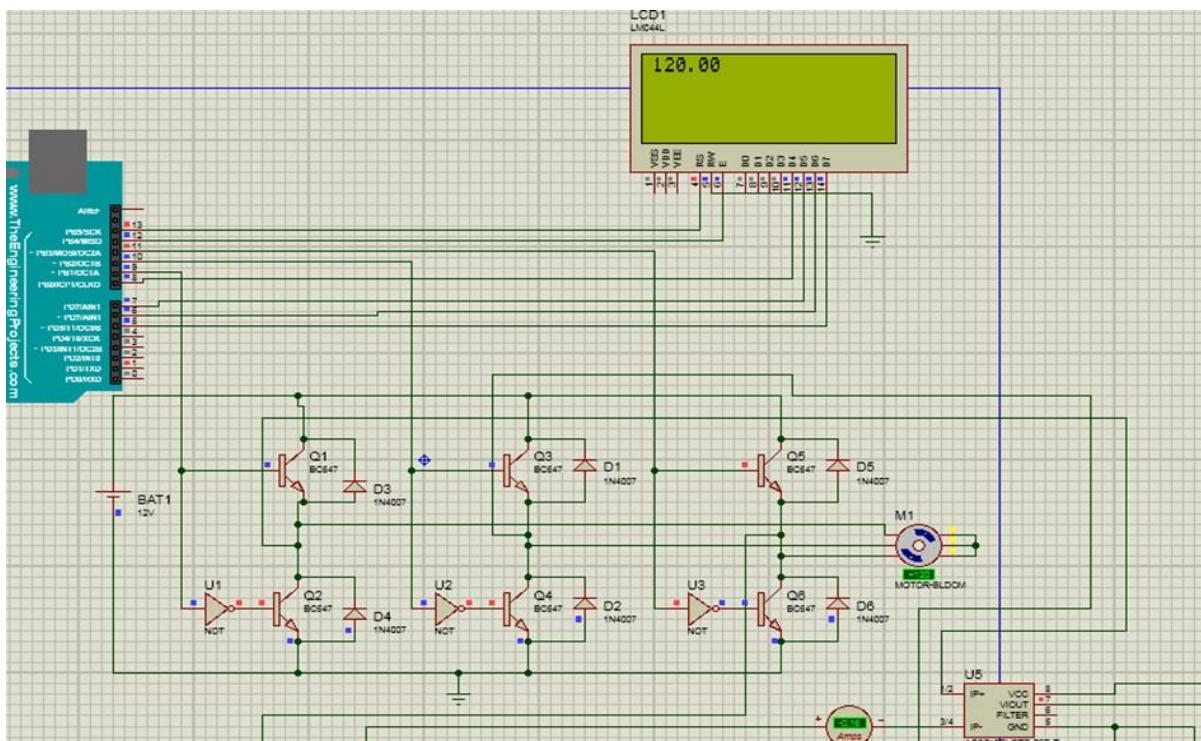


Figure 2.36 : Résultat de simulation de Schéma pour le calcul de la vitesse du moteur

Asservissement du système avec le régulateur Proportionnel avec $k_p=50$

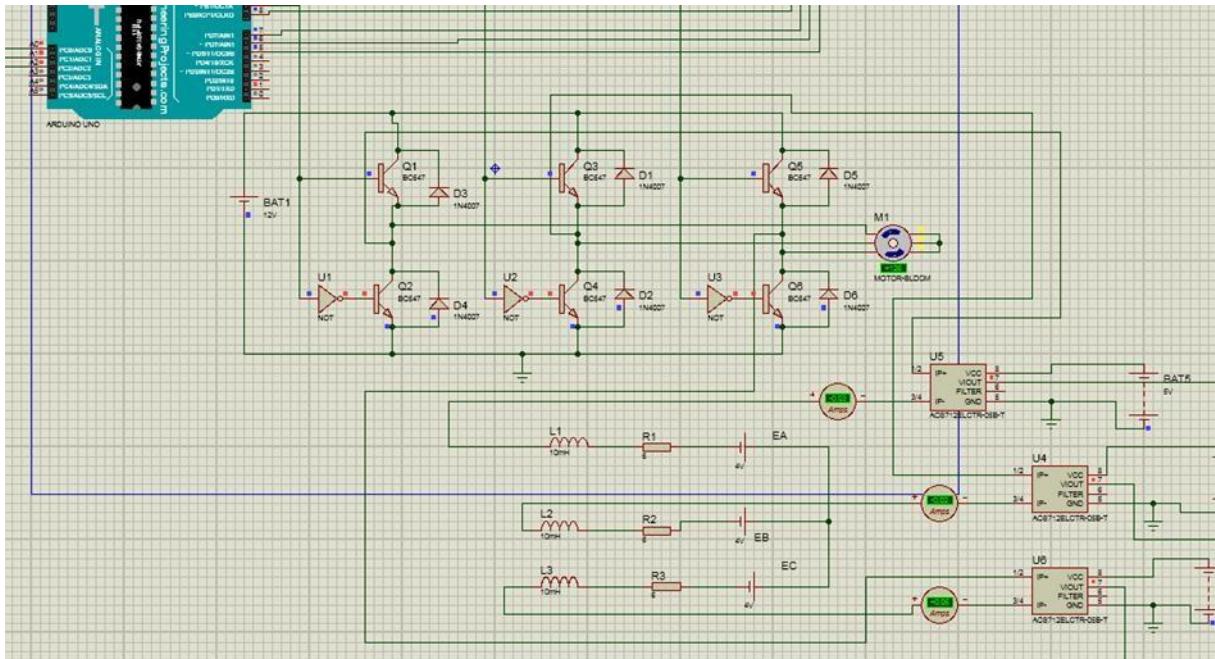


Figure 2.37 : Simulation de moteur Brushless avec Régulateur P

Asservissement du système avec le régulateur Pi avec $k_p=50$ et $k_i=0.1$

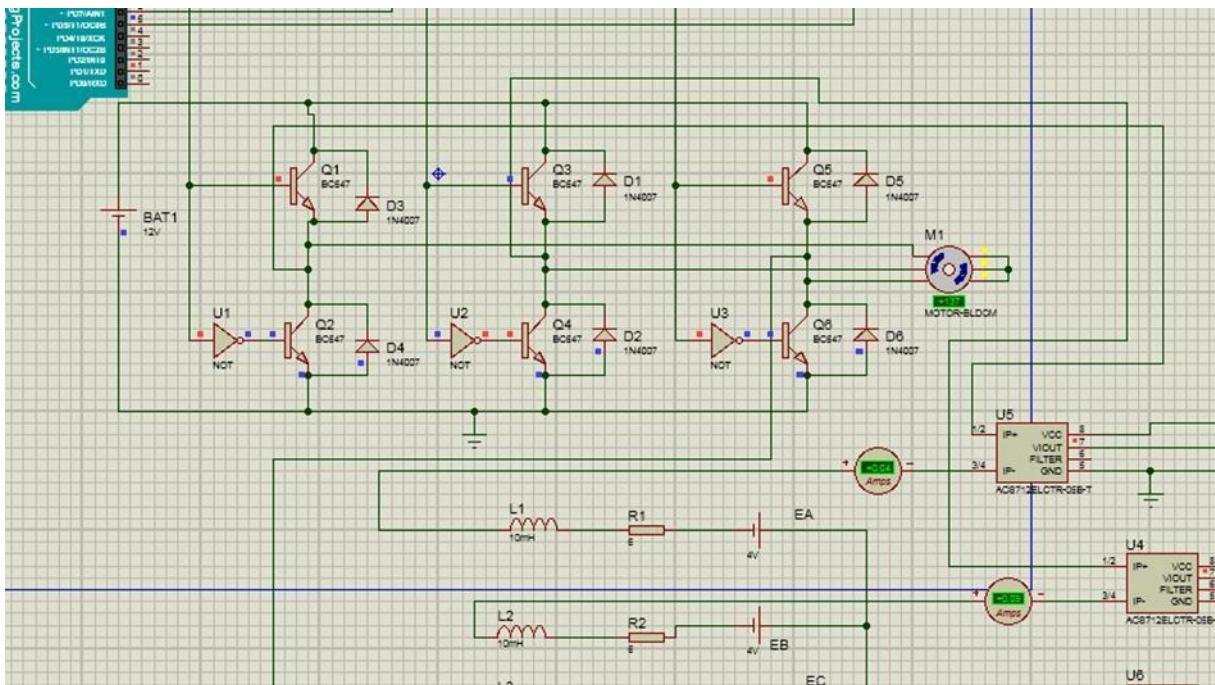


Figure 2.38 : Simulation de moteur Brushless avec Régulateur Pi

Asservissement du système avec le régulateur PID avec $k_p=50$ et $k_i=0.1$ et $k_d=1$

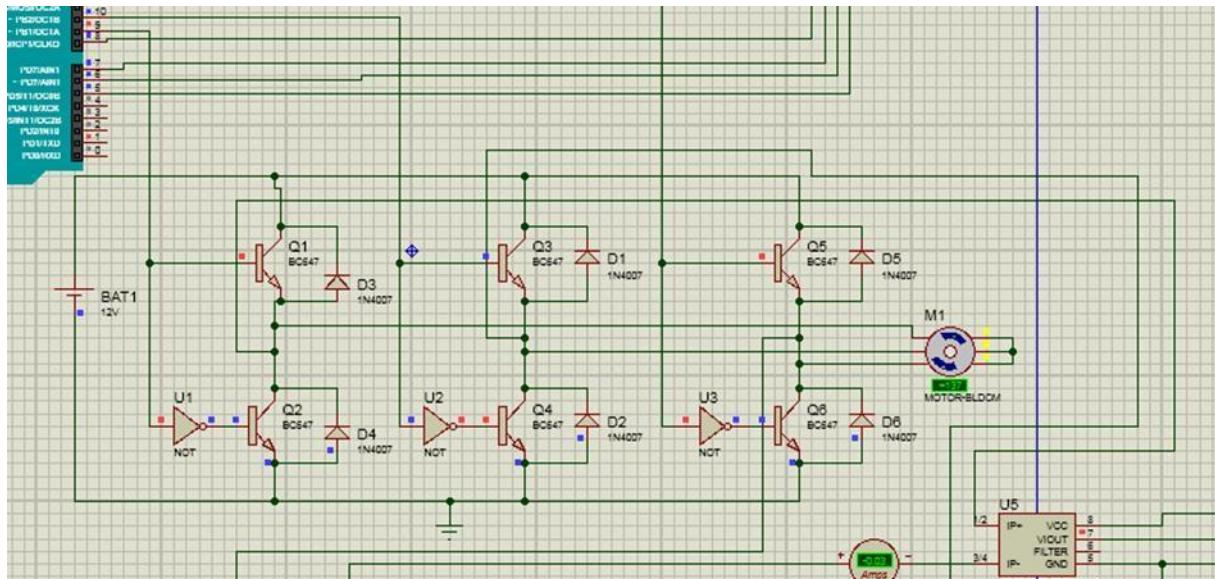


Figure 2.39 : imulation de moteur Brushless avec Régulateur PID

2.6.1.3 Le Module MPU6050

Avant de faire l’interfaçage de la carte avec le module MPU6050 il faut tout d’abord configurer le Protocole I²C sur la carte

Configuration de protocole I²C :

I²C est une norme très couramment utilisée conçue pour permettre à une puce de communiquer avec une autre. Ainsi, nous pouvons le connecter à une variété de puces et de modules compatibles I²C.

Le schéma ci-dessous représente l’interfaçage du module MPU6050 avec le module Arduino Uno :

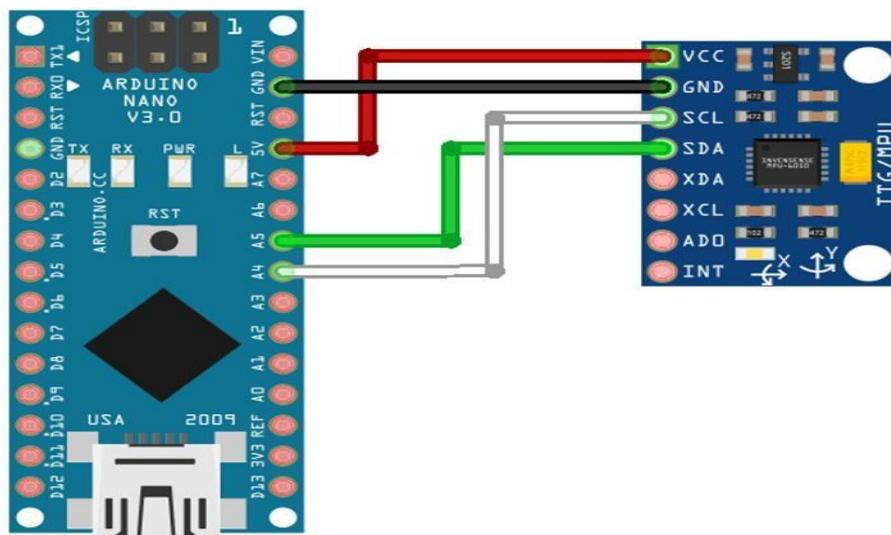


Figure 2.40 : l’interfaçage du module MPU6050 avec Arduino

Code :

Ce code va nous permettre d'afficher les valeurs de gyroscopes et de l'accéléromètre ainsi que la température ambiante.

2.6.1.4 Vérification de niveau de batterie

La notion de vérification de niveau de batterie est très importante pour un système autonome car le fonctionnement de tous les composants du système est relié à la présence d'une alimentation permanente

Pour faire cette vérification sous le simulateur Protues on va utiliser un LED BAR qui contient un ensemble de LED qui indique à chaque fois l'état de la batterie ,une LCD va nous permettre d'afficher cette valeur dans le programme principale ,cette valeur va être par la suite entrée dans le code comme première condition à vérifier avant d'exécuter le reste de programme car si le niveau de batterie est inférieur au valeur préciser le programme renvoie un erreur de low battery ,afin de garantir la sécurité des utilisateurs du drone et de matérielle

Le schéma :

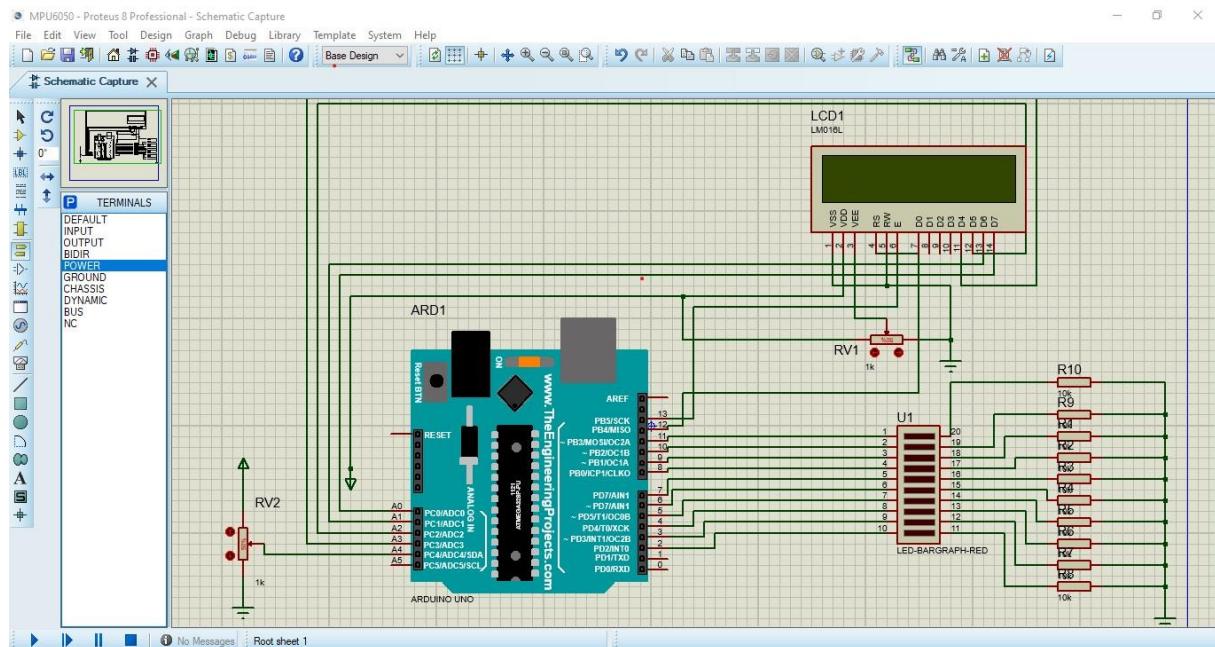


Figure 2.41 : Schéma pour la vérification de batterie

Code : (Voir Annexe)

Le présent code va nous permettre de jouer sur les LEDs de bar des LEDs selon les tensions en entrée, pour toute augmentation de niveau tension en entrée, un LED s'allume.

Conclusion

Le quadrirotor est l'un des drones les plus populaires et séduisants en raison de son élégance et de sa dynamique. Dans ce chapitre, nous avons décrit les mouvements de base du quadrirotor et présenté son modèle dynamique en utilisant le formalisme de Newton-Euler. Ce modèle a ensuite été simplifié en adoptant des hypothèses pour faciliter la synthèse des lois de commande.

Concernant la commande, nous avons présenté le schéma de commande PID appliqué au quadrirotor, qui assure la stabilisation en vol stationnaire. Nos objectifs pour la conception et la construction de notre drone étaient ambitieux.

Chapitre 3 : Outils Matériel et Réalisation

Introduction

La réalisation d'un drone nécessite divers outils et composants matériels, notamment un châssis léger, des moteurs « brushless », un contrôleur de vol, des régulateurs de vitesse électroniques (ESC), des hélices, une batterie « LiPo », et un système de transmission radio. Le processus implique la conception et la planification du drone, l'assemblage du châssis et des moteurs, l'installation des ESC et du contrôleur de vol, la connexion de la batterie et du récepteur radio, suivis de la calibration et des tests au sol avant le premier vol. Les drones peuvent également inclure des modules GPS et des caméras pour des fonctions avancées.

3.1 Outils matériels

Le choix des matériaux est crucial pour la construction d'un drone, car il garantit une structure à la fois légère et robuste, capable de résister à des atterrissages difficiles. Un bon équilibre entre poids et résistance est essentiel pour maximiser l'efficacité du vol et la durabilité du drone. Nous allons explorer les différents matériaux utilisés dans la conception de notre drone, et les plastiques renforcés qui offrent une bonne résistance aux chocs et une flexibilité accrue.

3.1.1 Moteur sans balais

Ce type de moteur est un moteur à courant continu, mais avec une différence notable : le collecteur et l'emplacement des bobines et des aimants permanents sont inversés. Il est constitué de deux parties principales : le rotor et le stator.

- **Le stator :** Cette partie fixe a pour rôle de créer un flux magnétique. Cela peut être réalisé soit par un aimant permanent, soit par un courant électrique circulant dans un bobinage.
- **Le rotor :** Également connu sous le nom d'induit, le rotor est un composant mobile composé d'un châssis métallique avec des encoches où sont placés les bobinages.

Le stator, en tant qu'élément fixe, génère le champ magnétique nécessaire, tandis que le rotor, en rotation, interagit avec ce champ pour produire le mouvement du moteur.

Le moteur est doté d'un arbre en acier trempé de 3,2 mm de diamètre, assurant une résistance et une durabilité élevées. Pour garantir une rotation fluide et réduire l'usure, il est également équipé de roulements à billes. Les trous de fixation sont conçus avec un espacement de 16 mm

et 19 mm, permettant une flexibilité dans le montage, et sont filetés pour des vis de 3 mm (M3). Ces caractéristiques facilitent l'intégration du moteur dans diverses configurations mécaniques, offrant à la fois robustesse et adaptabilité.

Tableau 3.1 : Caractéristiques

Caractéristique	Détail
Kv	1000 RPM / V
Efficacité maximale	80%
Max efficacité actuelle	4 – 10A (> 75%)
Courant à vide	0.5A pour 10V
Résistance	0.090 Ohm
Courant max	13A pendant 60s
Max Watts	150W
Poids	51 g / 1.86 oz
Taille	28 mm de diamètre x 28 mm de longueur cloche
Longueur totale	40 mm
Diamètre de l'arbre	3,2 mm
Pôles	14
Modèle Poids	300 – 800g / 10,5 – 28.2 oz



Figure 3.1 : Moteur sans balais

3.1.2 Contrôleur électronique de la vitesse

Un ESC, acronyme de « Electronic Speed Controller », est un variateur de vitesse électronique. Ce dispositif agit principalement comme un régulateur électronique de puissance pour moteurs électriques, spécialement conçu pour piloter des moteurs à courant continu, tels que les moteurs brushless triphasés.

Un contrôleur ESC se présente typiquement sous la forme d'une petite boîte rectangulaire, avec plusieurs fils entrants et sortants de chaque côté. La conception est généralement standardisée, mais il existe quelques variantes. Par exemple, certains ESC peuvent être équipés d'un ventilateur de refroidissement pour dissiper la chaleur, de condensateurs de puissance pour stabiliser le courant, ou d'un régulateur de tension déporté pour alimenter d'autres composants du système.

Le rôle principal d'un ESC est de contrôler la vitesse du moteur en modulant la puissance électrique transmise, ce qui permet des ajustements précis de la vitesse de rotation du moteur. En plus de la régulation de vitesse, un ESC peut également offrir des fonctions supplémentaires telles que le freinage, la marche arrière, et la protection contre les surcharges et les surtensions. Ces caractéristiques font de l'ESC un composant essentiel dans la gestion des performances et de la sécurité des systèmes motorisés.

➤ **Les caractéristiques :**

Courant constant : 30A (Max 40A<10s).

BEC : 5V 3A.

Tension d'entrée : 11.1 ~ 11.7 V.

Batterie Li-Po: 2S ~ 3S.



Figure 3.2 : Contrôleur électronique de la vitesse

3.1.3 Module d'alimentation gm v1.0

Le module d'alimentation GM V1 est une solution pratique et efficace pour alimenter notre contrôleur de vol APM 2.5 avec une source d'énergie propre issue d'une batterie LiPo. Ce module ne se contente pas de fournir une alimentation stable et fiable, il est également capable de mesurer la consommation de courant et la tension de la batterie, offrant ainsi des informations précieuses pour la gestion de l'énergie de notre système. Toutes ces données sont transmises via un câble à 6 positions, facilitant la connexion et l'intégration avec le contrôleur

de vol. Ce dispositif assure non seulement une alimentation électrique constante, mais aussi un suivi précis des performances de la batterie, contribuant ainsi à optimiser l'efficacité et la sécurité de nos opérations de vol.

➤ **Les caractéristiques :**

Tension d'entrée maximale : 25 v

Détection de courant maximale : 90A

Taille : 25x20x10mm/1x0.8x0.4 pouces

Taille de la livraison : env. 92x145x15mm/3.6x5.7x0.6

Poids Net : 29g



Figure 3.3 : Module d'alimentation gm v1.0

3.1.4 Batterie Li-Po 11.1V 2200 mAh

➤ **Les caractéristiques :**

Marque : rhinoceros power

Composition de la pile : Lithium-polymère

Usages recommandés pour le produit : Drone

Tension : 11.1 Volts

Réutilisation : Rechargeable

Capacité de la batterie : 2200 Milliampères-heure (mAh)



Figure 3.4 : Batterie Li-Po 11.1V 2200 mAh

3.1.5 ArduPilot APM 2.8

La carte ArduPilot APM 2.8 est un contrôleur de vol polyvalent capable de piloter divers types de véhicules, y compris des avions, des hélicoptères, et des rovers. Sa grande flexibilité permet de l'adapter facilement à différentes plateformes de vol et de déplacement terrestre. Pour

configurer la carte ArduPilot APM 2.8, il suffit de télécharger le logiciel approprié en quelques clics via le logiciel de planification de mission (Mission Planner). Ce processus simplifié permet aux utilisateurs de sélectionner et d'installer rapidement le firmware spécifique à leur type de véhicule. Grâce à cette fonctionnalité, la carte peut être reprogrammée et mise à jour aisément, garantissant une gestion optimale et un contrôle précis de notre véhicule, qu'il soit aérien ou terrestre.

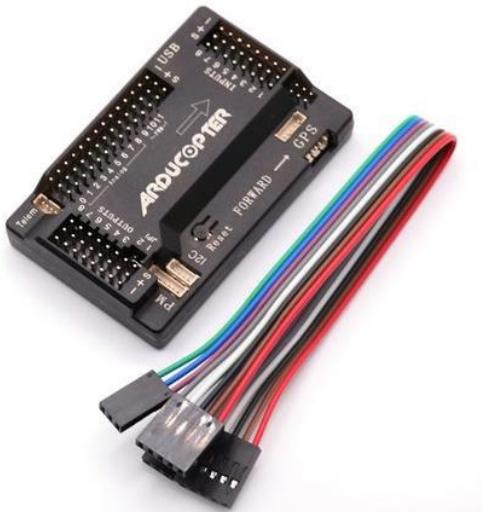


Figure 3.5 : ArduPilot APM 2.8

3.1.6 Transmetteur-recepteur Fly-sky

Le transmetteur-récepteur FlySky est un dispositif essentiel pour la communication sans fil entre le pilote et le drone. Il se compose d'un émetteur, tenu par le pilote, et d'un récepteur monté sur le drone. Le transmetteur envoie les commandes de vol, telles que la direction, la vitesse, et l'altitude, au récepteur, qui les transmet ensuite au contrôleur de vol du drone. FlySky est réputé pour sa fiabilité et sa portée étendue, offrant une connexion stable même à des distances significatives. La technologie de modulation de fréquence utilisée par FlySky réduit les interférences, assurant ainsi une communication claire et sans interruption. De plus, les émetteurs FlySky sont souvent dotés d'écrans LCD et de multiples canaux, permettant une personnalisation avancée des contrôles et un retour d'information en temps réel sur l'état du drone. Ce système de transmission robuste et flexible est indispensable pour les amateurs de drones et les professionnels, garantissant un pilotage précis et sécurisé.



Figure 3.6 : Transmetteur-récepteur Fly-sky

Tableau 3.2 : Caractéristiques du Dispositif RF

Caractéristique	Détail
Canaux	6 canaux
Puissance RF	moins de 20 dB
Modulation	GFSK
Sensibilité	1024
Port DSC	oui
Port chargeur	oui
Alimentation	12 V CC (1,5 AAA*8)
Poids	680g
Longueur d'antenne	26mm
Taille	190x80x240mm

3.1.7 Les hélices 1045

Les hélices 1045 sont des composants clés pour de nombreux drones, particulièrement appréciées pour leur performance et leur efficacité. Le chiffre "1045" indique les dimensions de l'hélice : 10 pouces de diamètre et un pas de 4,5 pouces. Ces hélices sont conçues pour offrir un équilibre optimal entre poussée et efficacité énergétique, permettant aux drones de voler de manière stable et réactive. Fabriquées à partir de matériaux robustes comme le plastique renforcé ou la fibre de carbone, elles résistent bien à l'usure et aux impacts. Les hélices 1045

sont compatibles avec une large gamme de moteurs brushless, rendant leur installation simple et rapide. Leur conception assure une bonne maniabilité et un vol fluide, même sous des charges variables. En utilisant des hélices 1045, les pilotes peuvent s'attendre à une amélioration des performances de vol, avec une augmentation de la durée de vol et une meilleure gestion de l'énergie. Ces hélices sont idéales pour les drones de taille moyenne à grande, utilisés pour des applications allant de la photographie aérienne aux courses de drones.



Figure 3.7 : les hélices des moteurs

3.1.8 Arduino nano et MPU6050

L'Arduino Nano et le capteur MPU6050 sont deux composants essentiels dans de nombreux projets de drones, offrant des fonctionnalités puissantes dans des formats compacts.

L'Arduino Nano est un microcontrôleur compact et polyvalent basé sur le microprocesseur ATmega328. Il est particulièrement apprécié pour sa taille réduite et sa facilité d'intégration dans des projets électroniques. Le Nano offre 14 broches d'entrée/sortie numériques et 8 entrées analogiques, ce qui permet de connecter divers capteurs et modules. Grâce à sa compatibilité avec l'environnement de programmation Arduino (IDE), il est facile de coder et de charger des programmes pour contrôler le drone. De plus, ses dimensions compactes permettent une intégration facile dans des espaces restreints, faisant de l'Arduino Nano un choix idéal pour les projets de drones où l'espace et le poids sont des contraintes critiques.

Le MPU6050 est un module capteur à 6 axes combinant un gyroscope et un accéléromètre, offrant une mesure précise des mouvements et de l'orientation. Il est crucial pour la stabilisation et le contrôle du drone, fournissant des données sur l'accélération linéaire et la rotation angulaire dans trois dimensions. Le MPU6050 communique avec l'Arduino Nano via le protocole I2C,

ce qui permet un échange de données rapide et efficace. En utilisant les données du MPU6050, le contrôleur de vol peut ajuster les commandes des moteurs pour maintenir l'équilibre et la stabilité du drone en vol. Grâce à sa haute précision et à son faible coût, le MPU6050 est largement utilisé dans les applications de contrôle de vol et de navigation pour drones.

En intégrant l'Arduino Nano et le MPU6050, les concepteurs de drones peuvent créer des systèmes de contrôle de vol sophistiqués, capables de maintenir une stabilité exceptionnelle tout en offrant une grande maniabilité. Ces composants permettent de réaliser des drones capables de performances élevées, tout en restant accessibles et faciles à programmer.

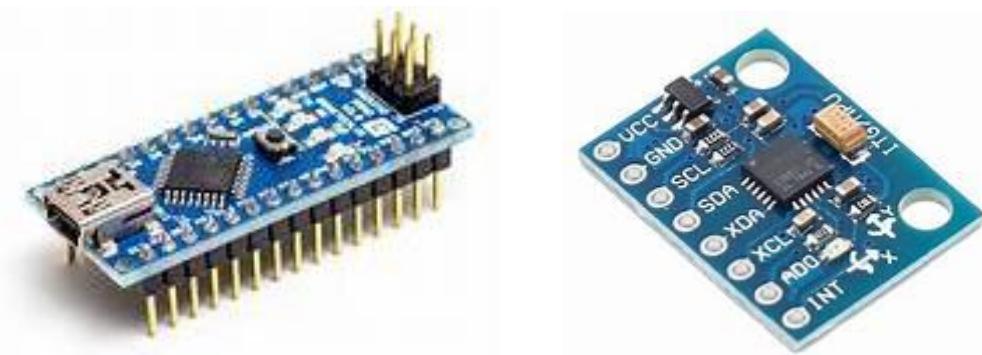


Figure 3.8 : Arduino nano et MPU6050

3.2 Assemblage du drone

L'assemblage d'un drone est un processus méthodique qui combine divers composants pour créer une machine volante efficace et stable. Tout commence par la construction du châssis, généralement en fibre de carbone ou en plastique renforcé, pour garantir à la fois légèreté et robustesse. Ensuite, les moteurs brushless sont fixés aux bras du châssis, suivis par l'installation des hélices, comme les hélices 1045, qui doivent être correctement équilibrées et orientées. Les contrôleurs de vitesse électroniques (ESC) sont ensuite connectés aux moteurs et montés sur le châssis. Au centre du drone, le contrôleur de vol, tel que l'Arduino Nano avec le capteur MPU6050, est installé pour gérer la stabilisation et le pilotage. La batterie LiPo est fixée solidement au châssis pour alimenter l'ensemble du système. Pour la communication, un récepteur radio FlySky est connecté au contrôleur de vol, permettant la transmission des commandes de l'émetteur. Une fois tous les composants fixés et connectés, des tests au sol sont effectués pour vérifier que les systèmes fonctionnent correctement et que le drone répond aux commandes. Après ces vérifications, le drone est prêt pour son premier vol d'essai, où des

ajustements supplémentaires peuvent être nécessaires pour optimiser les performances et la stabilité en vol.

3.3 L'assemblage du châssis

L'assemblage du châssis est la première étape cruciale dans la construction d'un drone, fournissant la structure de base à laquelle tous les autres composants seront fixés. Le processus commence par l'assemblage des bras et des plaques principales du châssis, souvent fabriqués en fibre de carbone ou en plastique renforcé pour assurer légèreté et robustesse. Les bras sont fixés aux plaques de base à l'aide de vis et de boulons, formant ainsi la structure centrale du drone. Une fois le cadre principal assemblé, les supports pour les moteurs sont montés à l'extrémité des bras. Il est important de vérifier que toutes les vis sont bien serrées pour éviter toute vibration ou désalignement. Ensuite, des amortisseurs ou des pads anti-vibrations peuvent être ajoutés à l'endroit où seront installés le contrôleur de vol et d'autres composants sensibles, afin de minimiser les interférences dues aux vibrations en vol. Une fois le châssis entièrement assemblé et solidement fixé, il est prêt à recevoir les autres composants tels que les moteurs, les ESC, le contrôleur de vol, et la batterie, qui complèteront la structure du drone.

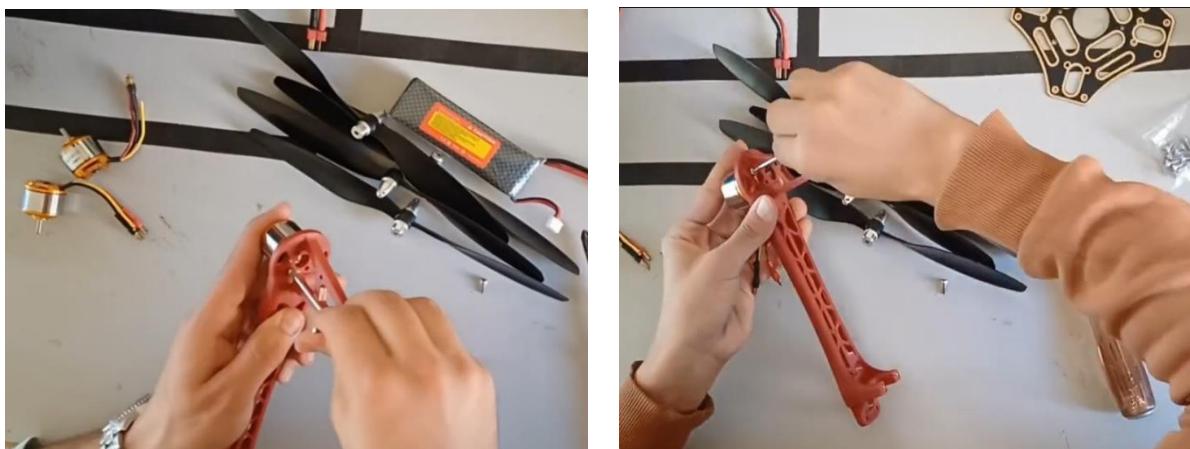


Figure 3.9 : L'assemblage du châssis

3.4 Soudage des composants

Le soudage des composants est une étape délicate et essentielle dans l'assemblage d'un drone, garantissant des connexions électriques fiables entre les différents éléments. Ce processus commence généralement par le soudage des câbles des moteurs aux contrôleurs de vitesse électroniques (ESC). Chaque moteur doit être connecté à son ESC correspondant, en

s'assurant que les connexions sont solides et exemptes de court-circuits. Ensuite, les fils d'alimentation des ESC sont soudés aux bornes de la distribution de puissance, souvent centralisée sur une plaque de distribution pour une meilleure organisation et gestion de l'énergie.

Le contrôleur de vol, tel qu'un Arduino Nano, doit être connecté au système de distribution d'énergie et aux ESC via des câbles signal, qui transmettent les commandes de vol. Si des capteurs additionnels comme le MPU6050 sont utilisés, ils doivent être soudés au contrôleur de vol en utilisant les broches appropriées pour la communication I2C. Lors de ces étapes, il est crucial de maintenir des soudures propres et précises pour assurer une conductivité optimale et éviter les pannes électriques.

Après le soudage des principaux composants, le récepteur radio FlySky est connecté au contrôleur de vol, souvent via des câbles servo qui doivent également être soudés ou insérés dans des connecteurs appropriés. Enfin, des vérifications minutieuses sont effectuées pour s'assurer que toutes les connexions sont sécurisées et qu'il n'y a pas de ponts de soudure indésirables. Une fois le soudage terminé, les composants sont soigneusement isolés et fixés pour prévenir tout mouvement ou déconnexion pendant le vol, garantissant ainsi une performance stable et fiable du drone.

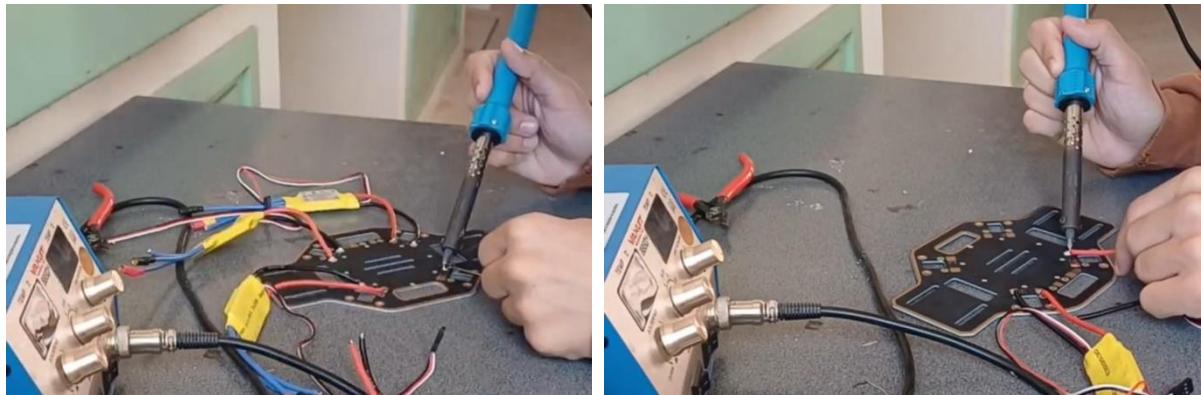


Figure 3.10 : Soudage des ESC

3.5 Schéma de câblage (en utilisant Ardupilot)

Le schéma de câblage d'un drone utilisant Ardupilot commence par l'installation du contrôleur de vol au centre du châssis, fixé avec des pads anti-vibrations pour minimiser les interférences dues aux vibrations. Les moteurs sont ensuite soudés aux ESC (contrôleurs de vitesse électroniques), qui sont connectés aux sorties motrices du contrôleur de vol. Les fils d'alimentation des ESC sont reliés à une plaque de distribution de puissance (PDB), qui est alimentée par une batterie LiPo. Le capteur MPU6050 est connecté au contrôleur de vol via les

broches I2C, en reliant les lignes SDA et SCL, ainsi que les fils d'alimentation VCC et GND. Le récepteur radio FlySky est branché aux entrées RC du contrôleur de vol en utilisant des câbles servo pour transmettre les commandes du pilote. Le module d'alimentation, placé entre la batterie et la PDB, surveille la consommation de courant et la tension, transmettant ces données au contrôleur de vol via un câble à 6 positions. Si un GPS et un module de télémétrie sont utilisés, ils sont connectés respectivement aux ports UART ou I2C du contrôleur de vol pour fournir des données de position et de vol en temps réel. Après avoir effectué toutes ces connexions, il est essentiel de vérifier chaque branche pour éviter les court-circuits, organiser les fils avec des serre-câbles et des gaines thermorétractables, et effectuer des tests initiaux pour assurer un câblage propre et fonctionnel, garantissant ainsi une performance optimale et une réponse précise du drone aux commandes.

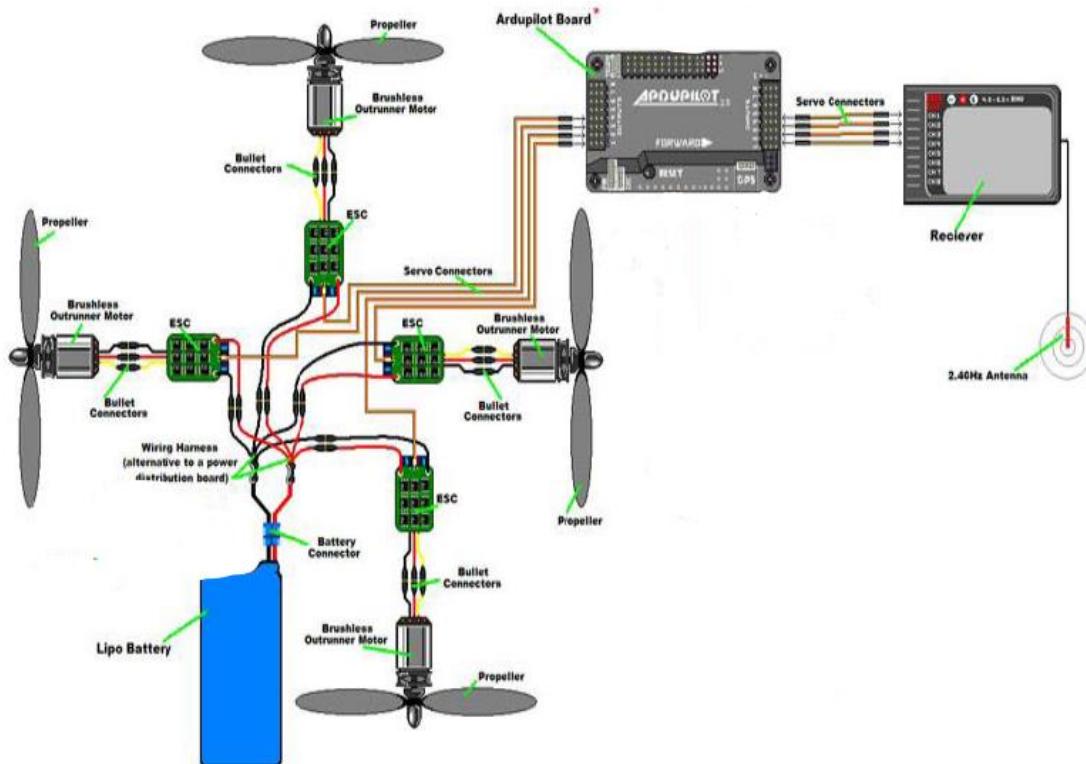


Figure 3.11 : Schéma de câblage (Ardupilot)

3.6 Configuration de Ardupilot

La configuration d'ArduPilot est une étape fondamentale pour préparer un drone à un vol réussi, impliquant plusieurs étapes clés pour paramétriser correctement le contrôleur de vol. Tout commence par la connexion de l'Arduino Nano au PC via un câble USB et le lancement du logiciel Mission Planner, l'outil de configuration d'ArduPilot. Une fois la connexion établie, il

est nécessaire de mettre à jour le firmware du contrôleur de vol avec la version la plus récente d'ArduPilot, en choisissant le bon type de véhicule, tel que quadcoptère, hélicoptère ou rover, en fonction de notre configuration. Après la mise à jour, l'étape suivante consiste à calibrer les composants du drone, en commençant par l'étalonnage des capteurs comme le MPU6050, où il faut suivre les instructions à l'écran pour assurer que les données d'accélération et de gyroscope sont précises. Ensuite, le calibrage des moteurs est réalisé en ajustant les paramètres des ESC pour garantir que chaque moteur réagit correctement aux signaux du contrôleur de vol. Le réglage des paramètres de vol est également crucial, impliquant la configuration des options telles que les limites de vitesse, les paramètres de stabilité, et les modes de vol (comme le mode stabilisé ou le mode GPS). Le processus inclut aussi la configuration des canaux du récepteur radio FlySky, en affectant les commandes aux canaux appropriés via les menus du logiciel Mission Planner. Une fois ces étapes complètes, des tests de vérification sont effectués pour s'assurer que toutes les fonctions sont opérationnelles et que le drone répond correctement aux commandes, incluant des tests de communication entre le récepteur et le contrôleur de vol, des simulations de vol, et des ajustements finaux basés sur les résultats des tests. En finalisant la configuration, il est crucial de documenter les paramètres et de sauvegarder la configuration pour des ajustements futurs ou pour reproduire la configuration sur un autre drone, garantissant ainsi une performance optimale et une réponse précise du drone aux commandes.

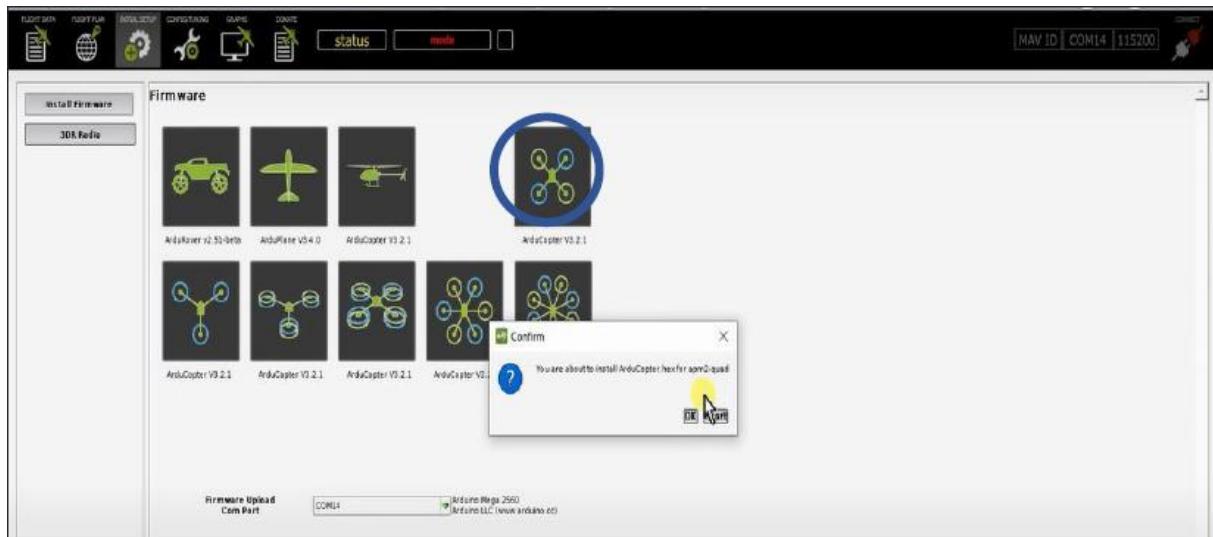


Figure 3.12 : Configuration d'Ardupilot

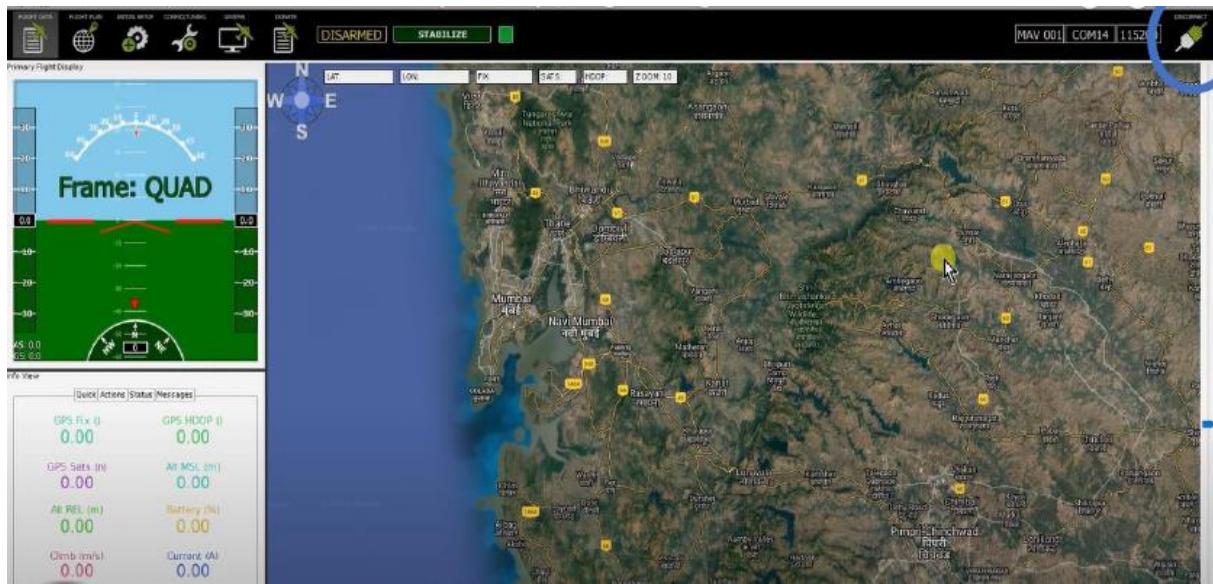


Figure 3.13 : Interface de connexion pour Ardupilot

Le contrôle d'un moteur en utilisant ArduPilot est une étape essentielle pour vérifier et ajuster le fonctionnement des moteurs d'un drone, afin d'assurer une performance optimale lors des vols. Cette procédure débute par l'ouverture du logiciel Mission Planner et la connexion de l'Arduino Nano au PC via un câble USB. Une fois connecté, il est crucial de s'assurer que le firmware ArduPilot est à jour et que le contrôleur de vol est correctement calibré. Le contrôle des moteurs commence par la vérification des connexions entre les ESC (contrôleurs de vitesse électroniques) et les moteurs, ainsi qu'entre les ESC et le contrôleur de vol, en s'assurant que toutes les connexions sont sécurisées et exemptes de court-circuits. Ensuite, dans Mission Planner, accédez à l'onglet "**Initial Setup**" puis "**Mandatory Hardware**" et sélectionnez "**Motor Test**", qui vous permet de tester chaque moteur individuellement en ajustant les paramètres de vitesse et de direction à l'aide de curseurs et en observant la réponse des moteurs en temps réel. Ce test permet de vérifier que les moteurs fonctionnent correctement et réagissent bien aux commandes, en détectant d'éventuels problèmes comme des bruits anormaux ou des comportements inattendus. Si des anomalies apparaissent, il peut être nécessaire de revoir les connexions des ESC, les configurations des paramètres de vol ou les calibrations des capteurs. Une fois les tests effectués avec succès, il est important de documenter les résultats et de sauvegarder les configurations ajustées pour garantir une performance optimale lors des futurs vols du drone.

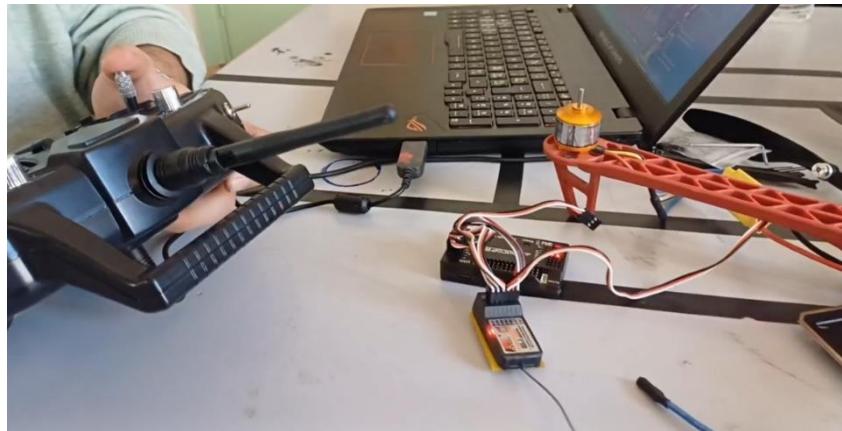


Figure 3.14 : Control d'un moteur par le transmetteur

3.7 Réalisation de la carte de commande

La réalisation de la carte de commande pour notre drone a impliqué l'utilisation d'une carte PCB vierge sur laquelle nous avons effectué les différentes connexions nécessaires entre les composants électroniques, notamment l'Arduino Nano et le MPU6050. Ce processus commence par la préparation de la carte PCB vierge, où nous avons identifié les points de connexion appropriés pour chaque composant. Ensuite, nous avons soigneusement soudé les broches de l'Arduino Nano et du MPU6050 aux emplacements désignés sur la carte, en suivant un schéma de câblage précis pour assurer des connexions correctes et sécurisées. Nous avons également intégré les connecteurs pour les ESC (contrôleurs de vitesse électroniques) et autres composants nécessaires au fonctionnement du drone. Une fois les composants fixés, nous avons procédé à des tests fonctionnels pour vérifier l'intégrité des connexions et le bon fonctionnement du système. Ces tests incluent la vérification des alimentations, des signaux de communication entre l'Arduino Nano et le MPU6050, ainsi que des réponses des ESC et des moteurs. Cette étape est essentielle pour les futurs ajustements et la maintenance du système, garantissant ainsi que la carte de commande est prête à intégrer et contrôler les différents systèmes du drone, en préparation des phases de test en vol et de mise en service.

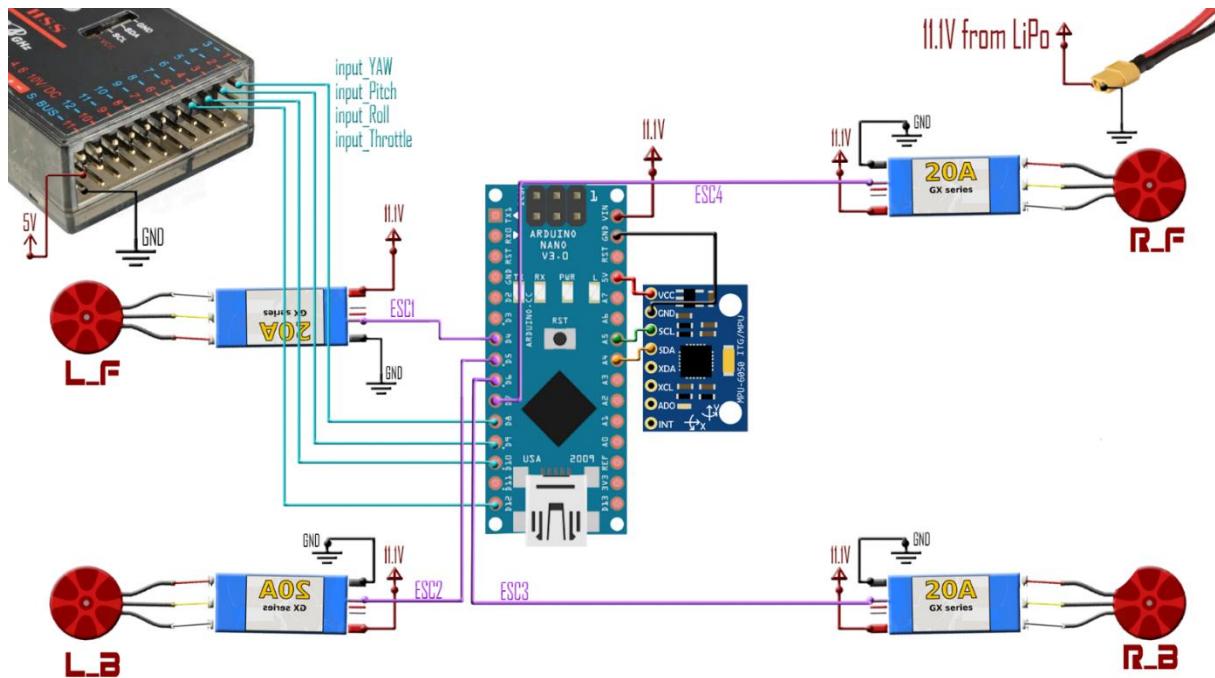


Figure 3.15 : Schéma de câblage (Arduino nano)

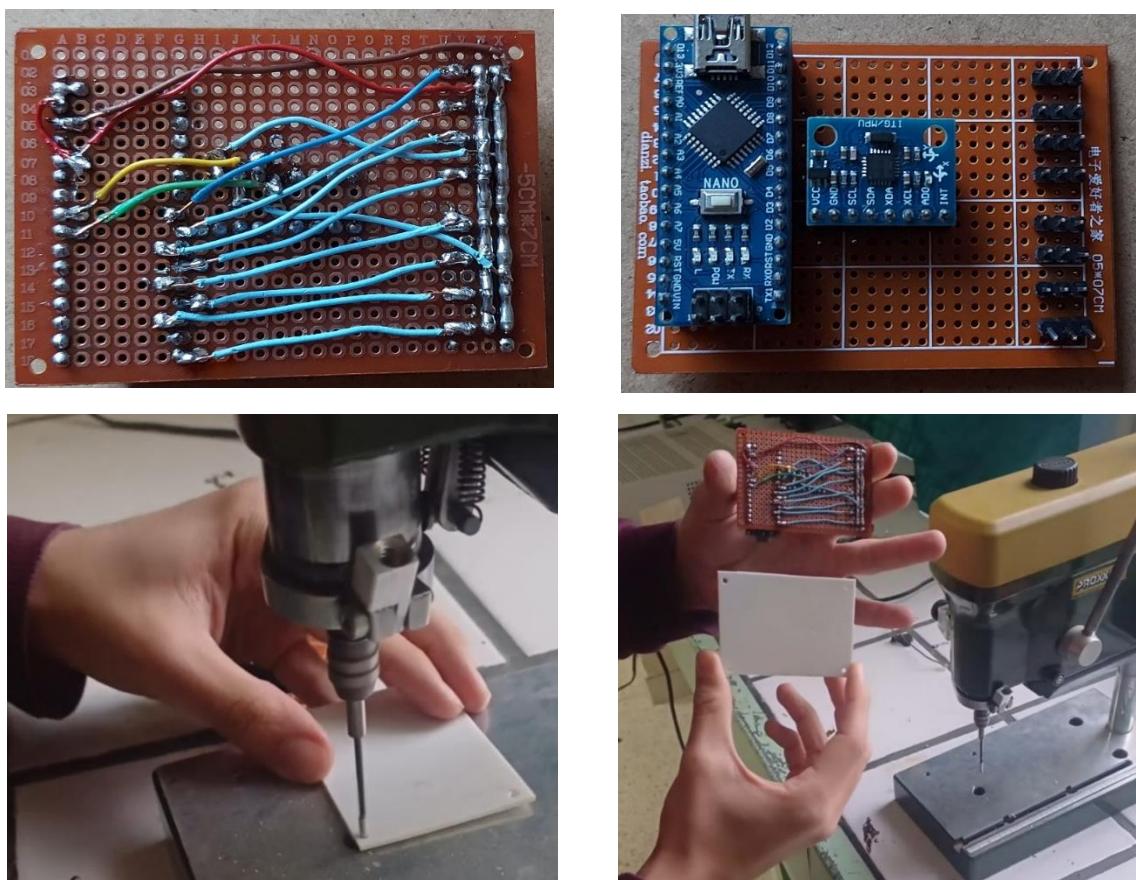


Figure 3.16 : la réalisation de la carte de commande

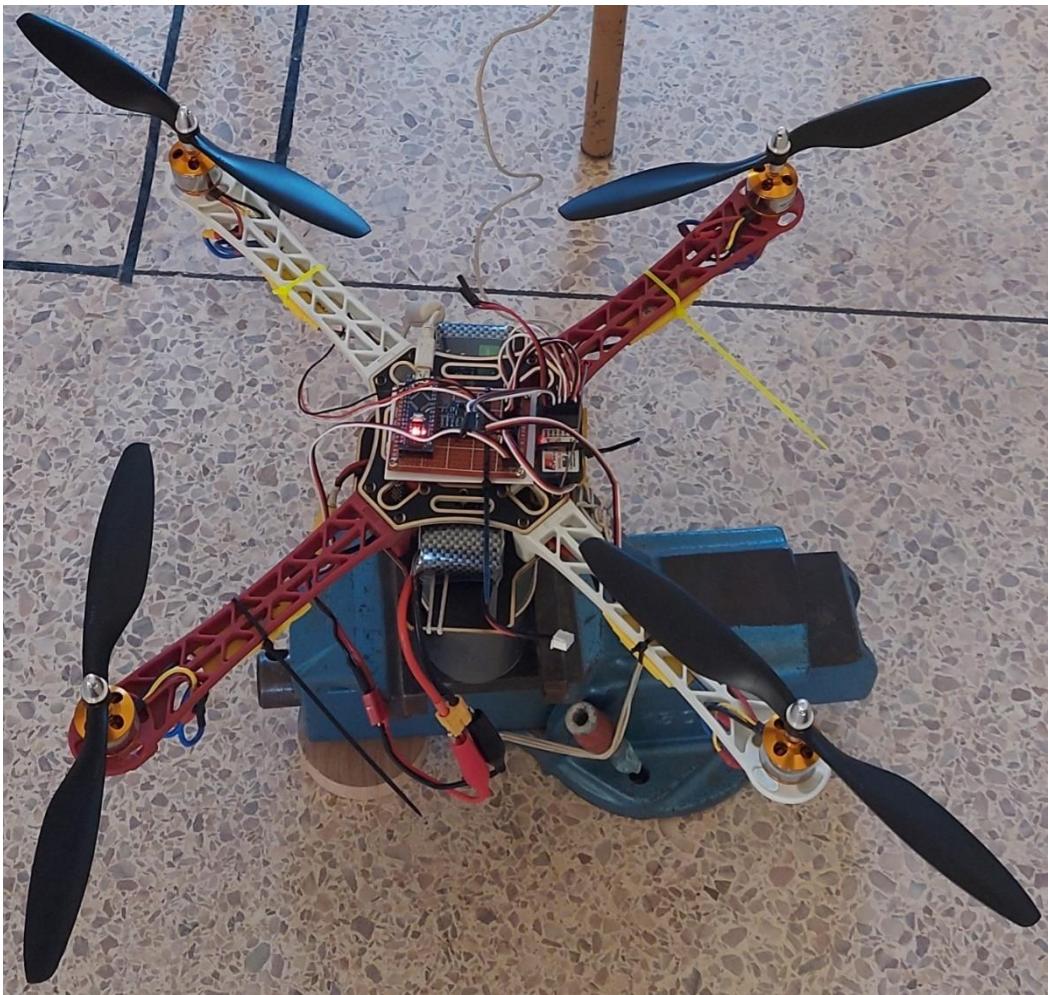


Figure 3.17 : Le drone après l'assemblage

Après l'assemblage du drone et l'implémentation du programme sur la carte de commande, nous avons procédé à une série de tests rigoureux pour s'assurer de son bon fonctionnement. Pour garantir la sécurité, ces tests ont été réalisés avec des supports de sécurité, minimisant ainsi les risques de dommages matériels ou de blessures. Ces tests comprenaient la simulation de variations de différents paramètres tels que la puissance des moteurs, la stabilité en vol, et la réactivité des commandes. Chaque paramètre a été ajusté et évalué afin de perfeuiner les performances du drone et d'identifier d'éventuels problèmes. Grâce à cette approche méthodique et sécurisée, nous avons pu optimiser les configurations du drone, assurant ainsi un fonctionnement stable et fiable avant les essais en conditions réelles.



Figure 3.18 : Essai du drone en conditions réelles.

Malgré les étapes d'assemblage complètes et l'implémentation réussie du programme, nous avons constaté que les paramètres du drone n'étaient pas encore optimisés lors des premiers vols d'essai. Bien que le drone ait pu voler, des ajustements étaient nécessaires pour améliorer sa stabilité, sa réactivité et sa performance globale. Nous avons identifié plusieurs domaines nécessitant des améliorations, tels que la calibration du capteur, l'ajustement des PID pour un contrôle plus précis des moteurs, et l'optimisation des algorithmes de vol. Ces observations nous ont permis de cibler les ajustements spécifiques nécessaires pour peaufiner le drone et garantir des performances optimales lors des vols futurs.

Conclusion

En conclusion, la réalisation et les premiers essais de notre drone ont démontré un potentiel prometteur malgré quelques ajustements nécessaires. Grâce à une approche méthodique et sécurisée, nous avons pu identifier et cibler les paramètres à optimiser. Les améliorations apportées permettront de garantir des performances optimales et une meilleure

fiabilité pour les vols futurs, renforçant ainsi notre expertise et notre savoir-faire dans la conception et l'assemblage de drones

Conclusion générale

À travers ce rapport, nous avons détaillé les étapes cruciales et les spécifications techniques de notre projet de modélisation et de réalisation d'une carte de commande pour un drone, mettant en lumière notre approche structurée et méthodique pour atteindre nos objectifs.

En résumé, nous avons présenté les objectifs, les spécifications techniques, et les étapes clés de notre projet. Les exigences de performance et de sécurité ont été définies, ainsi que les outils et méthodes que nous utiliserons. Ce cahier des charges nous fournit une feuille de route claire pour mener à bien ce projet, assurant une approche structurée et méthodique pour atteindre nos objectifs.

Le quadrirotor, en raison de son élégance et de sa dynamique, est l'un des drones les plus populaires et séduisants. Nous avons décrit les mouvements de base du quadrirotor et présenté son modèle dynamique en utilisant le formalisme de Newton-Euler. Ce modèle a été simplifié par des hypothèses facilitant la synthèse des lois de commande. Le schéma de commande PID appliqué au quadrirotor a été introduit, assurant la stabilisation en vol stationnaire. Nos objectifs pour la conception et la construction de notre drone étaient ambitieux et bien définis.

La réalisation et les premiers essais de notre drone ont démontré un potentiel prometteur, bien que quelques ajustements soient nécessaires. Une approche méthodique et sécurisée nous a permis d'identifier et de cibler les paramètres à optimiser. Les améliorations apportées garantiront des performances optimales et une meilleure fiabilité pour les vols futurs, renforçant notre expertise et notre savoir-faire dans la conception et l'assemblage de drones.

En conclusion, notre projet avance conformément aux prévisions, et chaque étape a contribué à renforcer notre compréhension et notre capacité à atteindre les objectifs fixés. Les bases solides posées permettront le succès de notre projet, et les ajustements futurs seront déterminants pour l'optimisation et la fiabilité du drone.

Annexe I

```
Wire.begin(); //begin the wire communication
Wire.beginTransmission(0x68); //begin, Send the slave adress (in this case 68)
Wire.write(0x6B); //make the reset (place a 0 into the 6B register)
Wire.write(0x00);
Wire.endTransmission(true); //end the transmission

Wire.beginTransmission(0x68); //begin, Send the slave adress (in this case 68)
Wire.write(0x1B); //We want to write to the GYRO_CONFIG register (1B hex)
Wire.write(0x10); //Set the register bits as 00010000 (1000dps full scale)
Wire.endTransmission(true); //End the transmission with the gyro

Wire.beginTransmission(0x68); //begin, Send the slave adress (in this case 68)
Wire.write(0x43); //First adress of the Gyro data
Wire.endTransmission(false);
Wire.requestFrom(0x68, 4, true); //We ask for just 4 registers

Gyr_rawX=Wire.read()<<8|Wire.read(); //Once again we shif and sum
Gyr_rawY=Wire.read()<<8|Wire.read();

Gyr_rawX=Wire.read()<<8|Wire.read(); //Once again we shif and sum
/*Now in order to obtain the gyro data in degrees/seconds we have to divide first
the raw value by 32.8 because that's the value that the datasheet gives us for a
1000dps range*/
/*---X---*/
Gyr_rawX = (Gyr_rawX/32.8) - Gyro_raw_error_x;
/*---Y---*/
Gyr_rawY = (Gyr_rawY/32.8) - Gyro_raw_error_y;

/*Now we integrate the raw value in degrees per seconds in order to obtain the angle
 * If you multiply degrees/seconds by seconds you obtain degrees */
/*---X---*/
Gyro_angle_x = Gyro_angle_x + Gyr_rawX*elapsedTime;
/*---X---*/
Gyro_angle_y = Gyro_angle_y + Gyr_rawY*elapsedTime;

Wire.beginTransmission(0x68); //begin, Send the slave adress (in this case 68)
Wire.write(0x3B); //Ask for the 0x3B register- correspond to AcX
Wire.endTransmission(false); //keep the transmission and next
Wire.requestFrom(0x68, 6, true); //We ask for next 6 registers starting withj the 3B
//If we read the datasheet, for a range oft-8g, we have to divide the raw values by
4096
Acc_rawX=(Wire.read()<<8|Wire.read())/4096.0; //each value needs two registres
Acc_rawY=(Wire.read()<<8|Wire.read())/4096.0;
Acc_rawZ=(Wire.read()<<8|Wire.read())/4096.0;

/*Now in order to obtain the Acc angles we use euler formula with acceleration values
after that we substract the error value found before*/
/*---X---*/
Acc_angle_x = (atan((Acc_rawY)/sqrt(pow((Acc_rawX),2) +
pow((Acc_rawZ),2)))*rad_to_deg) - Acc_angle_error_x;
/*---Y---*/
Acc_angle_y = (atan(-1*(Acc_rawX)/sqrt(pow((Acc_rawY),2) +
pow((Acc_rawZ),2)))*rad_to_deg) - Acc_angle_error_y;
///////////////////////////////Total angle and filter///////////////////////////////
/*---X axis angle---*/
Total_angle_x = 0.98 *(Total_angle_x + Gyro_angle_x) + 0.02*Acc_angle_x;
/*---Y axis angle---*/
Total_angle_y = 0.98 *(Total_angle_y + Gyro_angle_y) + 0.02*Acc_angle_y;
```