

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'enseignement supérieur et de la recherche scientifique
Université Mouloud Mammeri de Tizi-Ouzou
Faculté du Génie Electrique et d'Informatique.
Département d'Electronique.



Mémoire

En vue de l'obtention du Diplôme Master en
Electronique

Thème :

*Conception et réalisation d'un DRONE
Quadri-rotor*



Proposé et dirigé par :

Mr.: M. LAGHROUCHE

Réalisé par :

Mr.: SADDAOUI Rafik

Promotion « 2012/2013 »



Remerciements

Je commence par remercier Dieu le tout puissant de m'avoir donné le courage, la volonté, l'amour du savoir et surtout la patience pour pouvoir de réaliser ce modeste travail.

Je tiens à exprimer mes remerciements et mes profondes considérations à :

❖ *Mon promoteur : Mr LAGHROUCHE pour le temps précieux qu'il a consacré à m'orienter, me conseiller et pour sa sollicitude particulière à mon égard.*

❖ *Les membres du jury qui voudront bien évaluer mon travail.*

❖ *A toute personne qui a contribué de près ou de loin à la réalisation de ce modeste travail et à tous ceux qui m'ont apporté leurs aide.*

Merci



Dédicaces :

Je dédie ce modeste travail à :

Mes très chers parents.

Mes frères et mes sœurs : Rachida (Son marie Michand

Et son fils Hend), Lamia, Yassina,

Taefarinas et Massinissa.

Et Tous mes ami (e)s .

Liste des figures

Chapitre I :

<i>Figure I.1 : Les mouvements de quadri-rotor par rapport au référentiel mobile</i>	
<i>(x', y', z')</i>	3
<i>Figure I.2: La disposition et sens de rotation des différents moteurs</i>	4
<i>Figure I.3 : Demonstration de mouvement de glissement vertical.</i>	5
<i>Figure I.4 : Demonstration de mouvement de roulis</i>	5
<i>Figure I.5 : Demonstration de mouvement de tangage</i>	6
<i>Figure I.6 : Demonstration de mouvement de lacet.</i>	6
<i>Figure I.7: Références fixe et mobile auquel est liée la quadri rotor.</i>	7
<i>Figure I.8: Commande global de procédé de stabilisation.</i>	10
<i>Figure I.9: Schéma du modèle sous SimuLink</i>	11
<i>Figure I.10: Model dynamique de moteur.</i>	11
<i>Figure I.11: Solution mixeurs-gyroscopes</i>	12
<i>Figure I.12: Solution carte numérique embarqué.</i>	13
<i>Figure I.13: Les fonctions de différents blocs servants au contrôle automatique de vol</i> ...14	
<i>Figure I.14: Boucles de control PID pour le mode stabilisé.</i>	15
Chapitre II:	
<i>Figure II.1: Atmega328p (version 28 broches)</i>	18
<i>Figure II.2: Atmega8A (version 32 broches)</i>	18
<i>Figure II.3 : deux types d'interfaces spi.</i>	19

<i>Figure II.4: Connections spi d'un ATmega328P.</i>	20
<i>Figure II.5: Clone d'un programmeur d'AVR.</i>	20
<i>Figure II.6: Correspondance entre les broches de l'ATmega328 et celles de l'Arduino.</i> ...	23
<i>Figure II.7: Sélection de carte cible l'ATmega328P.</i>	23
<i>Figure II.8: Bouton permettant la compilation de code source.</i>	24
<i>Figure II.9: Choix de programmeur USBASP.</i>	24
<i>Figure II.10: Carte arduino duemilanove.</i>	25
<i>Figure II.11: Programmation d'une carte arduino UNO via UART intégrée.</i>	26
<i>Figure II.12: Carte Arduino UNO.</i>	26
<i>Figure II.13 : Carte arduino mega.</i>	27
<i>Figure II.14: Fenêtre principale d'édition des programmes Arduino.</i>	28
<i>Figure II.15: Le programme transféré avec succès se lance.</i>	28
<i>Figure II.16: Image de programmeur USBASP</i>	32
<i>Figure II.17: Un contrôleur de vitesse raccordé au programmeur via l'interface ISP...</i>	33
<i>Figure II.18: Fenêtre principale de type MSDOS de saisi et d'exécution des opérations par avrdude.</i>	34
<i>Figure II.18: Fenêtre signalant le non-établissement de la communication avec le matériel.</i>	35
<i>Figure II.19: Fenêtre d'affichage des opérations effectuées par avrdude.</i>	35
Chapitre III :	
<i>Figure III.1: Le dispositif gyroscopique mécanique</i>	36
<i>Figure II.2: Image de capteur gyroscopique ITG-3200.</i>	37

Figure III.3: Image montrant le module BMA180.	38
Figure III.4: Schéma de brochage de BMA180.	38
Figure IV.1: Moteur sans balais avec le rotor externe et stator au milieu.	44
Figure IV.2: Vue intérieur d'un moteur électrique sans balais.	48
Figure IV.3: le moteur en vue éclatée.	48
Figure IV.4: Représentation des enroulements en étoile et triangle.	49
Figure IV.5: Vue intérieur de contrôleur de vitesse commercial.	51
Figure IV.6: Image d'un contrôleur de vitesse commercial.	52
Figure IV.7: Canalisation des courants par les différents mosfets.	53
Figure IV.8: Etage de puissance de contrôleur de vitesse constituée de 6 Mosfets.	53
Figure IV.9: Exemple de contrôleur de vitesse commerciale a deux rangées de mosfets PMOS et NMOS (face inferieur de la carte électronique).	54
Figure IV.10: Disposition des capteurs a « effet Hall » de position du rotor.	54
Figure IV.11: Signaux de commande en entrées et sorties de contrôleur de vitesse et de moteur.	55
Figure IV.12: Equilibre 1a gauche et commutaion 1 a droite.	57
Figure IV.13: Equilibre 2 a gauche et commutaion 2 a droite.	57
Figure IV.14: Circuit de détection de position de rotor par comparaison de niveaux d'amplitude.	58
Figure IV.15: Aucun courant n'est débité et les bornes en l'air.	59
Figure IV.16: Après délais de commutation.	59
Figure IV.17: Détection de zéro sur A1.	60
Figure IV.18: Délais de commutation.	60

Figure IV.19: <i>Schema électrique generale de contrôle de moteur électrique.</i>	62
Figure IV.20: <i>Les paramètres électriques servants aux calcule de la puissance délivrée à chaque hélice.</i>	67
Figure V.1: <i>Schéma synoptique d'un émetteur.</i>	74
Figure V.2: <i>structure générale de l'émetteur.</i>	75
Figure V.3: <i>Différence de fonctionnement des amplificateurs de puissance classe A, B, AB et C. $I_{collecteur} = f(t)$.</i>	76
Figure V.4: <i>Schéma synoptique du récepteur à un changement de fréquence.</i>	78
Figure V.5: <i>Récepteur a double changement de fréquences.</i>	80
Figure V.6: <i>codage PPM/PCM</i>	83
Figure V.7: <i>structure d'un récepteur de télécommande 8voix.</i>	84
Figure V.8: <i>structure générale de la radiocommande (manuelle) pour le contrôle de drone.</i>	85
Figure V.9: <i>Modules Xbee en communications.</i>	86
Figure V.10: <i>illustration de la communication via Xbee.</i>	86
Figure V.11: <i>communications a base de modules xbee a travers l'interface labview.</i>	87
Figure V.12: <i>configuration a l'aide de X-CTU.</i>	88
Figure V.13: <i>Fenêtre Test X-CT</i>	89
Figure V.14: <i>Configuration du module xbee</i>	89
Figure V.15: <i>Comment couper l'alimentation pour retirer le module xbee.</i>	90
Figure V.16: <i>l'adresse haute de module xbee émetteur.</i>	91
Figure V.17: <i>L'adresse de module de destination.</i>	91

Figure V.18: mannette de jeu wiimote de Nintendo.	92
Figure V.19: diagramme (block-diagram) de l'interface virtuelle de contrôle via la wiimote.....	95
Figure VI.1: le rotor avec les aimants fixés a sa surface intérieure.....	97
Figure VI.2: plaquettes d'aimant néodyme.	97
Figure VI.3: stator avec enroulement en triangle.	98
Figure VI.4: image d'un tour universel pour l'usinage des pieces rondes.	98
Figure VI.5: pièce en aluminium brute qui deviendra le rotor, fixé sur le mandrin d'un tour.....	99
Figure VI.6: usinage de la surface extérieur de support-stator.	99
Figure VI.7: Alésage de l'emplacement de roulements	99
Figure VI.8: moteur sans balais finalisé.	100
Figure VI.9: Sous programme qui effectue les fonctions de stabilisation de drone.	104
Figure VI.10: la routine d'interruption gérant les communications.	105
Figure VI.11: signaux de commande des moteurs issus de contrôleur principale.	108
Figure VI.12: Fonction de stabilisation en boucle fermée.....	116
Figure VI.13: Représentation électrique d'un transistor mosfet de puissance avec sa diode de protection.	119
Figure VI.14: Schéma montrant plusieurs buffers en parallèles	120
Figure VI.15: le rapport entre le rapport cyclique de signal PWM et la tension moyenne de sortie.....	122
Figure VI.16: schéma électrique de la carte constituant la centrale inertielle (contrôleur de vol).	124

<i>Figure VI.17: schéma électrique de cœur de contrôleur de vitesse.....</i>	<i>125</i>
<i>Figure VI.18: schéma électrique de la partie de puissance de contrôleur de vitesse.</i>	<i>126</i>
<i>Figure VI.19: schema d'implantation des composants de controleur de vol sur ARES.</i>	<i>127</i>
<i>Figure VI.20: typons et emplacement des composants sur la carte de contrôleur de vitesse.....</i>	<i>128</i>
<i>Figure VI.21: Photographie de châssis en croix.....</i>	<i>129</i>
<i>Figure VI.22: image de l'emplacement moteur à l'extrémité des bras.</i>	<i>130</i>
<i>Figure VI.23: image globale de drone.</i>	<i>131</i>
<i>Figure VI.23: deux hélices avec sens des pales opposées.....</i>	<i>132</i>

Sommaire

Introduction générale	1
Chapitre I : Principe de vol d'un drone quadri-rotor	
I.1.Introduction	3
I.2.Les mouvements de quadri rotor	4
I.3.Le quadri-rotor a cinq mouvements principaux	4
I.3.1.Mouvement vertical	5
I.3.2.Mouvement de roulis	5
I.3.3.Mouvement de tangage	6
I.3.4.Mouvement de lacet	6
I.4.Le modèle dynamique du quadri-rotor	7
I.4.1.La portance	7
I.4.2.La traînée	7
I.4.3. L'effet gyroscopique	8
I.4.4.Fonction de transfert de l'ensemble moteur-variateur	9
I.5.Modèle SimuLink du quadri-rotor	10
I.6.Modèle dynamique du moteur	11
I.7.La stabilisation simplifiée a l'aide de mixeur.....	12
I.8.Solution à base d'une carte a microcontrôleur embarqué	13
I.9.La carte mère embarquée de contrôle de vol (Flight Control)	13
I.10.Le schéma de block qui montre les différentes fonctions d'un contrôleur de vol ..	14
I.11.Contrôle de stabilisation	15

I.12.Conclusion	16
-----------------------	----

Chapitre II : Processeurs de traitement et de control.

II.1. Introduction	17
II.2. Présentation et raison de choix des AVR	17
II.3. ATmega328P et atmega8	17
II.4.Programmation des microcontrôleurs AVR.....	18
II.5.Interface SPI	19
II.6.Le programmeur d'AVR	20
II.7.Qu'est-ce qu'Arduino	21
II.7.1. Objectif de l'Arduino	21
II.8.Configurer l'IDE Arduino	22
II.9.Programmer un ATmega328P nu avec Arduino	22
II.10.Télécharger le programme vers la cible	23
II.10.1.Télécharger « uploader » un programme avec l'IDE Arduino	24
II.11. Matériel Arduino	25
II.11.1.La Duemilanove	25
II.11.2.Arduino UNO	25
II.11.3.La carte Mega	26
II.12.Description de l'arduino IDE	27
II.13.Transfert des programmes vers la carte Arduino	28
II.14. Structure d'un programme Arduino	29

II.14.1. Les commentaires	29
II.14.2.Variables	29
II.14.3.Fonctions	30
II.14.3.1. Quelques fonctions très utilisées	31
II.14.3.1.1.La fonction pinMode()	31
II.14.3.1.2.La fonction digitalWrite()	31
II.14.3.1.3.La fonction delay()	32
II.15.Matériels et logiciels nécessaires et préparation à la mise à jour de contrôleur de vitesse à base d'atmega8A	32
II.14.1.Commençant par préparer le contrôleur de vitesse	32
II.15. Chargement du programme.....	33
II.15.1. Avrdude.....	33
II.15.1.1. La programmation démarre.....	34
II.16. Conclusion	35

Chapitre III : Capteurs de navigation aérienne

III.1. Introduction	36
III.2. Le gyroscope	36
III.3.Accéléromètre triaxial BMA180	37
III.3.1. Applications	37
III.3.2. Brochage de BMA180	38
III.3.3.Description des broches	38
III.3.4. Les caractéristiques de BMA 180	39

III.4. Baromètre	39
III.5. Boussole électronique (Compass)	40
III.6. Le GPS (Global Positioning System)	40
III.7. Télémètre	40
III.8. Camera	40
III.9. Conclusion	41

Chapitre IV : Système de propulsion à base de moteurs électriques sans balais.

IV.1. Introduction	42
IV.2. Fonctionnement d'un moteur électrique	42
IV.3. Les moteurs sans balais MSB	43
IV.3.1. Les moteurs DC-sans balais	43
IV.4. Comment fonctionne un moteur sans balais « brushless »	44
IV.5. Inconvénients	45
IV.6. Avantages de ces moteurs	45
IV.7. Choix d'un moteur sans balais	46
IV.7.1. Les kv	46
IV.7.2. Le couple	46
IV.8. Moteur sans balais à rotor externe « outrunner »	47
IV.8.1. Comment se présente le moteur ?	48
IV.9. Le contrôleur de vitesse (électronique speed contrôler ou ESC)	49
IV.9.1. Fonctionnement	49

IV.9.2. Les réglages nécessaires	50
IV.9.4. Contrôleur de commerce	50
IV.9.5. Choix de contrôleur de vitesse.....	52
IV.9.6. Les fonctions du contrôleur	52
<i>A-L'entree des signaux de commandes</i>	53
<i>b-La commutation des tensions sur les bornes A1, B1, C1 du moteur</i>	53
<i>c-Commande des interrupteurs</i>	53
<i>d- Le contrôleur mesure la tension induite dans la bobine laissée en circuit ouvert</i>	
IV.10. Logique de commande des interrupteurs	54
IV.11. Régulation de vitesse d'un moteur sans balais	56
IV.12. Modes de fonctionnement	56
IV.12.1. Le mode pas à pas	56
IV.12.1.1. Grand inconvénient du mode "pas à pas"	57
IV.12.2. Second mode « <i>Détection de position du rotor suivie de commutation</i> »	58
IV.12.3. Quelques valeurs	61
IV.13. Calcule de puissance fournit au moteur	62
IV.13.1. Calcul de vitesse de rotation.....	62
IV.13.2. Limitation du moteur	63
IV.13.3. Calcule des différents paramètres	63
IV.13.3.1. Le rendement maximum	64
IV.13.4. Choisir la motorisation électrique	64
IV.13.4.1. Choisir une puissance d'alimentation	64

IV.13.4.2. Evaluer la masse totale	65
IV.13.4.3. Choisir le moteur qui correspond à ce niveau de puissance.....	65
IV.13.4.4. Choisir un contrôleur et une batterie	65
IV.13.4.5. Refaire l'évaluation de la masse	67
IV.13.4.6. Choisir l'hélice	67
A)-Applications numériques	69
IV.13.5. Refaire l'évaluation de masse	72
IV.13.5.1. Choisir l'hélice	72
IV.14. Conclusion	73

Chapitre V : Communications et interfaces de commande à distance

V.1.Introduction	74
V.2.Composants essentielle d'une chaine de communication et contrôle à distance.....	74
V.2.1.L'Émetteur	74
V.2.2.1.Génération de la fréquence pilote	74
V.2.2.2.Amplificateur de sortie	76
a)-Classes de fonctionnement de l'amplificateur de sortie	76
V.2.2.3.Adaptation d'impédance	77
V.2.3.Le Récepteur	77
V.2.3.1.Rôle du récepteur	77
V.2.3.2.Récepteurs à un changement de fréquence	78
V.2.3.3.Récepteurs à double changement de fréquence	78

V.2.3.3.1.Filtres passe-bande d'entrée	80
V.2.3.3.2.Amplificateurs d'entrée	81
V.2.3.3.3.Mélangeurs.....	81
V.2.3.3.4.Oscillateur local	81
V.2.3.3.5.Filtres à la fréquence intermédiaire	81
V.2.3.3.6.Amplificateurs à la fréquence intermédiaire.....	82
V.2.3.3.7.Signaux perturbateurs externes	82
V.2.3.3.8.Démodulateur.....	82
V.3.La radiocommande.....	82
V.4.Communications a base de modules xbee	85
V.4.1.Présentation module xbee	85
V.4.2.Structure matérielle	86
V.4.3.Trames API	87
V.4.4.Configuration de module xbee.....	88
V.5.Commande de drone via l'interface labview	92
V.5.1.Arduino et LIFA.....	92
V.5.2.Matériels et logiciels nécessaire	93
a)- <i>Matériels</i>	93
b)- <i>Logiciels</i>	93
V.5.3.Installation du firmware LIFA sur la carte Arduino	93
V.5.4.Configuration de la communication Bluetooth entre le PC et la wiimote.....	94
V.5.5.Calibration des données de l'accéléromètre	95

V.5.5.1.Le code labview de graphe « Accelerometer.vi »	95
V.6.Conclusion	96

Chapitre IV : *Partie pratique*

VI.1.Introduction :	97
VI.2.Fabrication des moteurs sans balais.....	97
VI.2.1Quelques opérations réalisées.....	98
VI.2.2.Le moteur fini	100
VI.3.Choix du microcontrôleur pour la centrale inertielle (contrôleur de vol)	100
VI.3.1.Branchement de la liaison carte / ordinateur	101
VI.4.Algorithme général	102
VI.4.1.Le programme de stabilisation	102
VI.4.2.Explication de code source	102
VI.4.3.Organigramme de gestion de stabilisation.....	104
VI.4.4.Programme d'interruption	105
VI.4.5.Routine d'initialisation	105
VI.4.6.Signal de commande des contrôleurs de vitesse PWM	108
VI.4.7. Acquisition des données des capteurs	110
VI.5.Les modes de pilotage	116
VI.6.Réalisation des différentes la cartes électroniques	117
VI.6.1.Carte du contrôleur à base d' Atmega328	117
VI.6.2.Commande des moteurs par MOSFETs	118

VI.6.3.Les tensions PWM de commande des moteurs	121
VI.6.4.Carte des capteurs	122
VI.6.5.1.Catre de puissance	123
VI.6.6.1.Schéma de la carte de control de vole	124
VI.6.6.2. schéma électrique de la carte de control de vitesse	125
VI.6.7.Le module de puissance	125
VI.6.8.Routage	127
VI.6.9.Le châssis	128
VI.6.10.Conception de la structure	128
VI.6.10.1.Avantages de cette conception	129
VI.6.11.Les hélices	131
VI.6.12.Autonomie	132
VI.6.13.Aspects de sécurité dans les drones	132
VI.6.14.Conclusion	133
Conclusion générale	134

Références

Annexes

Introduction générale :

Depuis une dizaine d'années, les avancées technologiques ont permis la conception et la construction de mini-avions ou hélicoptères avec des capacités toujours plus développées pour réaliser des vols en autonome. Ces appareils sont connus sous le nom de drones. Parmi les applications civiles nous pouvons citer : la surveillance du trafic, des frontières, des barrages, des lignes de haute tension et des grands ouvrages d'art, le sauvetage des personnes en difficulté, la détection de feux de forêts, la retransmission de signaux de communication, etc. La recherche dans le domaine des véhicules aériens autonomes est essentiellement pluridisciplinaire. En effet elle fait intervenir des domaines très variés tels que l'aérodynamique, le traitement du signal et de l'image, la commande automatique, la mécanique, les matériaux et composites, les communications, l'informatique temps réel, l'architecture de microcontrôleurs, l'électronique, la sûreté de fonctionnement, les nanotechnologies, la gestion de l'énergie, la motorisation et même la biologie. Dans ce papier nous nous intéressons en particulier aux véhicules aériens miniatures ou mini-drones capables de réaliser du vol stationnaire, sachant que cette caractéristique est très utile pour des missions de surveillance aérienne. Nous nous intéresserons en particulier au quadrirotor (un mini-drone à quatre hélices) réputé par sa stabilité, et sa manœuvrabilité.

Le travail que nous avons réalisé est structuré en 6 chapitres :

Chapitre I : S'intitule «Etude dynamique de drone » comprend la formulation des équations régissant les différents phénomènes, la simulation et l'établissement d'un schéma préliminaire de contrôleur de vol.

Chapitre II : «Processeurs de traitement et de control » qui comprend le choix et la description des microcontrôleurs utilisés ainsi que les outils de programmation nécessaires.

Chapitre III : « Les capteurs de navigation aérienne» chapitre pour analyser les caractéristiques des capteurs à employer.

Chapitre IV : « Système de propulsion à base de moteurs électriques sans balais » est consacré à la présentation des moteurs utilisés pour le décollage de drone et les calculs nécessaires.

Chapitre V : « Communications et interfaces de commande à distance » comprend une étude de des systèmes de commande et de contrôle de drone en comparant entre des méthodes de contrôles manuels avec d'autres a travers des outils matériels et logiciels de commande évoluée.

Chapitre VI : « Partie pratique » est consacrée à la réalisation des différents composants et modules qui rentrent dans la fabrication et mise en œuvre.

Chapitre I:

Etude dynamique du drone.

I.1.Introduction :

Un quadri-rotor possède, comme son nom l'indique, quatre rotors pour se sustenter. Pour contrer un moment de lacet, il est nécessaire de faire tourner deux hélices dans un sens et les deux autres dans l'autre sens. En effet, lorsque l'on projette les forces aérodynamiques exercées par l'air sur la pale, on s'aperçoit qu'un rotor a toujours tendance à faire tourner le quadri-rotor dans le sens inverse de sa rotation.

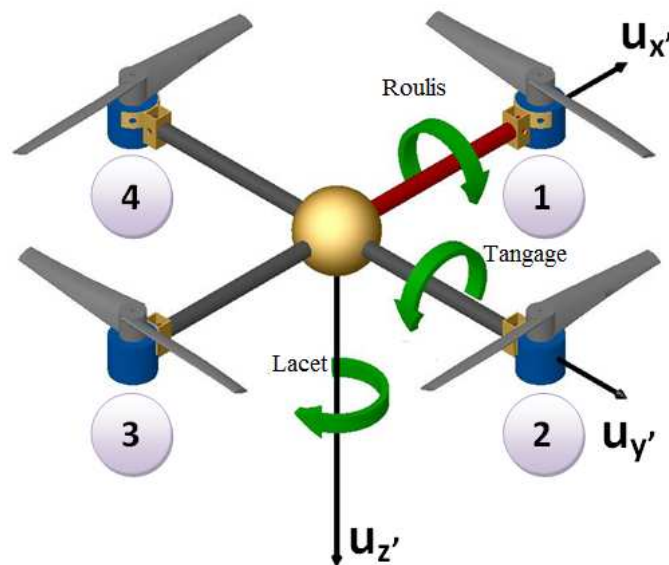


Figure I.1 : Les mouvements de quadri-rotor par rapport au référentiel mobil (x', y', z') .

De plus, pour plus de performances dans les commandes de tangage et de roulis, on choisit l'avant du quadri-rotor entre deux rotors et on fait tourner les moteurs qui sont face à face dans le même sens. Ainsi, les commandes des quatre moteurs seront utilisées pour modifier l'accélération verticale (commande de puissance) et faire tourner le quadri-rotor autour de l'axe vertical (commande de lacet).

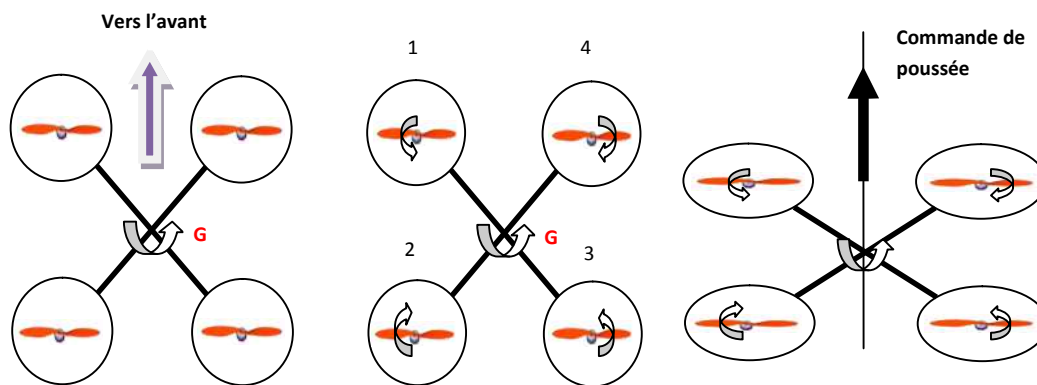


Figure I.2: La disposition et sens de rotation des différents moteurs.

I.2. Les mouvements de quadri rotor :

Les mouvements de base de quadri-rotor sont réalisés en variant la vitesse de chaque rotor changeant de ce fait la poussée produite. Le quadri-rotor incline vers la direction du rotor plus lent, qui tient compte alors de la translation le long de cet axe. Par conséquent, comme à un hélicoptère classique, les mouvements sont couplés, signifiant que le quadri rotor ne peut pas réaliser la translation sans roulement ou tangage, ce qui signifie qu'un changement de la vitesse d'un rotor se traduit dans un mouvement en au moins trois degrés de liberté. Par exemple, augmentant la vitesse de propulseur gauche aura comme conséquence un mouvement de roulis (le quadri-rotor incline vers le rotor plus lent, vers la droite), un mouvement de lacet (l'équilibre entre les rotors qui tourne dans le sens des aiguilles d'une montre et les rotors qui tourne dans le sens inverse est perturbé ayant pour résultat un mouvement de rotation horizontal), et une translation (le mouvement de roulis incline l'armature et avec lui, l'orientation de la force de poussée). Cet accouplement est la raison pour laquelle nous pouvons commander les six degrés de liberté de quadri-rotor avec seulement quatre commandes (le couple appliqué par les moteurs sur chaque propulseur).

I.3. Le quadri-rotor a cinq mouvements principaux :

- Mouvement vertical
- Mouvement de roulis
- Mouvement de tangage

- Mouvement de lacet
- Translations horizontales

I.3.1. Mouvement vertical :

Le mouvement ascendant et descendant est obtenu par la variation de la vitesse de rotation des moteurs (par conséquent la poussée produite), si la force de portance est supérieure au poids du quadri-rotor le mouvement est ascendant, et si la force de portance est inférieure au poids du quadri-rotor le mouvement est descendant.

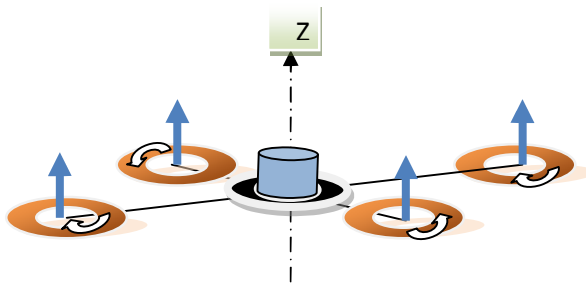


Figure I.3 : Demonstration de mouvement de glissement vertical.

I.3.2. Mouvement de roulis :

La figure montre comment un mouvement de roulis est obtenu. Dans ce cas, on applique un couple autour de l'axe x, c'est-à-dire en appliquant une différence de poussée entre le rotor 2 et le rotor 4. Ce mouvement (rotation autour de l'axe x) est couplé avec un mouvement de translation selon l'axe y.

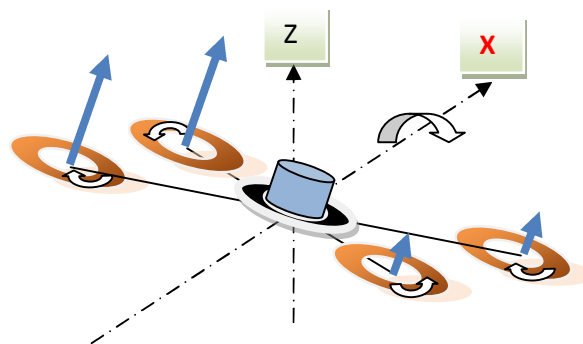


Figure I.4 : Demonstration de mouvement de roulis.

I.3.3. Mouvement de tangage :

La figure montre comment un mouvement de tangage est obtenu. Dans ce cas, on applique un couple autour de l'axe y , c'est-à-dire en appliquant une différence de poussée entre le rotor 1 et le rotor 3. Ce mouvement (rotation autour de y) est couplé avec un mouvement de translation selon l'axe x .

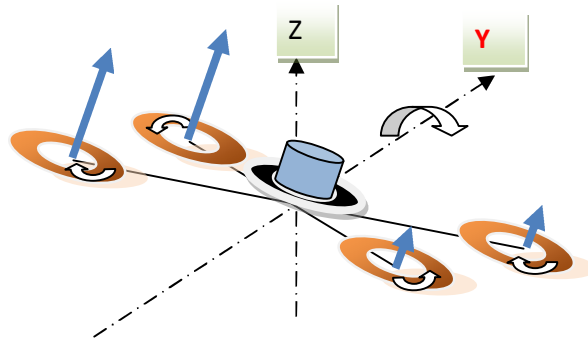


Figure I.5 : Demonstration de mouvement de tangage.

I.3.4. Mouvement de lacet :

La figure montre comment le mouvement de lacet est obtenu. Dans ce cas, nous voulons appliquer un couple autour de l'axe z , qui est fait en appliquant une différence de vitesse entre les rotors $\{1,3\}$ et $\{2,4\}$. Ce mouvement n'est pas un résultat direct de la poussée produit par les propulseurs mais par les couples réactifs produits par la rotation des rotors. La direction de la force de poussée ne décale pas pendant le mouvement, mais l'augmentation de la force de portance dans une paire de rotors doit être égale à la diminution des autres paires pour s'assurer que toute la force de poussée demeure la même.

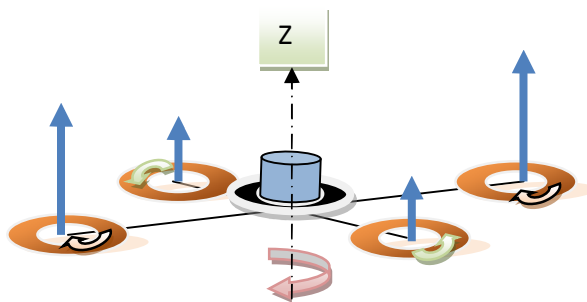


Figure I.6 : Demonstration de mouvement de lacet.

I.4. Le modèle dynamique du quadri-rotor :

Il peut être vu comme un système où les évolutions spatiales du quadri-rotor sont les sorties et la tension de chacun de ses moteurs sont les entrées. Le modèle est régi par les équations de la mécanique qui le font évoluer sur 6 axes. Il subit des forces d'accélération de différents types :

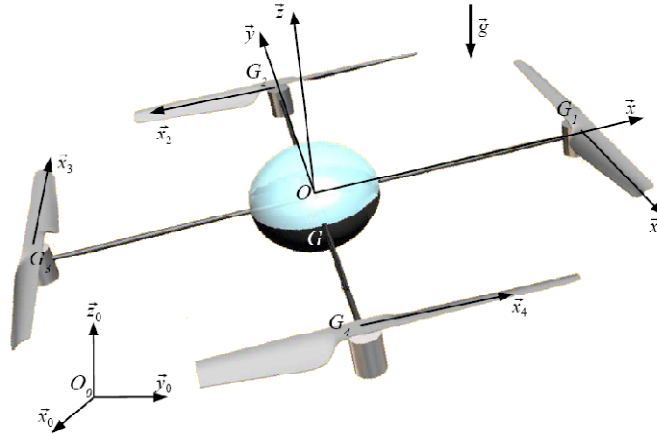


Figure I.7: Références fixe et mobile auquel est liée la quadri rotor.

I.4.1. La portance :

Générée par les quatre rotors en rotation, elle permet à l'appareil de s'élever si elle compense au moins la traînée. Elle s'écrit de la forme :

$$\tau_x = b \times L \times (\Omega_4^2 - \Omega_2^2) \quad ; \text{ Sur l'axe de roulis.}$$

$\tau_y = b \times L \times (\Omega_3^2 - \Omega_1^2) \quad ; \text{ Sur l'axe de tangage ; Avec } \Omega_i^2 : \text{ la vitesse angulaire de chaque moteur au carré en } (\text{rad/s})^2 . b : \text{ le coefficient de portée en } \text{kg} \times \text{m}/\text{rad}^2 .$

L : la demi-envergure du quadri- rotor en mètres.

I.4.2. La traînée :

Résultante des frottements de l'air sur le quadri-rotor, elle est parallèle et opposée à la trajectoire. Son expression est : $\tau_z = d \times (\Omega_1^2 + \Omega_3^2 - \Omega_2^2 + \Omega_4^2) .$

d : Le coefficient de traînée en $\text{kg} \times \text{m}^2/\text{rad}^2 .$

I.4.3. L'effet gyroscopique :

Lorsque le quadri-rotor est en rotation sur deux axes, cette force apparaît sur le troisième axe et tend à résister aux mouvements du quadri-rotor.

Elle vaut :

$$\tau_x = I_{\text{rotor}} \times \omega_y \times (\Omega_3 + \Omega_1 - \Omega_2 - \Omega_4) .$$

Et

$$\tau_y = I_{\text{rotor}} \times \omega_x \times (-\Omega_3 - \Omega_1 + \Omega_2 + \Omega_4) .$$

Avec I_{rotor} le moment d'inertie du moteur en $(\text{kg} \times \text{m}^2)$ et ω_x la vitesse angulaire suivant l'axe des X en (rad/s) .

En projetant les trois forces précédentes et en ajoutant l'effet sur l'accélération des moments d'inertie sur chaque axe, le quadri-rotor réagit alors en roulis, tangage et lacet de la manière suivante :

Équation de l'accélération angulaire suivant l'axe de roulis :

$$\ddot{\phi} = \dot{\theta} \dot{\psi} \times \frac{(I_y - I_x)}{I_x} + b l \frac{(\Omega_4^2 - \Omega_2^2)}{I_x} + \dot{\theta} I_{\text{rotor}} \times \frac{(\Omega_1 - \Omega_2 + \Omega_3 - \Omega_4)}{I_x} .$$

Équation de l'accélération angulaire suivant l'axe de tangage :

$$\ddot{\theta} = \dot{\phi} \dot{\psi} \times \frac{(I_z - I_x)}{I_y} + b l \frac{(\Omega_3^2 - \Omega_1^2)}{I_y} + \dot{\phi} I_{\text{rotor}} \times \frac{(\Omega_1 - \Omega_2 + \Omega_3 - \Omega_4)}{I_y} .$$

Équation de l'accélération angulaire suivant l'axe de lacet :

$$\ddot{\psi} = \dot{\theta} \dot{\phi} \times \frac{(I_x - I_y)}{I_z} + d \times \frac{(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2)}{I_z} .$$

Aussi, le quadri-rotor subit des accélérations sur les 3 axes de l'espace. Ces accélérations dépendent de la portance générée par les moteurs notée T_i qui vaut : $T_i = b \times \Omega_i^2$ avec $i \in (1, 2, 3, 4)$.

Du principe de la dynamique, chaque moteur développe une force :

$F = m \times g$ et $F = \sum T_i$; projetées dans les 3 dimensions de l'espace, ces accélérations se résument de la sorte :

Équation de l'accélération suivant l'axe x :

$$\ddot{x} = \frac{(\cos\psi \times \sin\theta \times \cos\phi) + (\sin\psi \times \sin\phi)}{m} \times \sum_{i=1}^4 T_i$$

Équation de l'accélération suivant l'axe y :

$$\ddot{y} = \frac{(\sin\psi \times \sin\theta \times \cos\phi) + (\cos\psi \times \sin\phi)}{m} \times \sum_{i=1}^4 T_i$$

Équation de l'accélération suivant l'axe z :

$$\ddot{y} = -g + \frac{(\cos\theta \times \cos\phi)}{m} \times \sum_{i=1}^4 T_i.$$

I.4.4.Fonction de transfert de l'ensemble moteur-variateur :

La fonction de transfert d'un moteur électrique est du second ordre et s'écrit :

$$H(p) = \frac{k}{k^2 + (R+Lp) \times (f+Jp)}.$$

K : gain du moteur en volt \times s/rad.

R : résistance interne du moteur

L : inductance

f : frottements

J : inertie du rotor

Cependant, on peut approximer cette fonction de transfert par un premier ordre. En effet, la résistance d'un moteur est de l'ordre d'Ohm ou de la dizaine d'Ohms, alors que l'inductance est de l'ordre du milli Henry. De même, les frottements d'un moteur électrique sont négligeables devant son inertie. On obtient ainsi :

$$H(p) = \frac{k}{k^2 + RIp}$$

$$H(p) = \frac{k}{1 + \tau p}$$

k : gain du moteur en rad/s/volt

τ : constante de temps de l'ensemble moteur+variateur.

A noter que le gain du moteur est donné par le constructeur

I.5.Modèle SimuLink du quadri-rotor :

Connaissant les relations entre les poussées des différents moteurs et les accélérations angulaires. On peut établir un modèle sous SimuLink. Le but est de déterminer le type de correcteurs à mettre en place afin d'assurer une bonne stabilisation du drone sur les trois axes de lacet, tangage et roulis.

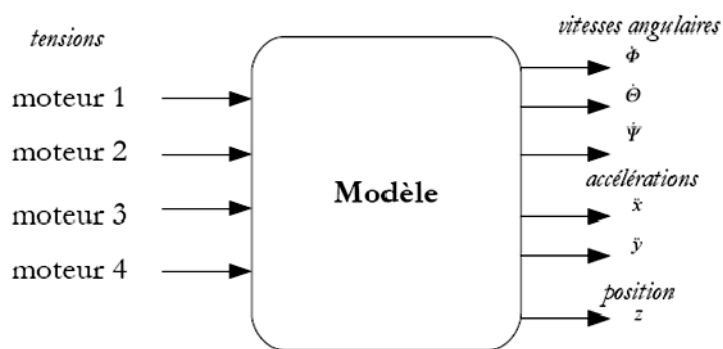
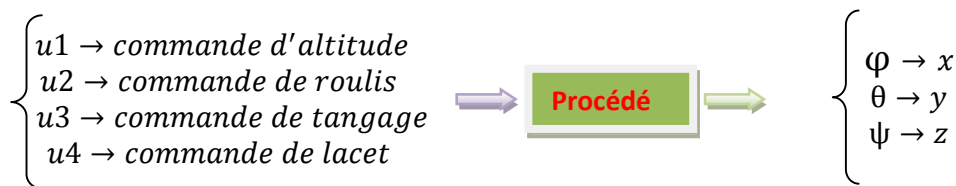


Figure I.8: Commande global de procédé de stabilisation.

Une fois les différentes forces modélisées, il est évident qu'un contrôleur agissant par le biais d'un correcteur PID est indispensable. Ce dernier reçoit l'écart entre l'angle du quadri-rotor donné par intégration des gyroscopes et la consigne en

angle, et l'écart entre sa position donnée par une double intégration des accéléromètres et la consigne en position.

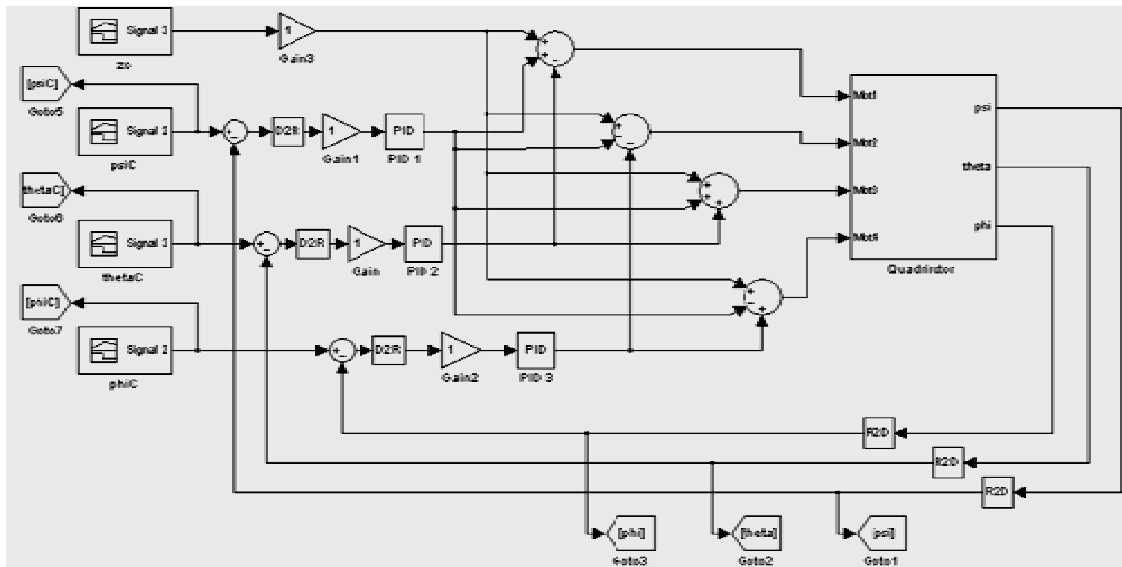


Figure I.9: Schéma du modèle sous SimuLink.

I.6.Modèle dynamique du moteur :

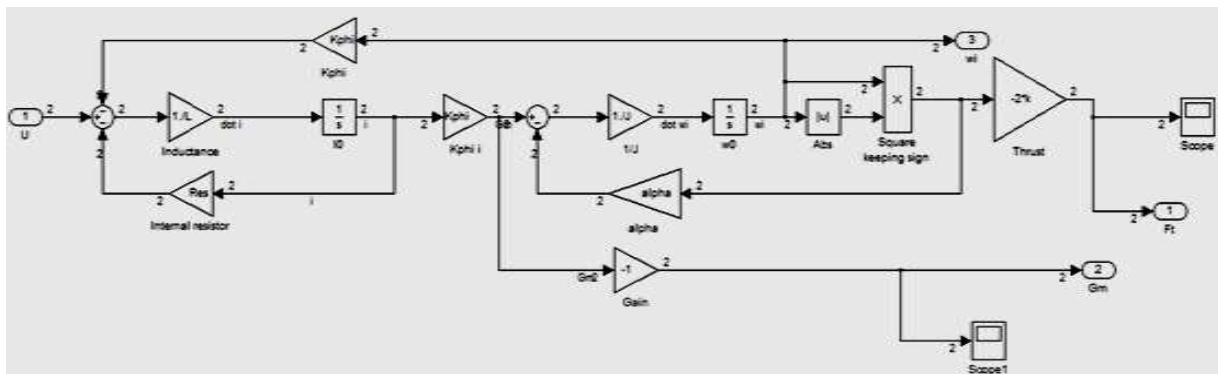


Figure I.10: Model dynamique de moteur.

- Le drone est piloté grâce à quatre consignes : les consignes de puissance ZC (qui permet de contrôler l'altitude), de roulis θ_C , de tangage ϕ_C et de lacet ψ_C . Ces consignes sont ensuite envoyées aux différents moteurs. La consigne de puissance est envoyée à tous les moteurs, tout comme le lacet. Le tangage et le roulis sont obtenus en contrôlant seulement deux moteurs (1 et 3 pour le tangage et 2 et 4 pour le roulis).
- Le sous-système Quadri-rotor modélise les équations mécaniques explicitées précisément.

- Les correcteurs implémentés sont des correcteurs Proportionnel-Intégral-Dérivé (PID).
- Les différents gains que l'on peut observer permettent de convertir les angles (radians) en tensions (volts) qui sont envoyées aux variateurs. Les valeurs de ces gains nous sont inconnues. Cependant, il est toujours possible d'étudier le type de correcteurs à mettre en place afin d'obtenir une dynamique satisfaisante.

I.7. La stabilisation simplifiée à l'aide de mixeur:

La stabilisation et la commande à l'aide de mixeurs et de gyroscopes. A partir des équations précédentes traitant le principe de vol d'un quadri rotor, nous tirons directement les 4 équations correspondant aux commandes des 4 moteurs :

$$M1 = P - L + T.$$

$$M2 = P + L - R.$$

$$M4 = P + L + R.$$

$$M3 = P - L - T.$$

Où P : puissance ; L : lacet ; T : tangage ; R : roulis.

Le schéma de la figure suivante explique les différentes connections réalisées pour la solution mixeurs-gyroscopes. Les gyroscopes peuvent être intercalés à la sortie des voies 1, 2 et 4, qui correspondent respectivement aux commandes de roulis, de tangage et de lacet.

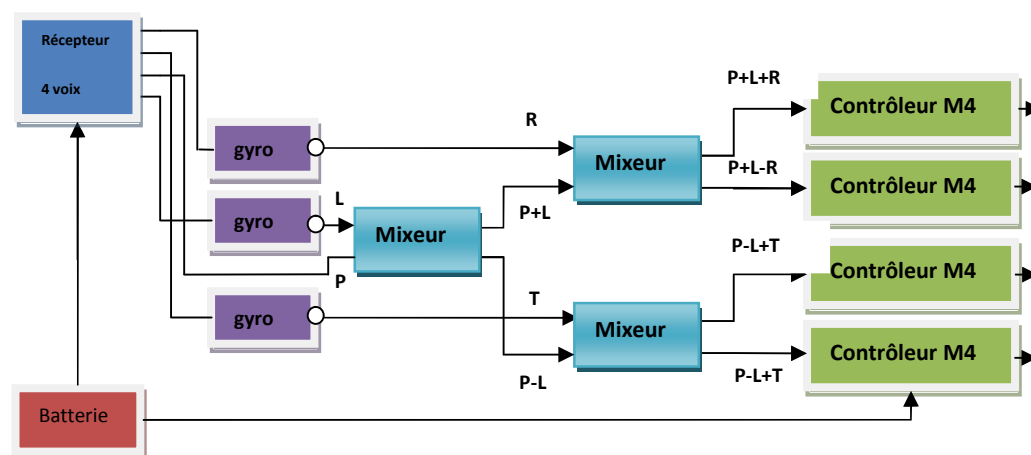


Figure I.11: Solution mixeurs-gyroscopes.

I.8.Solution à base d'une carte a microcontrôleur embarqué :

L'utilisation de la carte microcontrôleur permet d'effectuer les mêmes opérations que les mixeurs dans la solution précédente mais avec l'avantage vient de la possibilité d'implémenter des correcteurs (proportionnels, intégrales et dérivés) numériques. La Figure ci-dessous présente le schéma globale de cette deuxième solution pour la commande et la stabilisation du drone.

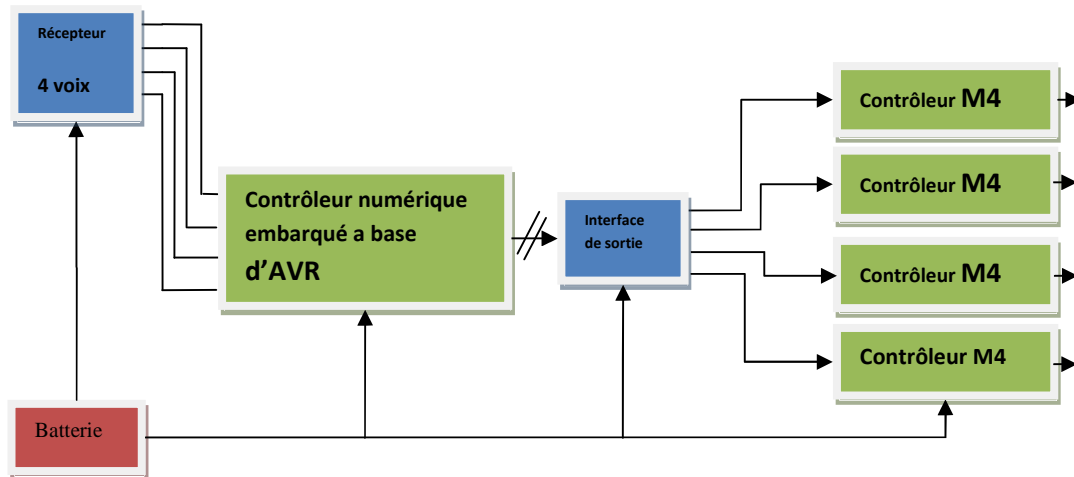


Figure I.12: Solution carte numérique embarqué.

I.9.La carte mère embarquée de contrôle de vol (Flight Control) :

Le contrôleur de vol (Flight-Ctrl) est la carte mère (carte principale) du drone. Elle contient le processeur principal et tous les capteurs nécessaires à un vol stabilisé. De tous les capteurs, les plus importants sont ceux concernant la vitesse de rotation. Le programme s'en sert pour déterminer la position dans les airs et pour compenser les perturbations extérieures (vent, ...). Pour chaque axe (x, y et z), un capteur de vitesse de rotation est nécessaire, soit trois éléments au total.

Habituellement, ces capteurs de vitesse de rotation sont appelés «Gyroscopes » ou « Gyros » et ils mesurent la vitesse angulaire en degrés par seconde.

Un autre capteur est celui des accélérations. Il détecte les accélérations sur les trois axes. Le capteur d'accélération verticale est également capable de mesurer l'angle du drone par rapport à l'horizon. Habituellement, ils sont appelés "Accéléromètres", ou "ACC". Il est possible de voler sans, mais ces capteurs permettent à l'appareil de rester

à niveau. Cela permet de lâcher les manettes et l'appareil se remet à plat. Sans ces capteurs il continue à voler avec son angle de départ. Ce type de vol est appelé "Conservation d'Assiette" (« "Heading Hold" en anglais) ».

Le contrôleur de vol (Flight-contrôler..) peut être doté de capteur de pression « baromètre ». Cela permet au drone de stabiliser sa hauteur de vol.

Une fois reliée à un PC, le contrôleur peut être consultée et configuré avec un logiciel qui prendra en compte les différents paramètres relatifs au matériel connectés :

I.10. Le schéma de block qui montre les différentes fonctions d'un contrôleur de vol :

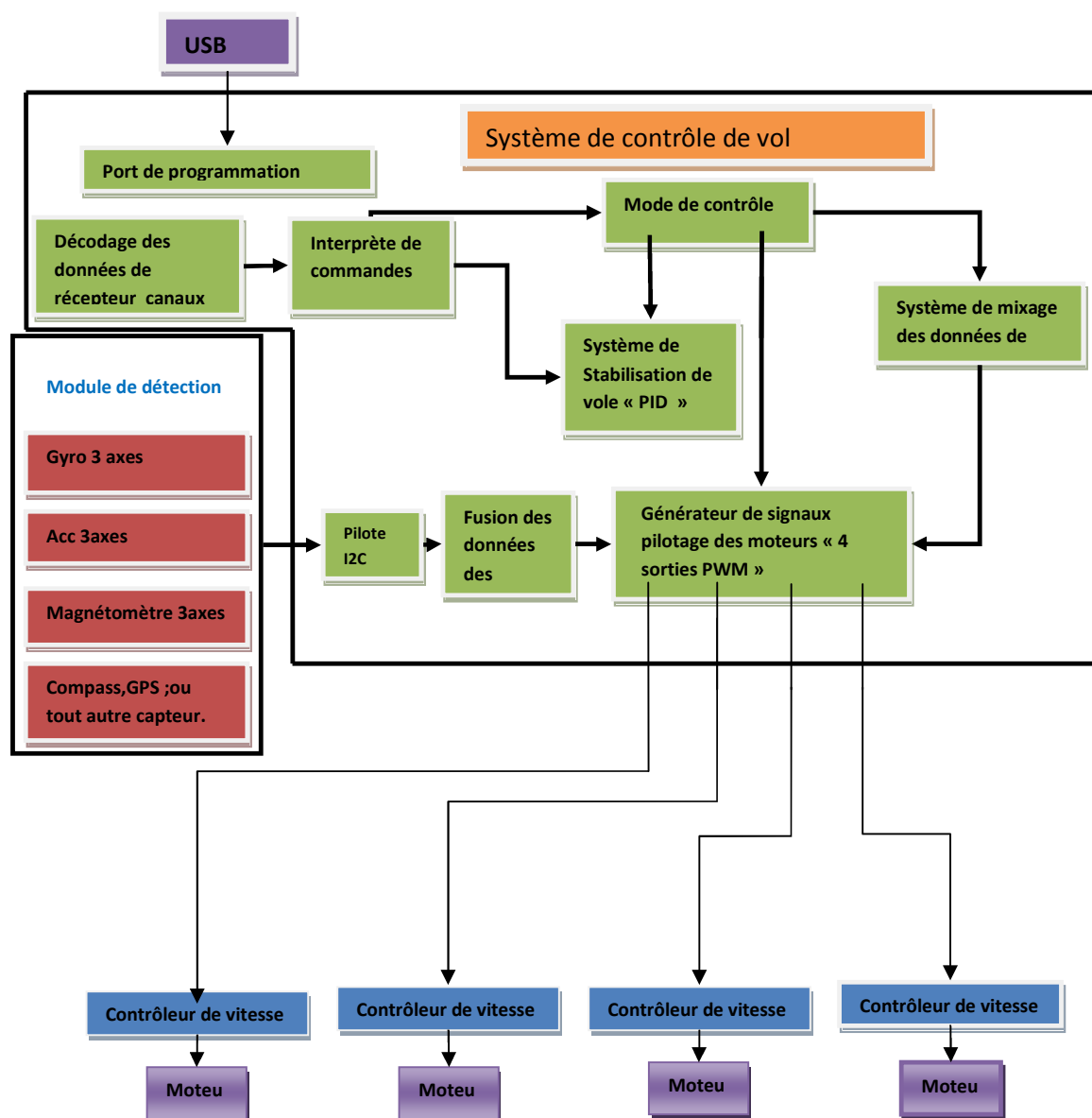


Figure I.13: Les fonctions de différents blocs servants au contrôle automatique de vol.

I.11. Contrôle de stabilisation:

La boucle de contrôle d'équilibre couramment utilisé dans les contrôleurs de vole est montrée dans la figure suivante.

L'entrée des signaux RC-roulis, RC-tangage, RC-lacet à partir de récepteur ou via le PC sont mises a jours dans une période de 1ms puis comparés avec les données des capteurs et alimentant de la boucle de verrouillage basé sur le mixeur PID. Les signaux résultants pilotent est stabilise le model.

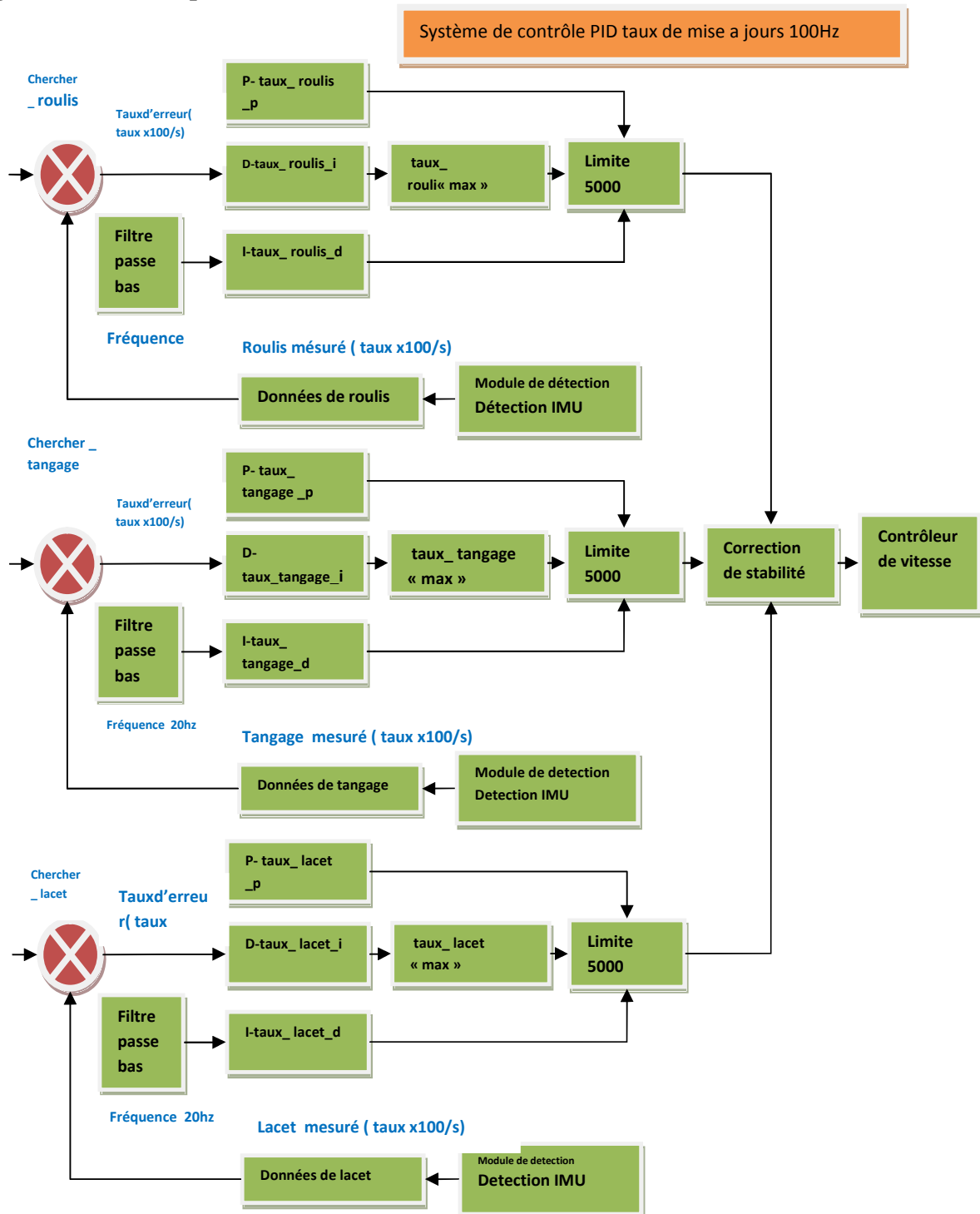


Figure I.14: Boucles de control PID pour le mode stabilisé.

I.12.Conclusion :

Dans ce chapitre l'étude c'est porté sur la stabilisation de drone suivant deux méthodes: l'une contenant la solution mixeurs-gyroscopes qui se base sur le model dynamique de drone simulé sur Matlab et l'autre constituée de la carte de contrôle de vol numérique qui a permis de le stabiliser le drone sur ces trois angles (grâce à un correcteur PID). Néanmoins, après cette étude il est nécessaire de se demander quelles performances doit réaliser le drone. En effet, le drone se doit de posséder une bonne stabilité mais aussi une bonne manœuvrabilité. Dans ce chapitre, nous nous sommes seulement intéressés à la stabilité (statique et dynamique). Cependant, la stabilité dégrade la manœuvrabilité. Pour déplacer le drone, il faut en effet l'écarter de la position d'équilibre afin de déclencher le mouvement. Il serait intéressant de trouver un compromis entre stabilité et manœuvrabilité. L'automatisation complète du drone serait alors possible en concevant une carte de contrôle de vol intelligente.

Chapitre II:

Processeurs de traitement et de control.

II.1. Introduction :

Dans le contexte d'étudier la réalisation des différentes tâches relatives au contrôle de vol d'une part et le contrôle des moteurs et la gestion des communications d'autre part sur le drone ; un système à base de microcontrôleur embarqué et performant paraît évident. Parmi ces derniers, les produits AVR d'Atmel semblent les plus adaptés.

II.2. Présentation et raison de choix des AVR :

Les AVR sont des microcontrôleurs (μ C) 8bits RISC d'ATMEL très performants (architecture de Harvard), dont la famille est très étendue. On y trouve des petits (ATtiny11, 8 pattes) et des très gros (ATmega128, 64 pattes,). Ils ont tous en commun le cœur (CPU) ; seuls les périphériques, la mémoire flash et la RAM qui diffèrent. Et c'est là un atout de taille, car un code écrit pour un ATmega8 fonctionnera sans modifications majeures sur un ATmega328 par exemple (à une recompilation près). Un autre avantage en leur faveur face à d'autres μ C comme les PICs par exemple, c'est la suite de logiciels libres permettant de compiler, programmer, déboguer les AVR, en live, in situ.

II.3. ATmega328P et atmega8 :

L'ATmega328P et ATmega8 sont deux microcontrôleurs de la famille AVR qui équipent certains des modèles les plus populaires de cartes Arduino. Il se caractérise notamment par ses nombreuses broches d'entrées/sorties et par sa mémoire (32 KB de flash, 1KB d'EEPROM et 2KB de RAM). à noter que les deux versions sont compatibles de côté matériel (même nombre d'entrées sorties) et logiciel, néanmoins l'ATmega328P est plus récente et contient une mémoire flash beaucoup plus étendue « 8ko » donc permet d'accepter des programmes plus volumineux. Plus d'infos dans le catalogue « datasheet ATmega328P et ATmega8 ».

Ces deux versions seront utilisées dans notre projet pour effectuer les différentes fonctions de contrôle de drone. Le ATmega328P sera le cœur de contrôleur de vol principale qui est gourmand en termes d'instructions de programme. En effet ce dernier doit gérer toutes les fonctions de stabilisation, de décodage et traitement des

communications, commande des contrôleurs de vitesse. et autres et cela en temps réel, ce qui encor demande beaucoup de ressources en terme d'interruptions.



Figure II.1: atmega328p (version 28 broches).

D'autre part le Atmega8 est requis pour ces caractéristiques relatives aux nombres de modules PWM (de nombre 6) attachées aux pates de sorties qui attaquerons les drivers de Mosfet de puissance. Trois autres pates auquel sont reliées des comparateurs nécessaires sont disponibles dans cette AVR. A noter que c'est la version SMD qui sera employé pour les contrôleurs des moteurs.



Figure II.2: Atmega8A (version 32 broches).

Pour utiliser ce microcontrôleur sur un montage, on doit l'alimenter avec une source de tension régulée. Les tensions de 5V ou 3.3V sont populaires, mais l'ATmega328P autorise d'autres tensions d'alimentation. Ceci dit, ces détails se trouvent dans les schémas électriques de l'appareil.

II.4.Programmation des microcontrôleurs AVR:

- Programmation in-situ (ISP) : c'est à dire qu'on reprogramme le μ C alors qu'il est dans le montage final, avec un câble adapté branché que le port parallèle de l'ordinateur ;

- Utilisation du câble JTAG : qui permet de reprogrammer, débogger, exécuter pas à pas, etc.
- Utilisation d'un bootloader : petit morceau de code qui reste en mémoire et qui communique via une interface série avec l'ordinateur lorsque l'on veut reprogrammer le μ C.

II.5.Interface SPI :

Les microcontrôleurs de la famille AVR utilisent l'interface SPI pour permettre de faire de la programmation sur site (*In Site Programming – ISP*). Ce mode de programmation nécessite un *programmeur externe*. Il en existe de nombreuses versions avec des fonctionnalités plus ou moins évoluées. Dans sa version de base, c'est un petit dispositif électronique qui est relié d'une part au port USB de l'ordinateur, et de l'autre au microcontrôleur cible, via l'interface SPI.

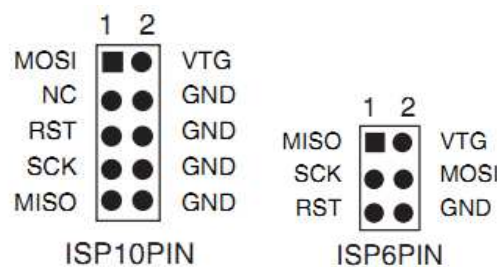


Figure II.3 : deux types d'interfaces spi.

Concrètement l'interface SPI est composée de trois lignes de communication: MISO, MOSI et SCK. La ligne supplémentaire /RESET (formellement /SS pour Slave Select) sert à commuter le circuit cible en mode esclave – ce qui correspond pour le microcontrôleur au mode programmation. En plus, le programmeur peut généralement fournir la tension d'alimentation pour le microcontrôleur (V_{cc}). Qu'on utilise l'alimentation fournie par le programmeur ou pas, il est impératif d'interconnecter les masses (GND).

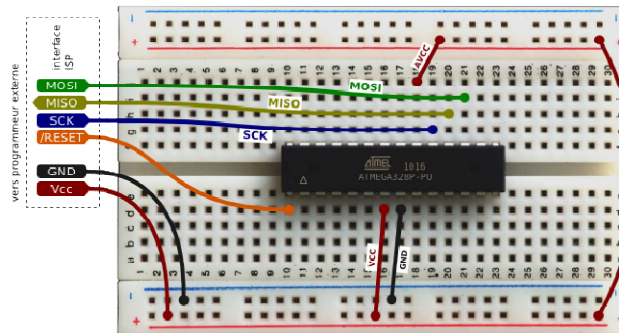


Figure II.4: *Connections spi d'un ATmega328P.*

II.6. Le programmeur d'AVR:

Parmi les différents programmeurs disponibles, l'un pourra porter son choix sur un programmeur parallèle ;série ou USBasp ou un de ses *clones bon marché* doté d'interface SPI utilisant un connecteur 2×5 broches qui reprend à la fois les lignes de l'interface SPI (MISO, MOSI et SCK), ainsi qu'une source de tension pour alimenter le microcontrôleur à programmer (Vcc), et une référence de masse (GND). La ligne /R Connecte la machine de développement et la cible AVR.



Figure II.5: *Clone d'un programmeur d'AVR.*

Une fois que le matériel est prêt, on raccorder le PC de développement à la cible en suivant les instructions spécifiques au programmeur. Pour les petits programmeurs bon marché, cela signifiera certainement relier le programmeur au PC par un câble USB, et le programmeur à la cible via l'interface SPI.

II.7. Qu'est-ce qu'Arduino :

Le système Arduino est un outil pour fabriquer de petits ordinateurs qui peuvent capter et contrôler davantage de choses du monde matériel que votre ordinateur de

bureau. C'est une plateforme open-source (gratuite) d'électronique programmée qui est basée sur une simple carte à microcontrôleur (de la famille AVR), et une portion de programme appelé « boot loader », véritable environnement de développement intégré, pour écrire, compiler et transférer le programme vers la carte à microcontrôleur.

Arduino peut être utilisé pour développer des objets interactifs, pouvant recevoir des entrées d'une grande variété d'interrupteurs ou de capteurs, et pouvant contrôler une grande variété de lumières, moteurs ou toutes autres sorties matérielles. Les projets Arduino peuvent être autonomes, ou bien ils peuvent communiquer avec des logiciels tournant sur ordinateur (tels que Flash, Processing ou MaxMSP). Les cartes électroniques peuvent être fabriquées manuellement ou bien être achetées pré-assemblées; le logiciel de développement open-source peut être téléchargé gratuitement.

Le langage de programmation Arduino est une implémentation de Wiring , une plateforme de développement similaire, qui est basée sur l'environnement multimédia de programmation Processing.

II.7.1. Objectif de l'Arduino :

Il y a de nombreux microcontrôleurs et de nombreuses plateformes basées sur des microcontrôleurs disponibles pour l'électronique programmée. Parallax Basic Stamp...Etc. Et beaucoup d'autres qui offrent des fonctionnalités comparables. Tous ces outils prennent en charge les détails compliqués de la programmation des microcontrôleurs et les intègrent dans une présentation facile à utiliser. De la même façon, le système Arduino simplifie la façon de travailler avec les microcontrôleurs, tout en offrant plusieurs avantages pour les enseignants, les étudiants et les amateurs intéressés par les autres systèmes :

Pas cher : les cartes Arduino sont relativement peu coûteuses comparativement aux autres plateformes. La moins chère des versions du module Arduino peut être assemblée à la main, et même les cartes Arduino pré-assemblées coûtent moins de 30,000DA (microcontrôleur inclus...).

Multiplateforme : Le logiciel Arduino, écrit en Java, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux. La plupart des systèmes à microcontrôleurs sont limités à Windows.

Un environnement de programmation clair et simple: L'environnement de programmation Arduino (le logiciel Arduino) est facile à utiliser pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.

II.8. Configurer l'IDE Arduino :

Les versions récentes de l'IDE Arduino permettent d'utiliser un programmeur externe pour télécharger un programme. Par contre, il n'y a pas en standard d'option dans la liste des *cartes* pour programmer directement un microcontrôleur *ATmega328P nu*. Une petite étape de configuration est donc à prévoir avant la première utilisation.

La configuration va consister à ajouter un nouveau dossier dans *votre* répertoire *sketchbook/hardware* et à y ajouter la description de votre matériel cible. À savoir ici, un *ATmega328P nu* (sans bootloader) dans sa *configuration d'usine*.

II.9. Programmer un ATmega328P nu avec Arduino :

Pour tester la programmation d'un *ATmega328P*, on commence par réaliser le montage basique et classique qui consiste à faire clignoter une LED reliée aux broches du microcontrôleur. Ici, on a connecté l'anode de la LED à la broche 5 du microcontrôleur, et la cathode de la LED à la broche 6 via une résistance de 220Ω . Comme on peut le voir dans la table de correspondance ci-dessous, ces deux broches correspondent respectivement aux broches Arduino *digital pin 3* et *digital pin 4*.

description de Atmega328

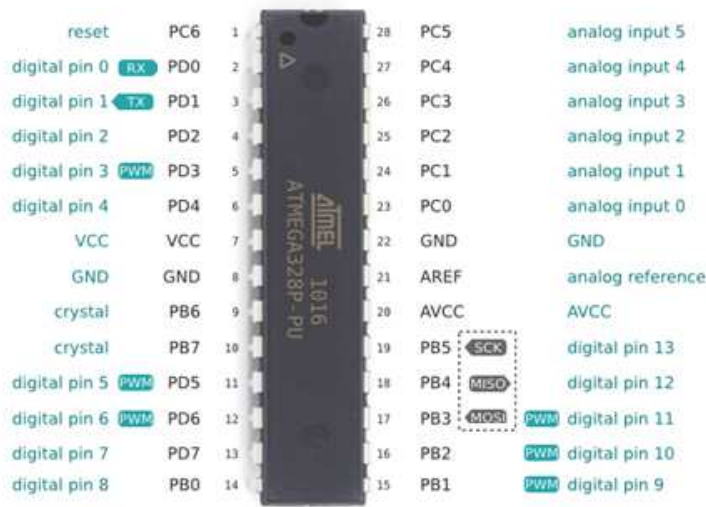


Figure II.6: Correspondance entre les broches de l'ATmega328 et celles de l'Arduino.

On doit également sélectionner une carte utilisant le *même* microcontrôleur que celui qu'on possède. Exemple *Arduino Nano w/ATmega328*.

II.10. Télécharger le programme vers la cible :

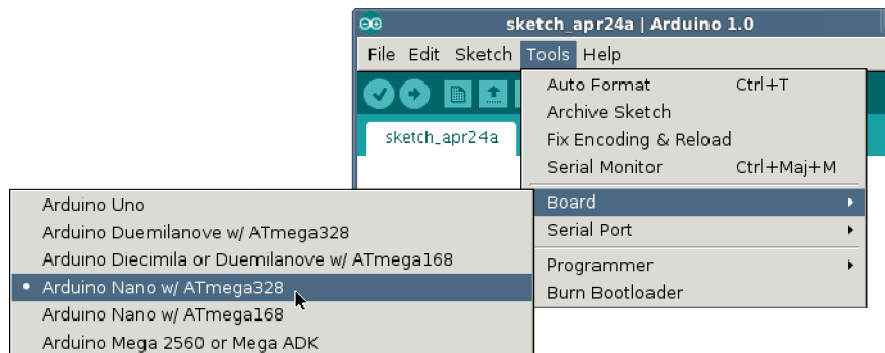


Figure II.7: Sélection de carte cible l'ATmega328P.

Une fois le programme saisi, et après avoir sélectionné comme carte cible l'ATmega328P nu, on doit télécharger le programme *exactement comme d'habitude*, c'est à dire en utilisant le bouton *Upload* de l'IDE (ou en passant par le menu *File > Upload*):



Figure II.8: Bouton permettant la compilation de code source.

II.10.1. Télécharger « uploader » un programme avec l'IDE Arduino :

Le bouton *Upload* de l'IDE Arduino permet de compiler et télécharger un programme vers la cible.

Si l'on préfère, on pourra également utiliser le menu *File > Upload using programmer* pour télécharger le programme. L'option *USBasp* devrait être présente en standard:

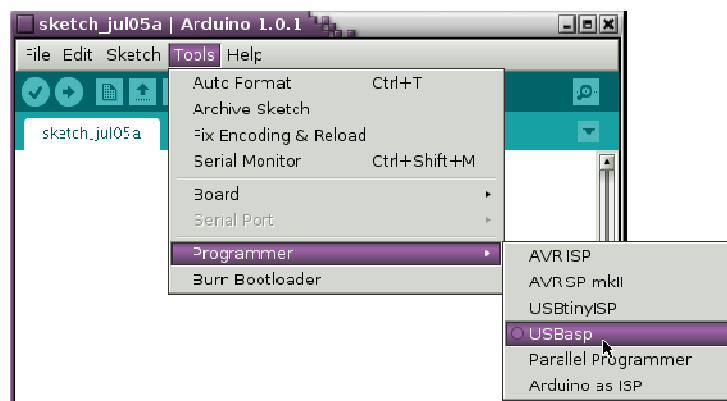


Figure II.9: Choix de programmeur USBASP.

Une fois le téléchargement terminé, vous devriez voir clignoter la LED. En cas d'échec, on commence par vérifier que le programmeur est bien relié à la cible. On peut aussi examiner les messages de téléchargement qu'on peut trouver en bas de la fenêtre de l'IDE: ceux-ci peuvent contenir des informations utiles pour déterminer et spécifier les bugs qui peuvent survenir.

II.11. Matériel Arduino :

Il existe plusieurs versions de cartes Arduino. La version courante de base, la Duemilanove (2009 en italien ; les créateurs de l'Arduino sont des italiens...), utilise le microcontrôleur Atmel. La version précédente Diecimila, et les premières versions de la Duemilanove utilisent le microcontrôleur ATmega168 puis Arduino uno avec Atmega328 tandis que les cartes plus anciennes utilisent le microcontrôleur ATmega8. La carte Arduino Mega elle est basée sur l'ATmega1280.

II.11.1. La Duemilanove :

C'est la révision la plus récente de la carte de base Arduino USB. Elle se connecte sur l'ordinateur avec un câble USB standard et contient tout ce dont nous avons besoin pour programmer et utiliser la carte. Elle peut être complétée avec toute une variété de circuits : des cartes-filles personnalisées avec des caractéristiques spécifiques.



Figure II.10: Carte arduino duemilanove.

II.11.2. Arduino UNO :

Le modèle UNO de la société ARDUINO est une carte électronique dont le coeur est un microcontrôleur ATMEL de référence ATmega328. Le microcontrôleur ATmega328 est un microcontrôleur 8bits de la famille AVR dont la programmation peut être réalisée en langage C.

L'intérêt principal des cartes ARDUINO (d'autres modèles existent) est leur facilité de mise en oeuvre. ARDUINO fournit un environnement de développement s'appuyant sur des outils open source.

Le chargement du programme dans la mémoire du microcontrôleur se fait de façon très simple par port USB.

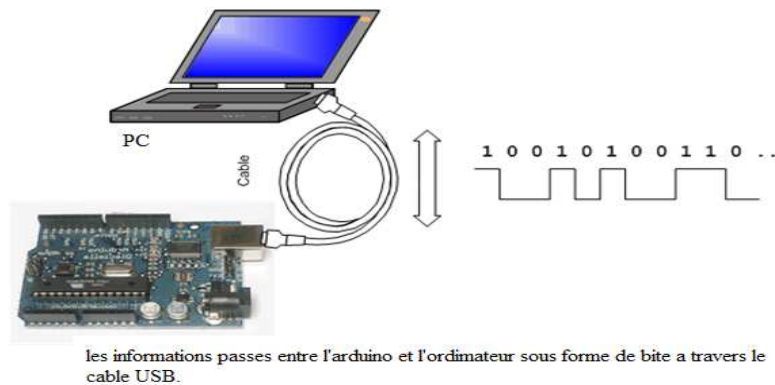


Figure II.11: Programmation d'une carte arduino UNO via UART intégrée.

En outre, des bibliothèques de fonctions "clé en main" sont également fournies pour l'exploitation d'entrées-sorties courantes : gestion des E/S TOR, gestion des convertisseurs ADC, génération de signaux PWM, exploitation de bus TWI/I2C, exploitation de servomoteurs ...



Figure II.12: Carte Arduino UNO.

II.11.3.La carte Mega :

Une carte Arduino aux plus grandes capacités, plus puissante, au circuit compatible avec les cartes Duemilanove et Diecimila.



Figure II.13 : Carte arduino mega.

II.12.Description de l'arduino IDE :

Le logiciel Arduino a pour fonctions principales :

- de pouvoir écrire et compiler des programmes pour la carte Arduino ;
- de se connecter avec la carte Arduino pour y transférer les programmes ;
- de communiquer avec la carte Arduino.

Cet espace de développement intégré (IDE) dédié au langage Arduino et à la programmation des cartes Arduino comporte :

- une BARRE DE MENUS comme pour tout logiciel une interface graphique (GUI) ;
- une BARRE DE BOUTONS qui donne un accès direct aux fonctions essentielles du logiciel et fait toute sa simplicité d'utilisation ;
- un EDITEUR (à coloration syntaxique) pour écrire le code de nos programmes ;
- avec onglets de navigation ;
- une ZONE DE MESSAGES qui affiche, indique l'état des actions en cours ;
- une CONSOLE TEXTE qui affiche les messages concernant le résultat de la compilation du programme.

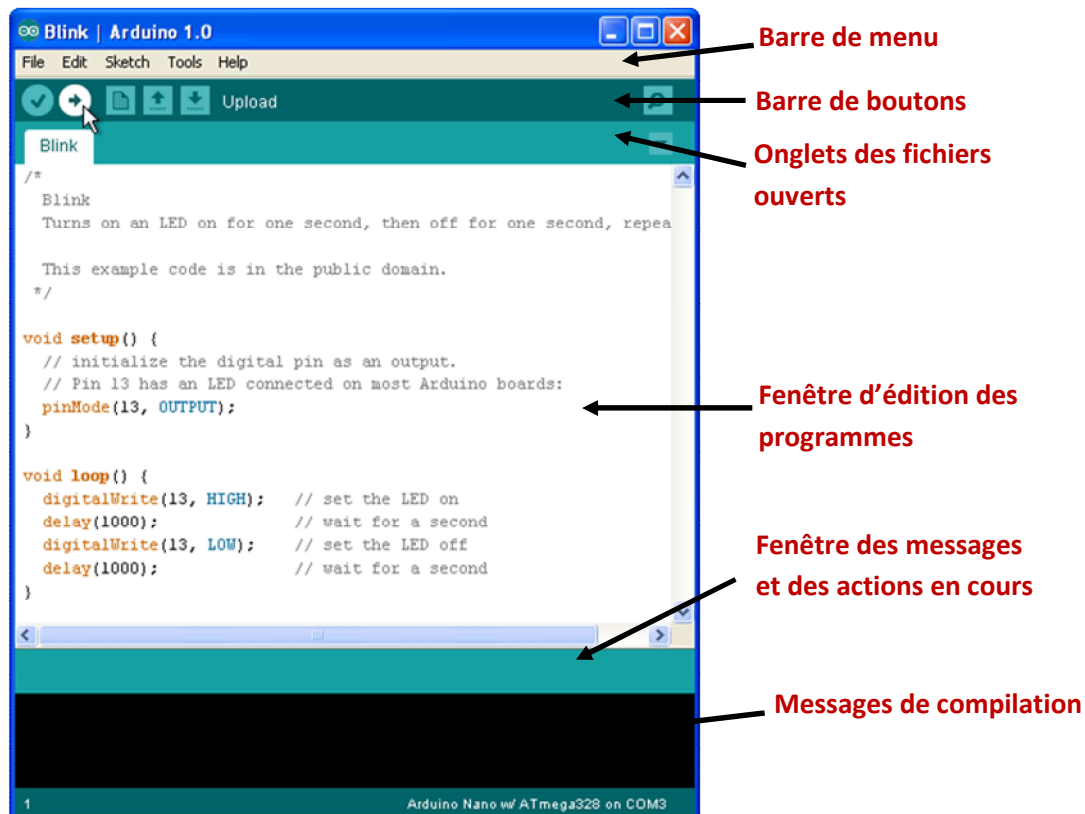


Figure II.14: Fenêtre principale d'édition des programmes Arduino.

II.13. Transfert des programmes vers la carte Arduino :

On suppose ici qu'un programme correctement écrit se trouve dans la fenêtre éditeur. Pour votre première programmation de la carte, aller dans le menu **File>Examples>Digital>Blink** : un programme s'ouvre avec du code dans la fenêtre éditeur. Il revient enfin d'appuyer alors sur le bouton **Verify** de la barre d'outils pour lancer la vérification du code. Si tout va bien, aucun message d'erreur ne doit apparaître dans la console, et la zone de message doit afficher **Done Compiling** attestant que la vérification s'est bien déroulée.

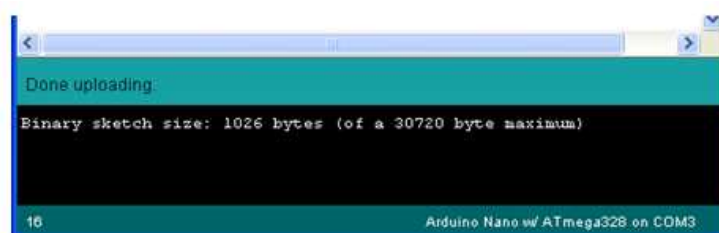


Figure II.15: Le programme transféré avec succès se lance.

II.14. Structure d'un programme Arduino :

Un programme Arduino (ou séquence - "sketch" en anglais) désigne un fichier de code écrit en langage Arduino que l'on peut transférer après compilation et exécuter sur une carte Arduino. Le code programme est constitué de :

II.14.1. Les commentaires :

Les premières lignes du programme Blink sont des *commentaires* :

```
/*
```

```
Ceci est un commentaire...
```

```
*/
```

Tout ce qui est entre `/*` et `*/` (signes de début et de fin d'un commentaire sur plusieurs lignes) est ignoré par le compilateur Arduino quand il compile le programme (autrement dit, les commentaires ne sont pas programmés dans la carte Arduino). Les commentaires du début de programme sont là pour les personnes qui liront le code : pour expliquer ce que le programme fait, comment il fonctionne ou pourquoi il est écrit de cette façon. C'est une bonne habitude de commenter (abondamment...) les programmes, et de mettre à jour les commentaires quand on modifie le code. Cela aidera les autres (et parfois soi-même..) pour comprendre le code afin de le modifier par exemple.

Il y a une autre manière de faire des commentaires, plus courts, sur une seule ligne. Ceux-ci commencent par `//` et se terminent à la fin de la ligne. Par exemple, dans la ligne :

```
int brocheLED = 13; // LED connectée à la broche numérique 13
```

Le message « LED connectée à la broche numérique 13 est un commentaire.

II.14.2.Variables :

Une variable est un emplacement mémoire utilisé pour stocker une donnée. Une variable a un nom, un type et une valeur. Par exemple la ligne du programme Blink ci-dessus déclare une variable appelée `brocheLED`, de type `int` et de valeur initiale 13 qui sera dans ce cas le numéro de la patte. L'avantage de l'utilisation d'une variable est qu'il est ainsi beaucoup plus facile de changer la broche utilisée pour connecter la LED : on aura seulement besoin de modifier la ligne unique où la valeur est attribuée à la variable. Souvent, cependant, la valeur de la variable changera durant l'exécution du programme. Par exemple, vous pourrez stocker la valeur lue depuis une entrée dans une variable.

II.14.3.Fonctions :

Une fonction (également désigné sous le nom de procédure ou de sous-routine) est un morceau de code ayant un nom et que vous pouvez utiliser à un autre endroit dans un programme. Par exemple, voici la définition de la fonction `setup` dans le programme d'exemple Blink :

```
void setup()  
  
{  
  
pinMode(brocheLED, OUTPUT); // configure la broche numérique //en sortie.  
  
...}
```

La première ligne fournit des informations sur la fonction, tel que son nom, `setup`. Le texte avant et après le nom spécifie le type de valeur renvoyée par la fonction et les paramètres qu'elle utilise. Le code entre les accolades `{ }` est le corps de la fonction : le code qui définit ce que fait la fonction.

On peut appeler une fonction qui est déjà définie (soit dans le programme, soit dans une librairie, soit une fonction qui est partie intégrante du langage Arduino). Par exemple, la ligne :

```
pinMode (brocheLED, OUTPUT);
```

//qui appelle la fonction *pinMode*, lui faisant passer 2 paramètres : la variable *brocheLED* et la constante prédéfinie *OUTPUT*. Ces paramètres sont utilisés par la fonction *pinMode* pour savoir quelle broche et quel mode de fonctionnement utiliser.

II.14.3.1. Quelques fonctions très utilisées :

Il y a deux fonctions spéciales, obligatoires, qui sont partie intégrante de tout programme en langage Arduino : les fonctions *setup()* et *loop()*. La fonction *setup* doit être appelée en premier, lorsque le programme commence. Cette fonction n'est exécutée qu'une seule fois au démarrage du programme. C'est le lieu privilégié pour réaliser les tâches d'initialisation des broches numériques ou des bibliothèques.

La fonction *loop()* est appelée ensuite et est exécutée en boucle sans fin de façon indéfinie, et le code qu'elle contient est exécuté à chaque passage. C'est à ce niveau que sera écrit le cœur de programme.

A noter qu'on est obligé d'inclure ces deux fonctions dans son programme, même si elles ne font rien.

II.14.3.1.1. La fonction *pinMode()* :

La fonction *pinMode()* configure une broche numérique de la carte Arduino soit en entrée soit en sortie. Pour l'utiliser, on lui passe le numéro de la broche et la constante prédéfinie *INPUT* ou *OUTPUT*. Quand elle est configurée en entrée, une broche peut détecter l'état d'un capteur tel qu'un bouton poussoir. En sortie, une broche peut commander par exemple une LED.

II.14.3.1.2. La fonction *digitalWrite()* :

La fonction *digitalWrite()* met une valeur (0 ou 1) sur une broche. Par exemple, la ligne

```
digitalWrite(ledPin, HIGH) ;
```

//qui met la broche *brocheLED* (broche 13) à *HIGH* ou niveau

HAUT à 5 volts. Ou *LOW* (niveau BAS) sur la broche la connecte à la masse ou 0V.

II.14.3.1.3. La fonction delay() :

La fonction delay() oblige la carte Arduino à attendre (= ne rien faire) pendant un nombre spécifié de millisecondes avant de continuer à la lignes suivante. Il y a 1000 millisecondes dans 1 seconde, ainsi la ligne :

```
delay(1000); //qui entraîne une pause de 1 seconde.
```

II.15. Matériels et logiciels nécessaires et préparation à la mise à jour de contrôleur de vitesse à base d'atmega8A.

Nous avons besoin de :

- un programmeur d'AVR "ISP" comme USBASP, un jtag, ou d'un Arduino modifié en AVR
- un bon fer à souder 20 à 30W maxi.
- de l'étain d'électronicien de qualité.
- un peu de fils (nappe de disque dur c'est parfait).
- barrette "tulipe" sécable.
- un peu de gaine thermo rétractable de petit diamètre.
- Raccorder son programmeur d'AVR, en respectant l'ordre des fils donné par le schéma.

II.14.1. Commençant par préparer le contrôleur de vitesse:

- ❖ -Connecter le programmeur AVR ISP ou jtag:



Figure II.16: Image de programmeur USBASP .

- ❖ Raccorder donc le programmeur en suivant ce câblage et celui donné pour le contrôleur.

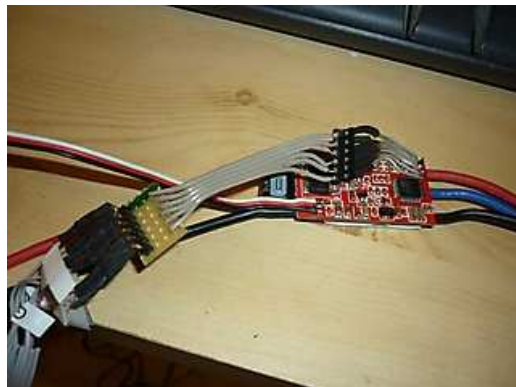


Figure II.17: Un contrôleur de vitesse raccordé au programmeur via l'interface ISP.

- ❖ Maintenant la même méthode, mais on soude et on dessoude le connecteur après programmation.
- ❖ Voilà le montage est presque prêt pour reprogrammer le contrôleur... *On passe maintenant à la partie software.*

II.15. Chargement du programme:

II.15.1. Avrdude:

Est un logiciel de programmation de microcontrôleurs de la famille AVR d'ATMEL. Il permet de programmer la mémoire flash et eeprom via le protocole série. Il permet entre autre de modifier les fusibles (fuse bits –pour la configuration des différentes options de contrôleur tel que le type de quartz ou l'activation ou pas de verrouillage de microcontrôleur ou pas ..Etc.). Le point fort de logiciel AVRDUDE est le fait que ce dernier peut supporter des modes d'instructions et des spécificités liées aux matériels qui ne sont pas implémentées dans ce dernier à l'origine.

Pour programmer avec AVRDUDE :

- Nous devons avoir sur notre disque dur le fichier « *cev1.hex* », (celui correspondant à notre contrôleur).
- Maintenant tout est prêt, il ne reste plus qu'à programmer !

- Pour ce faire, on ouvre une fenêtre de commande dos (cmd) ou Shell sous linux.

Nous tapons :

avrdude -c usbasp -b 19200 -p ?

vous faite apparaitre un listing avec les puces prise en charge par notre programmeur...

dans notre cas c'est une « Atmega8 » dont le code sur avrdude « *m8* ».

nous allons donc tapez ensuite: avec mon firmware « *cev1.hex* ».

avrdude -c usbasp -b 19200 -p m8 -U flash:w:cev1.hex .

II.15.1.1. La programmation démarre:

```

C:\>avrdude -p m168 -c avrisp -P com4 -b 19200 -F -U flash:w:Blink.hex
avrdude: AVR device initialized and ready to accept instructions
Reading ! ##### ! 100% 0.05s
avrdude: Device signature = 0x000000
avrdude: Yikes! Invalid device signature.
avrdude: Expected signature for ATMEGA168 is 1E 94 06
avrdude: NOTE: FLASH memory has been specified, an erase cycle will be performed
To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "Blink.hex"
avrdude: input file Blink.hex auto detected as Intel Hex
avrdude: writing flash (1108 bytes):
Writing ! ##### ! 100% 0.98s
avrdude: 1108 bytes of flash written
avrdude: verifying flash memory against Blink.hex:
avrdude: load data flash data from input file Blink.hex:
avrdude: input file Blink.hex auto detected as Intel Hex
avrdude: input file Blink.hex contains 1108 bytes
avrdude: reading on-chip flash data:
Reading ! ##### ! 100% 0.84s
avrdude: verifying ...
avrdude: 1108 bytes of flash verified
avrdude: safemode: Fuses OK
avrdude done. Thank you.
C:\>

```

Figure II.18: Fenêtre principale de type MSDOS de saisie et d'exécution des opérations par avrdude.

Si la programmation ne démarre pas, nous avons certainement une erreur dans le câblage et dans l'ordre des fils. Sinon cela proviens d'un mauvais paramétrage de la ligne de commande avrdude...

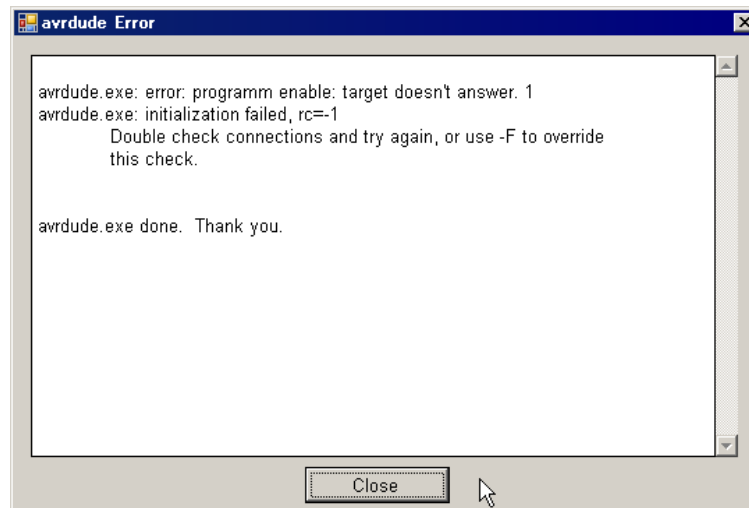


Figure II.18: Fenêtre signalant le non-établissement de la communication avec le matériel.

- A la fin de la programmation, avrdude fait un test afin de vérifier que la puce soit bien programmée.
- C'est très important que AVRDUDE nous indique "**FLASH VERIFIED**".

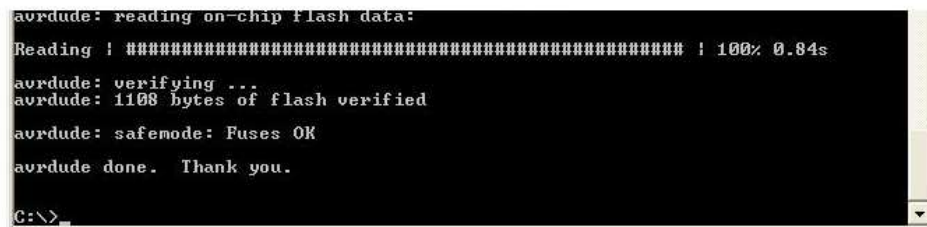


Figure II.19: Fenêtre d'affichage des opérations effectuées par avrdude.

II.16. Conclusion :

Dans ce chapitre l'étude s'est porté sur le fonctionnement et la préparation des deux types de microcontrôleurs AVR employées servant de processeur de contrôleur de vol pour le premier et de contrôleur de vitesse pour l'autre. Aussi l'intérêt s'est porté sur les deux méthodes de programmation des AVR qui sont celle utilisant l'ISP et l'autre utilisant le bootloader et la communication série pour l'Arduino.

Chapitre III:

Les capteurs de navigation aérienne.

III.1. Introduction :

Dans cette section nous présentons un tour d'horizon des capteurs qui sont couramment utilisés pour l'estimation de la localisation et de l'orientation des véhicules aériens.

III.2. Le gyroscope :

Est un dispositif de mesure ou de maintenir l'orientation angulaire basé sur les principes de moment angulaire (principe de la toupie). Mécaniquement, un gyroscope est une roue ou disque où l'axe est libre d'accuser n'importe quelle orientation sur les trois axes de l'espace. Bien que cette orientation ne reste pas fixe, il change en réponse à un moment de torsion extérieur .



Figure III.1: Le dispositif gyroscopique mécanique

L'orientation du gyro reste quasiment fixe indépendamment le mouvement de la plateforme qu'il le support ce qui minimise le moment de torsion externe.

Les gyroscopes basés sur les autres technologies existent aussi, comme les gyros électroniques de type piézo en MEMS ; Lasers ou a fibres optiques. La précision et le poids des gyromètres varient en fonction du prix.

Dans notre application le gyro est intercalé entre le récepteur et le circuit de commande du quadri-rotor, il pourra contrer une variation brusque et non souhaitée de l'angle de roulis, de lacet ou de tangage, suivant la voie à laquelle le gyroscope est associé.

Nous utiliserons un ITG2300 montré dans la Figure **II.2** .Ce type de gyroscopes offrent la possibilité de communiquer via le bus I2C lui offrant la possibilité de

l'interfacer sur le même bus que les autres capteurs et travaillant à une fréquence allant jusqu'à 400kbit/s.



Figure II.2: Image de capteur gyroscopique ITG-3200.

III.3.Accéléromètre triaxial BMA180:

Il existe plusieurs types, comme par exemple: ITG3200, ADXL345, ADXL335, HMC5883L, MPU6050, BMP085, MMA7361, mais celui qui sera employé sera le BMA180 .ce module encore plus performant est un capteur d'accélération triaxiale numérique ultra-performant, destiné aux applications grand public à faible consommation. Le BMA180 prend en charge la mesure précise d'accélération sur trois axes perpendiculaires, détectant ainsi l'inclinaison, les mouvements, les chocs et les vibrations dans les téléphones portables, les dispositifs portables, les périphériques informatiques, les interfaces homme-machine, les systèmes de réalité virtuelle et les périphériques de jeu.

Le BMA180 représente une nouvelle génération de capteurs d'accélération numériques, grâce à des performances et à un ensemble de fonctionnalités uniques dans un boîtier LGA standard de 3 mm x 3 mm x 0,9 mm. Hautement configurable, le capteur BMA180 offre aux concepteurs une totale flexibilité d'intégration à de nombreux systèmes. Toutes ses fonctionnalités peuvent être configurées au moyen d'un logiciel, via l'interface numérique. L'utilisateur peut choisir entre un mode d'interface I²C ou SPI (4 fils).

III.3.1. Applications :

- L'ensemble des paramètres et des fonctionnalités sont programmables par l'utilisateur
- 7 plages de g (gravité) programmables par l'utilisateur et 10 paramètres de largeur de bande sélectionnables

- Faible consommation
- Mode de réactivation automatique à ultra-faible
- Consommation Moteur d'interruption programmable par l'utilisateur
- Régulation de décalage 0g et réétalonnage de décalage sur le terrain à ultra-haute précision
- Interface SPI (4 fils)/I²C



Figure III.3: Image montrant le module BMA180.

III.3.2. Brochage de BMA180 :

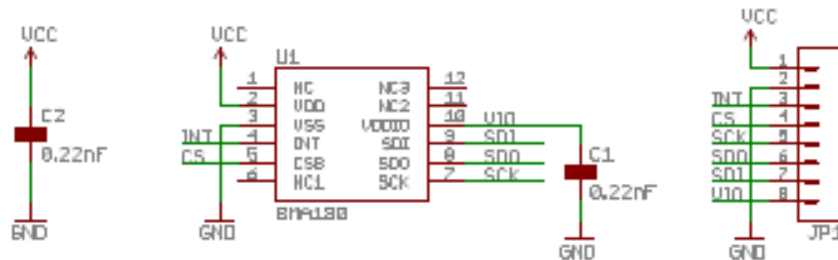


Figure III.4: Schéma de brochage de BMA180.

III.3.3. Description des broches :

- **VDD:** capteur d'alimentation.
- **GND:** masse.
- **NT:** Sortie numérique. Indique si un événement d'interruption s'est produit en Logique positive.

- **CS:** Entrée numérique .Chip select. Correspond à la broche CSB. Lorsque cette broche est à 0V, la fonctionnalité SPI est activée.
- **SCK:** entrée numérique. Le signal d'entrée d'horloge.
- **SDO:** entrée / sortie numérique. Sortie SPI ou I2C configuration d'adresse.
- **SDI** d'entrée / sortie numérique. D'entrée SPI ou I2C données
- **VIO:** puissance d'entrée numérique. Correspond à la broche VDDIO. L'interface série avec le microcontrôleur peut gérer des tensions différentes que l'alimentation du capteur. Par exemple, le capteur pourrait être amené à 2.0V et l'interface série peut fonctionner à 3.6V.

III.3.4. Les caractéristiques de BMA 180 :

Accéléromètre	Niveau Max mesuré	Interface	Nombre d'axes	Bande passante Max	Puissance	Autres options
BMA180	+ -1.15, 2, 3, 4, 8,16g	I2C et SPI	3	2400Hz	2-3.6, 650-975uA	<i>Filtres numériques programmables, plusieurs modes d'interruptions, alimentation programmable, détecteur de température ...</i>

III.4. Baromètre :

Il est possible de piloter un drone manuellement sans un baromètre, mais il ne sera pas possible de travailler en mode stationnaire automatique. En effet le baromètre qui est à l'origine un capteur de pression atmosphérique, le system de contrôle de vol exploite cette caractéristique pour contrôler l'altitude de l'appareil. Un baromètre donc est nécessaire pour le maintien d'altitude. le plus recommandée est Le BMP085.

Enfin d'autres capteurs peuvent être utilisés pour rendre l'engin encore plus autonome, stable et précis.

Parmi ces capteurs on distingue:

III.5. Boussole électronique (Compass) :

Qui permet au drone de s'orienter par rapport au nord magnétique. En fait, il est possible de piloter un drone manuellement sans boussole. Mais pour un contrôle total à grande distance, l'utilisation d'un compas associé à la navigation GPS est indispensable. Parmi les produits les plus récents, il y a le HMC6352.

III.6. Le GPS (*Global Positioning System*):

Il permet de connaître la position du drone, en trois dimensions (x, y, z). Cette information est envoyée par des satellites en orbite autour de la terre. La sélection du GPS embarqué dans le drone, doit se faire sur sept caractéristiques principales : la consommation électrique, la taille, le poids, la précision globale, la précision, le nombre de ports série et la vitesse de transmission.

III.7. Télémètre :

Les télémètres à ultrasons fonctionnent en mesurant le temps de retour d'une onde sonore inaudible émise par le capteur. La vitesse du son dans l'air étant à peu près stable, on en déduit la distance de l'obstacle.

Le microcontrôleur est une partie très importante dans la réalisation de tous les systèmes embarqués. En effet, c'est lui qui regroupe, extrait et calcule toutes les informations dont dispose le véhicule aérien.

III.8. Camera :

La caméra vidéo est un des capteurs extéroceptifs les plus performants, qui fournit une information particulièrement riche sur l'environnement. Elle permet de transmettre les images vues par le drone, vers l'opérateur au sol. Dans certains cas, on traite les informations transmises par la caméra sur un PC au sol via une carte d'acquisition d'images pour déterminer la position de l'engin, pour faire du suivi de trajectoire, ou bien pour la détection d'obstacles.

III.9. Conclusion :

L'un des objectifs essentiels lors de la réalisation en cours est la stabilité en vole de l'appareil. Ainsi il est évident que celle-ci repose essentiellement sur les détecteurs de navigation associés à ce système. En effet un mécanisme d'asservissement ne changera pas grand-chose si les détecteurs fournissent des valeurs très approximatives. Heureusement les performances de ces détecteurs évoluent rapidement avec les progrès technologiques, et de multitude de produit nouveaux envahis le marché avec des précisions de plus en plus importantes.

Chapitre IV:

**Systeme de propulsion à base de
moteurs électriques sans balais.**

IV.1. Introduction :

La première de la chose auxquelles en réfléchira lorsqu'on s'apprête à construire un drone est le type propulsion à envisager. En effet celui-ci dépend de plusieurs paramètres tel que la structure physique de celui-ci, la nature de mouvements qu'il devrait effectuer, les vitesses espérées et suivant les technologies et le matériel disponible. Ainsi dans ce contexte les progrès récemment réalisés dans les domaines de l'Electronique de Puissance et de la Commande numérique ont permis depuis peu l'essor des contrôleurs de vitesse pour les moteurs à courant continu en augmentant leurs performances et les rendant incontournables dans les domaines où l'on besoins d'un très grandes vitesses associé a un couple moteur élevé .

IV.2. Fonctionnement d'un moteur électrique :

Avant de parler des moteurs sans balais en particulier, un petit rappel du fonctionnement des moteurs électriques "traditionnels", c'est à dire à charbons, ne paraît pas superflu. Il faut savoir que, lorsqu'un courant passe dans un fil et que ce fil est soumis à un champ magnétique (aimant par exemple), ledit courant fait naître une force magnétique. Dans un moteur électrique, cette force, engendrée par le passage du courant dans les bobinages (fils émaillés du moteur) qui sont soumis au champ magnétique des aimants, est utilisée pour produire une rotation. Le moteur électrique est donc un "convertisseur" transformant l'énergie électrique en énergie mécanique... plus de la chaleur car le rendement n'est jamais de 100%. Pour créer le "champ tournant" des bobinages, il faut utiliser un système de commutation de manière à faire passer le courant dans ces bobinages dans un ordre précis et au bon moment.

Dans un moteur à charbons ou a balais (appelé brushed en anglais), cette commutation des bobinages est réalisée mécaniquement par l'intermédiaire des lamelles du collecteur situées sur l'axe moteur. Lors de la rotation, les charbons sont successivement en contact avec ces lamelles qui vont transmettre le courant continu délivré par le variateur aux bobines du moteur. Le passage du courant dans le fil des bobines fait naître une force magnétique qui repousse les aimants de même pôle et attire ceux de pôles opposés. Ce système de commutation mécanique détermine

l'architecture type d'un moteur à charbons qui comprend : un rotor interne portant les bobinages, et un stator externe où sont fixés les aimants. Cette commutation mécanique est facile à réaliser mais le frottement des charbons (ou balais) sur le collecteur entraîne une perte, un échauffement du collecteur et une usure inévitable de celui-ci. D'où un rendement relatif et la nécessité d'entretenir le moteur en vérifiant régulièrement l'état des charbons et du collecteur.

IV.3. Les moteurs sans balais MSB:

Les moteurs à courant alternatifs sont plus robustes que les moteurs à courant continu, plus faciles à entretenir, possèdent moins d'inertie (pas d'enroulements au rotor pour les sans balais et les moteurs asynchrones à cage, pas de collecteur) et moins chers à construire, cependant ils sont plus difficiles à piloter. Il existe 2 sortes de moteur MSB qui ne se distinguent, essentiellement, que par leur capteurs de position et par leur électronique de commande intégrée (et non pas par leur principe de fonctionnement qui sont identiques) : le moteur synchrone AC MSB qui possède des f.e.m à vide sinusoïdales, un codeur de position précis et est piloté en MLI par des courants sinusoïdaux (en régime permanent) plus cher (à cause du capteur de position) mais plus performant (ondulations de couple inexistantes), il est plutôt utilisé pour des puissances allant jusqu'à 400 W.

IV.3.1. Les moteurs DC-sans balais :

Le collecteur du moteur DC est remplacé par 3 détecteurs de présence à effet Hall « qui détectent la présence d'un champ magnétique en fournissant par ex 5 V en présence d'un pôle Nord et 0 en présence d'un pôle sud ». Ces capteurs sont disposés à 120° ou à 60° l'un de l'autre (dans ce dernier cas le capteur "du milieu" donnera tout simplement la tension complémentaire à celle qu'il aurait produite s'il avait été placé à 120° des 2 autres) :

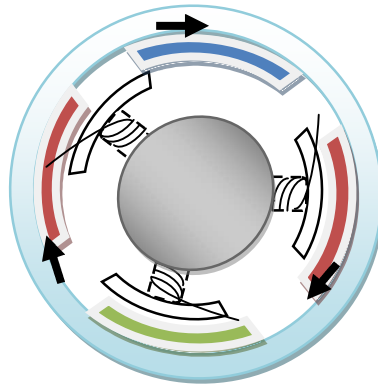


Figure IV.1: Moteur sans balais avec le rotor externe et stator au milieu.

Le moteur sans balais est un moteur synchrone « c'est-à-dire qu'il tourne à la même vitesse que le système de tensions qui l'alimente. Tant que le couple moteur est supérieur à la charge à entraîner, la rotation du rotor est synchronisée avec le champ magnétique » autopiloté à aimants permanents (absence de balais, d'où le nom). C'est un moteur avec électronique de réglage intégrée au châssis. (carte d'asservissement). Les moteurs sont les moteurs actuellement les plus performants « plus faible inertie que les MCC de même puissance ».

Le moteur DC-sans balais possède des aimants permanents qui créent des f.e.m. trapézoïdales dans les enroulements statoriques triphasés parcourus par des courants $1/3$ - $2/3$ idéalement en créneaux. le moteur est à rotor sans fer d'où une inertie très faible, parfaitement adapté aux servomécanismes.

IV.4. Comment fonctionne un moteur sans balais « brushless » :

Dans un moteur sans balais, la commutation des enroulements est faite non pas mécaniquement comme précédemment mais de manière électronique par un système complexe appelé "contrôleur". Celui-ci transforme le courant continu en courant triphasé à fréquence variable et va alimenter successivement les bobines du moteur pour créer le champ tournant et donc la rotation qui nous intéresse. On comprend aisément qu'avec ce principe d'alimentation, il est impératif que les bobines soient fixes dans le moteur, et non pas en rotation comme dans un moteur à charbons. Tous les moteurs sans balais ont donc la même architecture de construction : un stator fixe qui porte les bobines, et un rotor mobile sur lequel les aimants permanents sont collés.

Les bobinages peuvent être réalisés de manières différentes : en étoile ou en triangle (appelé également delta), mais on trouve toujours trois fils à la sortie du moteur, qui réunissent les bobinages.

La majorité de ce type de moteurs possèdent un rotor interne qui tourne très vite jusqu'à 100.000 tr/mn. Si la vitesse est là, le couple est médiocre et il est alors obligatoire de les réduire fortement de manière à pouvoir utiliser des hélices de taille adaptée.

IV.5. Inconvénients :

- Absence de balais oblige à intercaler un contrôleur entre la batterie et le moteur.
- Dispositif complexe, génère une tension triphasée synchrone avec le rotor.
- Dispositif devenu fiable mais plus onéreux qu'un variateur de moteur classique.

IV.6. Avantages de ces moteurs :

Les avantages des moteurs sans balais sont tellement nombreux qu'ils font vite oublier les inévitables inconvénients. En premier, tous les soucis liés aux frottements des charbons sur le collecteur disparaissent : plus de parasites, plus d'échauffement du collecteur et de pertes dues à l'étincelage, plus d'usure mécanique (si ce n'est celle des roulements), et pas besoin de rodage long et fastidieux non plus, le moteur étant utilisable au sortir de sa boîte. Le rendement est également bien supérieur à celui d'un moteur ferrite, de même que légèrement supérieur aux meilleurs moteurs samarium-cobalt et néodyme.

Mais l'avantage majeur et incontournable est bel et bien l'énorme gain de masse. A puissance développée équivalente, un tel moteur pèse deux à trois fois moins lourd qu'un moteur à balais, ce qui n'est pas rien ! En moyenne, le rapport poids/puissance d'un moteur sans balais est de 20 à 25 g pour 100 W développés.

En gros :

- **Bon rendement** : >70% pour les petits, >90% pour les plus gros.

- A puissance égale il tourne moins vite et donne plus de couple qu'un autre moteur électrique.
- L'hélice peut fonctionner au mieux sans réducteur.
- **Absence de balais** :
- pas d'étincelles, donc moins de parasites électriques
- souvent les balais sont les éléments qui se détériorent le plus vite.

IV.7. Choix d'un moteur sans balais :

Plusieurs chiffres sont à prendre en compte :

IV.7.1. Les kv :

Il nous indique le nombre de tours/min par volt. A vide, un moteur de 5 800 kv sera donc capable de tourner à 5 800 trs/min pour 1 volt, soit 42 920 tours avec les 7,4 volts d'une batterie Li-Po. Donc plus le nombre de kv est grand, plus le moteur ira vite.

IV.7.2. Le couple :

Il est donné par le nombre de tours (T). Plus T est grand, plus le couple ne sera important. Malheureusement, la fabrication des moteurs sans balais implique que la vitesse sera également réduite. Il n'y a donc pas de miracle, et les kv sont étroitement liés à T.

Donc, pour résumer, un fort kv sera lié à un petit nombre T, et le moteur sera fait pour la vitesse de pointe. Un faible kv sera lié à un grand nombre T et le moteur sera fait pour arracher le sol à l'accélération.

Dans la pratique, notre drone est suffisamment léger pour que la perte de couple liée à l'augmentation des kv ne soit pas aussi pénalisante que ça. On choisit donc un moteur à fort kv (donc à faible T) pour avoir un vol plus performante. Cependant, "sans maîtrise la puissance "une grande vitesse produira pas grand chose.

Il y a bien entendu d'autres facteurs liés aux choix technologiques et à la qualité de fabrication qui vont jouer sur la vitesse et le couple d'un moteur, et c'est ce qui va faire la différence d'une marque à l'autre ; comme par exemple le nombre de tours des

stators, et le type d'aimants employés, existence de frottements, qualité de roulements...etc.

Enfin, on peut aussi jouer sur le rapport de vitesse (la taille des pignons s'il ya lieu) pour donner plus ou moins d'accélération ou de vitesse de pointe. Et ce point est au moins aussi important que le choix du moteur.

Une dernière caractéristique très importante, ce sont les watts qui donneront la consommation maximale du moteur. Plus le moteur a un fort kv, et donc un faible T, plus il sera gourmand en watts. Ce chiffre est important pour le choix du contrôleur qui devra être capable de délivrer cette puissance.

IV.8. Moteur sans balais à rotor externe « outrunner »:

On appelle « outrunner » les moteurs sans balais dont le rotor est autour du stator. Cette configuration est intéressante en termes de couple moteur, car les aimants sont disposés sur un diamètre important, ce qui crée un bras de levier très intéressant. De plus, cette disposition permet de placer facilement plusieurs séries d'aimants (jusqu'à 32 pôles sur certains moteurs) et de bobines. Les bobines sont toujours câblées par groupes de 3, et les aimants sont soit collés par groupes de 2, soit constitués d'une partie magnétique comprenant plusieurs pôles. Comme pour un moteur pas à pas, les moteurs sans balais a rotor externe comprenant plus de 3 bobines et 2 pôles ne font qu'une fraction de tour lorsque le champ a tourné de 180°. Leur fréquence de rotation est donc plus faible mais le couple très élevé. Ces moteurs sont souvent utilisés dans des applications qui nécessitent un fort couple comme dans le cas de drone, car ils peuvent être reliés à la charge sans nécessiter de dispositifs de réducteurs de vitesse. Leur coefficient Kv est relativement faible par rapport aux autres types de moteurs sans balais. Les principales applications des moteurs a rotor externe sont les suivantes : ventilateurs, moteurs de disques durs, Cd-rom, moteurs de vélos électriques, les voitures et avions électriques...etc.

Figure IV.2: Vue intérieur d'un moteur électrique sans balais.

IV.8.1. Comment se présente le moteur ?

Le stator, matériau à haute perméabilité magnétique, souvent formé de 9 ou 12 dents porte des bobines. Les aimants sont collés sur la cloche externe pour présenter une alternance nord sud forment le rotor. Souvent 12 aimants pour 9 dents, et 14 aimants pour 12 dents. Les enroulements de fil de cuivre peuvent être sur chaque dent, ou sur une dent sur deux.

Les fils de cuivres sur les dents sont reliés en série pour ne former que 3 bobines au final.

Figure IV.3: le moteur en vue éclatée.

La connexion des bobines entre elles forme une disposition en étoile ou en triangle.

- Dans la disposition étoile, les bobines en série présentent une impédance plus grande.

- Dans la disposition triangle, les bobines en parallèle présentent une impédance plus petite.

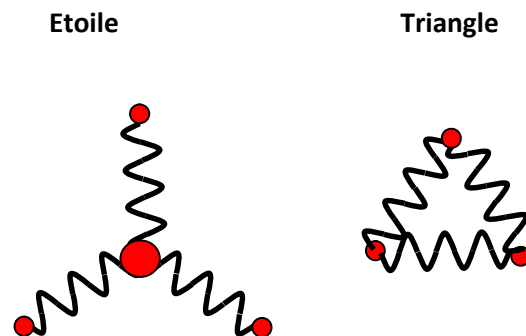


Figure IV.4: Représentation des enroulements en étoile et triangle.

On obtient la même puissance dans :

- la disposition étoile, avec plus de tension pour moins de courant.
- la disposition triangle, avec moins de tension pour plus de courant.

IV.9. Le contrôleur de vitesse (électronique speed contrôler ou ESC) :

IV.9.1. Fonctionnement :

Élément primordial pour les performances du moteur, le contrôleur est un élément électronique complexe, généralement protégé par une simple gaine thermo rétractable. C'est lui qui va gérer les commutations des bobinages pour créer la rotation du moteur. Ce module électronique va ainsi générer six commutations successives pour chaque cycle. Chaque commutation est réalisée par un groupe de transistors montés en parallèle qui vont agir comme un interrupteur. Ce contrôleur est constitué d'un ensemble de systèmes électroniques ayant chacun un rôle primordial :

- un système qui détecte le moment précis où le courant passe dans les bobines.
- un système pour mesurer la valeur du courant dans les bobines.
- un système qui commande le fonctionnement des transistors.
- un système qui gère le décalage de phase (timing).

- un système qui gère la durée d'ouverture en fonction du signal de la radio.
- Quels que soient le type de moteur et de bobinage (étoile ou triangle), ou la marque et le type de
- contrôleur, la "séquence" de commutation est toujours la même.

IV.9.2. Les réglages nécessaires:

Ces contrôleurs offrent de plus en plus de possibilités de réglages. Les premiers contrôleurs avaient besoin de capteurs pour déterminer la position du rotor et envoyer les commutations au bon moment.

En plus des trois fils, les moteurs possédaient donc une nappe équipée d'un connecteur que l'on devait brancher au contrôleur. Depuis, les contrôleurs sont devenus "sensorless", c'est-à-dire sans capteurs. Les contrôleurs récents offrent des possibilités de réglages très étendues, que ce soit au niveau de l'adaptation au type de moteur utilisé, à celui des accus utilisés, ou à bien d'autres paramètres encore. Pour les réglages concernant le moteur, on peut choisir entre plusieurs valeurs de timing, plusieurs valeurs de fréquence, choisir la présence et la force du freinage, ainsi que le démarrage du moteur. On peut également programmer une régulation automatique de la vitesse (option très utile pour notre utilisation, afin de réguler automatiquement le régime du rotor). Pour l'accumulateur, il est possible de régler le seuil de coupure du contrôleur en fonction du type d'éléments utilisés (Ni-Cd, Ni-MH, Li-Po) afin de ne pas endommager ceux-ci. Enfin, la plupart des contrôleurs intègrent également une protection contre la surchauffe ou une consommation trop importante.

IV.9.4. Contrôleur de commerce :

Les contrôleurs électroniques de vitesse du commerce ne sont pas spécialement dédiés aux drones, la conception de ces contrôleurs est faite pour permettre une accélération et décélération douce des moteurs.

Figure IV.5: Vue intérieur de contrôleur de vitesse commercial.

De plus les contrôleurs bon marché sont conçus autour d'une rampe de résolution de « 25 à 40 pas » suivant le modèle (25 à 40 vitesses possibles de moteur). Sur certain modèle la courbe d'accélération est exponentielle (accélération douce, et maîtrise de la vitesse, puis d'un coup, une grosse montée en régime produisant ainsi comme un « flou » dans la gestion de la vitesse).

Sur un drone, il nous faut justement l'inverse : des accélérations franches et des changements de régimes rapides, une courbe d'accélération linéaire ; la possibilité d'avoir un grand nombre de vitesses possibles avec le moteur, et une précision accrue sur la retranscription d'un ordre du système de réception.

Bien que les ESC du commerce soient largement suffisants pour faire voler un engin de ce type, il existe la possibilité d'améliorer leurs fonctionnements en modifiant leur programme interne.

On aboutit ainsi à une :

- Meilleure vitesse de réaction
- Accélération/ décélération rapide et précise
- Meilleure résolution (jusqu'à 200 pas !)
- Réponse linéaire aux commandes.

Bien que très bénéfique à la performance, cette modification n'en demeure pas "magique", il faudra quand même régler les PID et régler sa machine.

IV.9.5. Choix de contrôleur de vitesse:

Le choix d'un contrôleur est plus simple que celui d'un moteur, même si l'offre est de plus en plus fournie. Le premier critère de sélection est tout simplement celui du nombre d'éléments que nous comptons utiliser. Plus le couple tension/intensité est important, plus il faut de transistors et plus le contrôleur est onéreux. Les fabricants donnent la consommation admise en continu et celle en pointe pendant quelques secondes. Par exemple, un moteur est donné pour une alimentation en 6 à 18 éléments, et une consommation de 40 A en continu et 60 A pendant 15 secondes. On doit prendre en considération les performances données en continu, car la plus part de temps on travail dans ce régime. Si la consommation au sol est de 40 A, on choisira alors un contrôleur capable d'encaisser 50 ou 60 A.

Figure IV.6: Image d'un contrôleur de vitesse commercial.

IV.9.6. Les fonctions du contrôleur :

Equipé d'un micro contrôleur il gère 3 fonctions :

A-L'entree des signaux de commandes :

Il reçoit l'entrée du récepteur, produit un découpage de la tension de batterie pour modifier la tension moyenne appliquée au moteur. C'est la fonction classiquement gérée par le variateur.

b-La commutation des tensions sur les bornes A1, B1, C1 du moteur :

Il possède 6 interrupteurs, qui en combinant leurs états connectent indépendamment chaque borne à la batterie, ou la laisse "en l'air".

Par exemple :

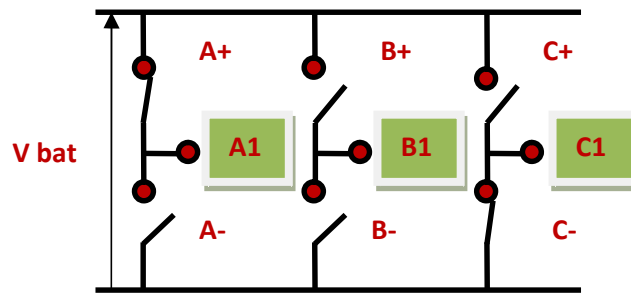


Figure IV.7: Canalisation des courants par les différents mosfets.

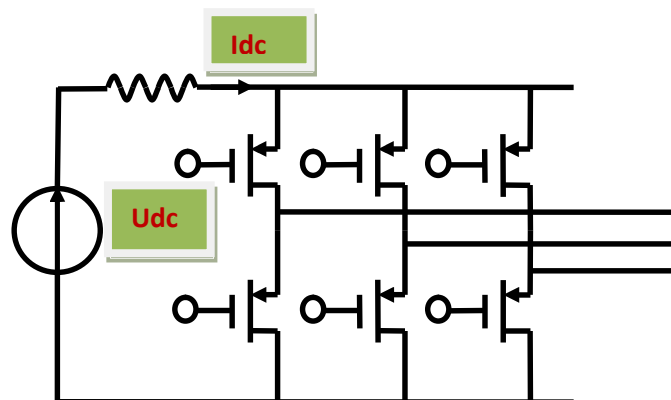


Figure IV.8: Etage de puissance de contrôleur de vitesse constituée de 6 Mosfets.

c-Commande des interrupteurs :

Les interrupteurs sont des transistors MOS pilotés par le micro contrôleur. Leurs d'utilisation détermine la tension maximale d'utilisation du contrôleur : 10volts, 20 volts, ou 30 volts, ou 50 volts...etc. Leur courant maximum (de l'ordre de 10A) détermine le courant d'utilisation maximum du contrôleur. Pour l'augmenter ; chaque interrupteur est formé de plusieurs MOS en parallèle. On peut trouver $6 \times 12 = 72$ MOS dans un contrôleur de 100A.

Figure IV.9: Exemple de contrôleur de vitesse commerciale à deux rangées de mosfets PMOS et NMOS (face inférieure de la carte électronique).

d- Le contrôleur mesure la tension induite dans la bobine laissée en circuit ouvert :

Il en déduit la position des aimants du rotor pour produire la commutation synchrone qui génère le courant triphasé que voit le moteur.

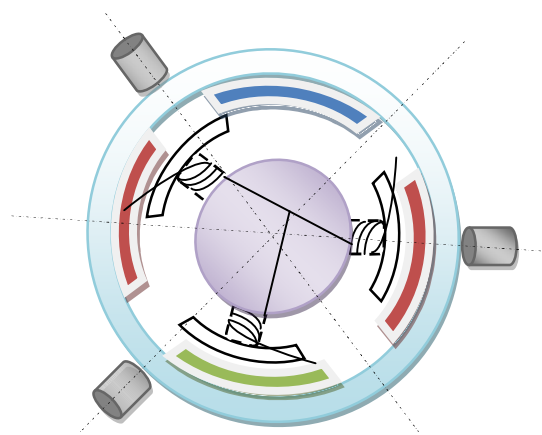


Figure IV.10: Disposition des capteurs à « effet Hall » de position du rotor.

IV.10. Logique de commande des interrupteurs :

Pour imiter les 3 lames de collecteur des moteurs triphasés ordinaires, on utilise pour le moteur à 6 interrupteurs (mosfets ou IGBT..) qui seront pilotés par les 3 détecteurs de présence du champ rotorique. Ces 3 capteurs a, b et c sont les analogues des 3 lames du moteur DC. Ces détecteurs génèrent, en fonction de la position de l'aimant rotorique, une tension qui, grâce à une logique de décodage, commande les interrupteurs qui dirigent les courants dans chaque enroulement. Pour obtenir un sans balais on bloque le rotor du moteur DC 3 lames précédent et on remplace le système

balais collecteur par une alimentation fixe, commandée par des interrupteurs. Lorsqu'on alimentera les bobines de façon adéquate, l'aimantation qu'elles produiront aura pour résultat l'attraction des bobines par l'aimant périphérique. Comme les bobines sont à présent fixées sur le châssis, on ne verra rien de particulier...sauf si on laisse libre le "stator" en le désolidarisant du châssis : dans ce cas il se mettra à tourner en sens inverse de ce que faisait le moteur DC, en réaction à ce que produisent les bobines, le stator du moteur DC devient rotor (extérieur) et on a fabriqué un moteur sans balais.

Sortie des détecteurs

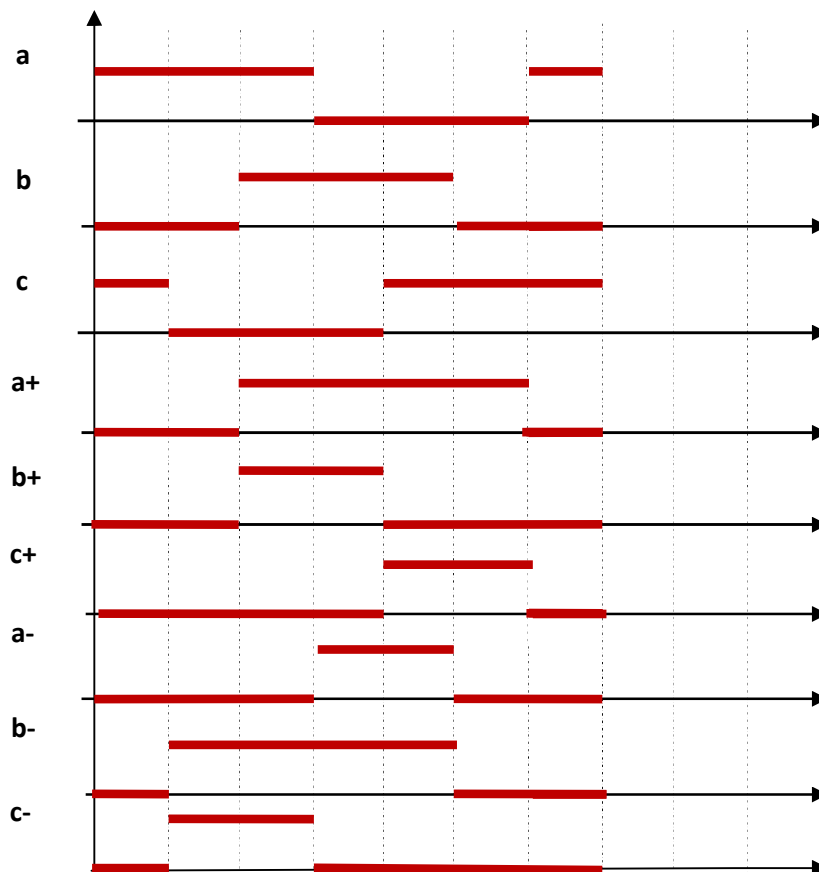


Figure IV.11: Signaux de commande en entrées et sorties de contrôleur de vitesse et de moteur.

IV.11. Régulation de vitesse d'un moteur sans balais :

Globalement, la vitesse maximale d'un moteur sans balais est liée à sa tension d'alimentation, ou du rapport cyclique dans le cas d'une commande en PWM, par le coefficient K_v . La valeur de ce coefficient dépend de la construction du moteur (nombre de bobines, de pôles, d'encoches, type de fil pour les bobines,...). Contrairement à un moteur synchrone triphasé ou on ajuste la fréquence pour obtenir la vitesse souhaitée, ici c'est la vitesse de ce moteur qui va indiquer au contrôleur à quelle fréquence il doit assurer la commutation. Pour réguler la vitesse d'un moteur sans balais il faut donc faire varier la tension d'alimentation de chaque bobinage, tout en maintenant une fréquence de commutation adaptée à la fréquence de rotation mesurée du moteur. En pratique, les contrôleurs les plus performants de ce type peuvent intégrer les deux fonctions : commutation des bobines en fonction des données des capteurs à effet hall, et régulation de la vitesse en PWM sur l'alimentation de chaque bobine.

IV.12. Modes de fonctionnement :

IV.12.1. Le mode pas à pas :

- On applique une tension entre 2 pôles des bobines, le troisième est "en l'air". On attend que le rotor vienne au repos, dite position d'équilibre.
- On commute la tension d'une des bornes vers la borne qui était "en l'air", changeant la polarité magnétique d'une dent du stator, et on attend que le rotor prenne sa nouvelle position d'équilibre. Le moteur a tourné d'un pas et ainsi de suite, en commutant convenablement pour que la suite de pas engendre une rotation dans le même sens.
- Les dessins suivants illustrent la suite des pas en montrant par des flèches les forces d'attraction et de répulsion qui apparaissent au moment de la commutation pour faire tourner le rotor. En bleu le pôle NORD des bobines, en rouge le pôle SUD.

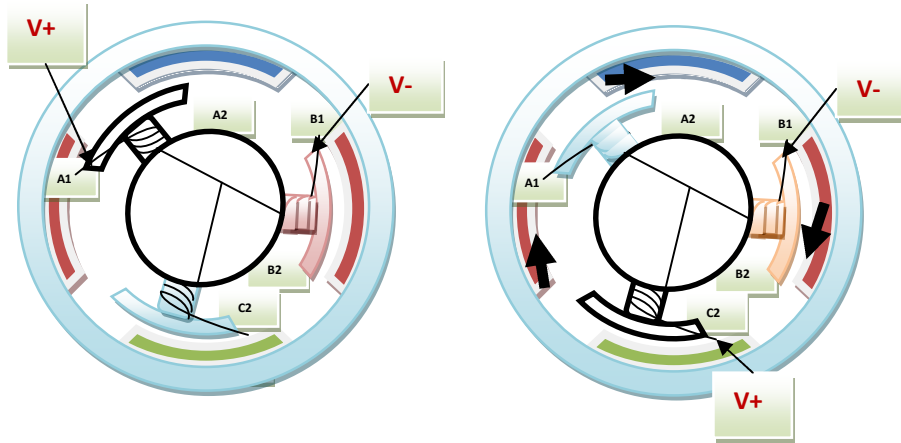


Figure IV.12: Equilibre 1 à gauche et commutation 1 à droite.

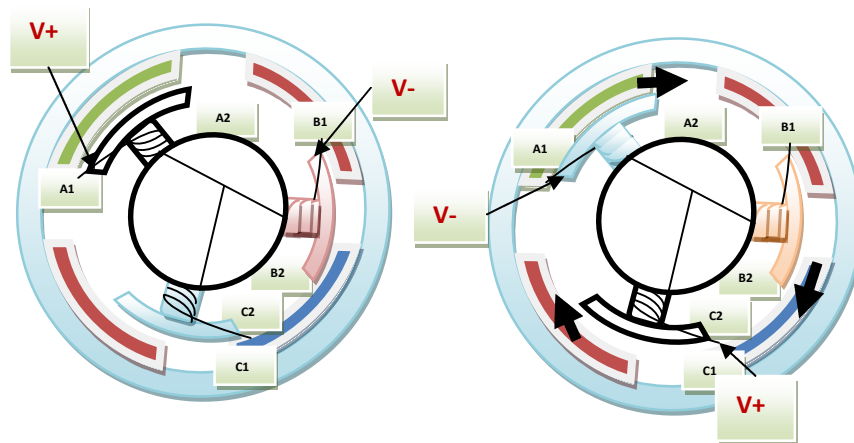


Figure IV.13: Equilibre 2 à gauche et commutation 2 à droite.

Le fait qu'il faille deux cycles de champ magnétique du stator pour obtenir un tour du rotor est parfois appelé "Effet réducteur". Pour un moteur à 12 dents et 14 aimants l'effet réducteur est de « 7 : 1 ». La commutation électrique a effectué un cycle : elle est dans la même situation que pour le dessin de l' "Equilibre 1". Pour un cycle électrique, soit un cycle de champ magnétique du stator, on a obtenu un demi-tour du rotor.

IV.12.1.1. Grand inconvénient du mode "pas à pas" :

- le couple moyen du moteur est faible parce qu'à l'approche de chaque équilibre ce couple s'affaiblit rapidement pour devenir nul à l'équilibre.

- le moteur ne peut pas tourner vite parce qu'il faut assurer un temps minimum entre chaque commutation pour que le rotor finisse son mouvement. Puisque la puissance mécanique est le produit du couple par la vitesse de rotation, la puissance obtenue est médiocre. Cependant c'est la meilleure de démarrer le moteur.

IV.12.2. Second mode « Détection de position du rotor suivie de commutation » :

La figure suivante représente un circuit de relevé de tension sur la borne C1 "en l'air". Bien sûr, ce n'est pas la seule méthode possible. Des circuits analogues se trouvent aussi sur les autres bornes pour le relevé de tension quand elles sont "en l'air". Pour faire une figure plus simple, ils ne sont pas représentés. Les tensions e_1 , e_2 , e_3 , sont induites dans les bobines par le passage des aimants.

La symétrie de construction impose que ces tensions induites sont identiques mais déphasée l'une de l'autre de plus ou moins 120° .

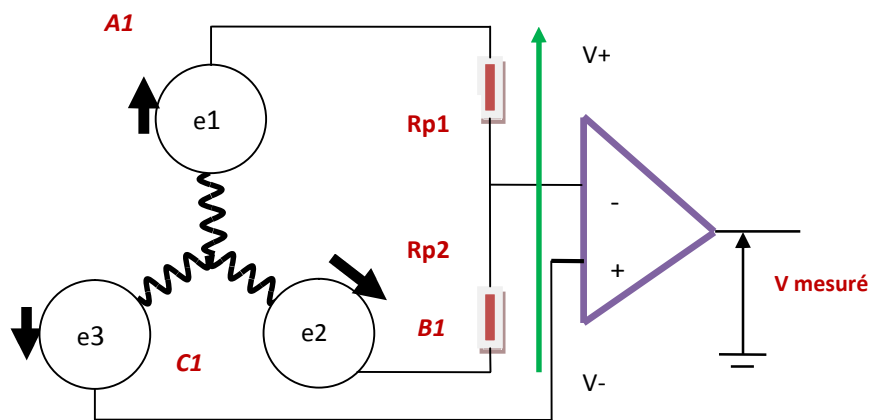


Figure IV.14: Circuit de détection de position de rotor par comparaison de niveaux d'amplitude.

Lorsque A1 et B1 ne sont pas connectés ou lorsque qu'ils sont connecté à la tension d'alimentation, par un développement trigonométrique on montre que la composante fondamentale mesurée est :

$$V_{\text{mesuré}} = \frac{3 \times e3}{2}$$

Après avoir lancé le moteur en mode pas à pas, le micro contrôleur laisse les bornes des bobines en l'air, et surveille la tension qu'elles développent par "effet dynamo" du au passage des aimants, puis au "bon moment" lance un nouveau cycle de commutation plus efficace qu'on va décrire.

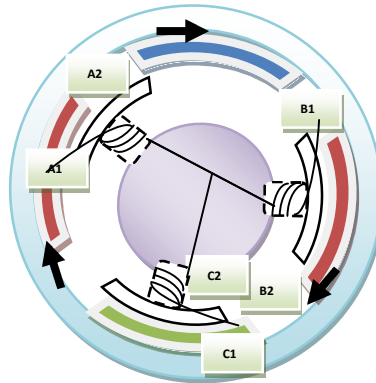


Figure IV.15: Aucun courant n'est débité et les bornes en l'air.

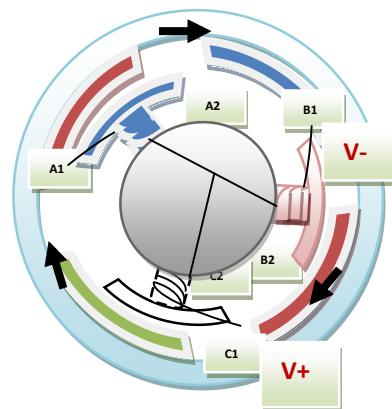


Figure IV.16: Après délais de commutation.

Quand le rotor arrive dans cette position le flux dans la bobine C1 C2 dans cette position passe par le maximum parce que la dent et l'aimant sont au plus près. La tension induite e_3 , (égale à la dérivée du flux) passe par zéro. Le contrôleur note cette position et laisse passer un délai avant d'effectuer une commutation.

Cette figure montre le moteur après ce délai au moment où s'établit la commutation. Le rouge figure le SUD des bobines et le bleu le NORD. Les flèches montrent les forces de répulsion et d'attraction qui entraînent le rotor.

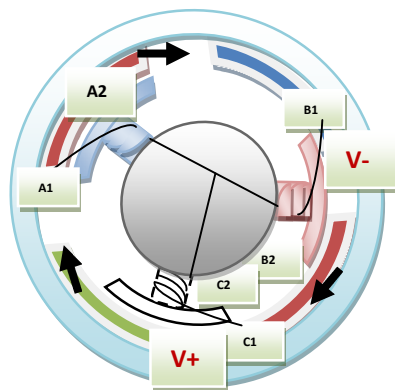


Figure IV.17: Détection de zéro sur A1.

Après cette commutation c'est la borne A1 qui est en l'air, le contrôleur surveille la tension sur cette borne et attend le passage à zéro de la tension induite. Il se produit au maximum de flux magnétique. Ceci correspond à la figure ci-contre. Les flèches représentent les forces qui entraînent le rotor.

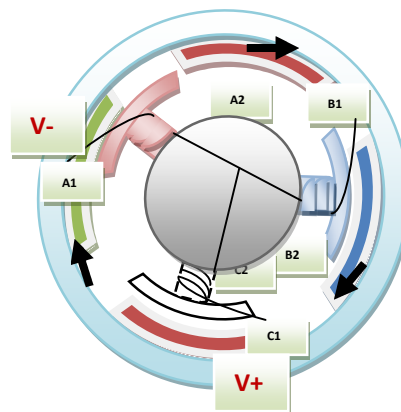


Figure IV.18: Délais de commutation.

Nouveau délai et nouvelle commutation. Et ainsi de suite....

On remarque :

- à chaque moment de l'évolution du rotor, les forces magnétiques sont toujours présentes, gage d'un couple important sur le rotor.
- les commutations suivent les positions du rotor : celui-ci peut accélérer, les commutations le suivront, et ainsi il tournera plus vite.

Fort couple x grande vitesse = grande puissance

- Le délai entre la détection du zéro de tension sur la borne "en l'air" et la commutation par le contrôleur est appelé TIMING.

On attribue la valeur de 360° à la durée du cycle et la valeur :

$$\text{Timing en degré} = \frac{360 \times t_d}{T}$$

IV.12.3. Quelques valeurs :

Timing bas ~ 4°.

Timing moyen ~ 15.

Timing fort ~ 30°.

- En rappelant une figure précédente, lorsque l'alimentation est connectée à 2 bornes, on voit 2 tensions induites qui s'opposent à l'alimentation et réduisent le courant des bobines.

Note :

- La vitesse limite du moteur est « V x Kv », car à cette vitesse la tension d'opposition égale à la tension d'alimentation, annulant le courant dans les bobines. Les forces magnétiques du stator deviennent nulles, annulant tout couple d'accélération du rotor.

- L'augmentation du timing permet d'augmenter K_v , donc la vitesse du moteur, mais elle augmente les pertes magnétiques, et le courant consommé. Il faut l'adapter en fonction de la tension de batterie et de l'hélice.

IV.13. Calcule de puissance fournit au moteur :

IV.13.1. Calcul de vitesse de rotation :

Pour déterminer les propriétés de puissance et d'entraînement de l'hélice, c'est un schéma électrique global qui sera utilisé.

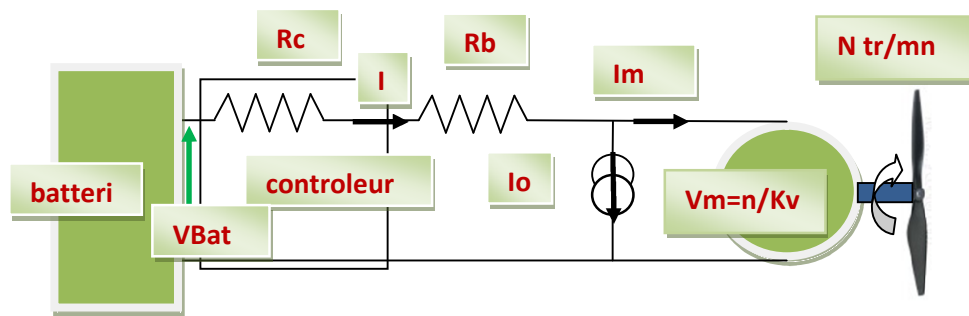


Figure IV.19: Schéma électrique générale de contrôle de moteur électrique.

- Ce schéma débute à la sortie de la batterie d'alimentation qui donne V volts.
- Ensuite le contrôleur dont les interrupteurs MOS aiguillent le courant, tout en présentant une résistance résiduelle R_c (de 30 à 50 $m\Omega$ pour les contrôleurs de 15 à 20A, de l'ordre du $m\Omega$ pour 100A). On peut inclure dans R_c les résistances de connecteurs et câble de liaisons dont la valeur peut aller de quelques $m\Omega$ à plus de la dizaine de $m\Omega$.
- Le moteur dont la résistance R_b des bobines s'oppose aussi au passage du courant, ainsi que la tension d'opposition engendrée par la vitesse de rotation (effet dynamo). Dans la suite des commutations elle se présente comme une tension quasi continue de valeur $v = \frac{n}{k_v}$, où n est la vitesse de rotation en tours/minutes.
- Enfin le courant I_o . C'est le courant consommé par le moteur lorsqu'il tourne sans hélice. Il ne fournit aucune puissance mais le courant consommé n'est pas nul, parce qu'il faut un minimum de puissance pour vaincre les frottements, les pertes

du circuit magnétique. On pose ce courant du moteur à vide égal à I_o . Avec cette représentation, les pertes de puissance magnétiques et de frottement étant pratiquement constante, sont figurées par le courant I_o , les pertes Joule par R_b , il s'en suit que la puissance utile est donnée par :

$$P_h = V_m \times I_m = \frac{n}{K_v} \times I_m \text{ en Watts.}$$

C'est la puissance transmise à l'hélice.

Beaucoup de moteurs ont une valeur de « I_o » comprise entre 1 et 2A.

IV.13.2. Limitation du moteur:

Elle est essentiellement due à son échauffement. Elle est produite par l'effet Joule = $R_b \times I^2$, par les frottements et les pertes magnétiques.

L'effet Joule est très important à fort courant. Un courant de 50A dans des bobines de $30\text{m}\Omega$ développe une puissance de 75W. De bons fers à souder ont ce niveau de puissance. Il faut bien aérer le moteur et ne pas dépasser le courant maximum préconisé par le constructeur.

Une température excessive peut :

- détériorer les vernis de fixation des bobines, et entraîner des courts-circuits,
- décoller les aimants du stator
- démagnétiser partiellement les aimants.

IV.13.3. Calcule des différents paramètres :

Puissance fournie par la batterie : $P_{\text{batt}} = V \times I$

Puissance perdue par effet Joule : $P_j = (R_c + R_b) \times I^2$

Puissance perdue par frottement et le circuit magnétique :

$P_f = V_m \times I_o$.

$$V_m = V - (R_c + R_b) \times I, \text{ soit } P_f = [V - (R_c + R_b) \times I] \times I_o.$$

Puissance utile du moteur transmise à l'hélice :

$$P_h = V_m \times (I - I_o).$$

$$P_h = [V - (R_c + R_b) \times I] \times (I - I_o).$$

Un critère usuel de bon fonctionnement est un bon rendement, rapport de la puissance utile transmise à l'hélice sur la puissance fournie par la batterie :

$$R = \frac{P_h}{P_{\text{batt}}}$$

En remplaçant P_h et P_{batt} par les expressions précédentes, dérivant par rapport à la variable I , on trouve que le rendement maximum est obtenu au courant optimum.

$$I_{\text{opt}} = \sqrt{I_o \times I_{\text{cc}}}$$

Le courant de court-circuit est :

$$I_{\text{cc}} = \frac{V}{R_c + R_b}$$

IV.13.3.1. Le rendement maximum :

Il vaut :

$$R_{\text{max}} = \frac{(\sqrt{I_{\text{cc}}} - \sqrt{I_o})^2}{I_{\text{cc}}}.$$

En fait le rendement ne décroît pas beaucoup pour des courants allant de 1.5 à 2 fois le courant optimum. C'est une valeur à choisir à "plein puissance" au sol. ensuite le courant diminue un peu et le moteur tourne au meilleur rendement.

IV.13.4. Choisir la motorisation électrique :

IV.13.4.1. Choisir une puissance d'alimentation :

En moyenne, on peut choisir :

- 150 watts/kilo pour un fonctionnement calme, capable de décoller d'une piste,

- 180-220 watts/kilo pour la voltige douce,
- 250-300 watts/kilo pour la voltige puissante.
- 350 watts/kilo ou plus pour les montées verticales, ce qui nous intéresse.

IV.13.4.2. Evaluer la masse totale :

Ce n'est pas chose aisée. Si l'on peut connaître la masse du modèle sans la motorisation, en moyenne la masse de la motorisation (batteries, contrôleur, moteur, fixations, hélice) voisine la moitié de la masse finale du modèle avec des batteries NiMH, et le tiers avec des batteries Li-Po (lithium-polymère).

Suivant le type de l'appareil, on applique le critère du nombre de watts/kilo pour donner une valeur à la puissance d'alimentation.

$$P_{\text{batt}} = \frac{\text{masse du drone} \times \text{watts}}{\text{kilo}}$$

Ce n'est qu'a après avoir effectué des évaluations plus précises que l'on corrige cette valeur.

IV.13.4.3. Choisir le moteur qui correspond à ce niveau de puissance :

Parmi les catalogues des fournisseurs retenir les moteurs qui peuvent développer la puissance demandée ou un peu plus. Un moteur pouvant donner beaucoup plus de puissance serait plus lourd et plus cher. A puissance égale, un moteur à fort KV et un plus fort courant maximum permet d'utiliser une batterie avec moins d'éléments, mais à plus forte capacité pour une autonomie équivalente.

IV.13.4.4. Choisir un contrôleur et une batterie :

Le moteur étant choisi, on peut décider de le faire fonctionner près de son courant maximum dans la période où il consommera le plus, c'est-à-dire, a pleine puissance et faible vitesse au décollage. Après le choix de la tension de batterie on peut revenir sur cette valeur pour chercher un courant de l'ordre de 1.7 fois le courant de rendement maximum (s'il est compatible avec le courant maximum spécifié par le constructeur). Le rendement de la motorisation sera meilleur au cours du vol. La

batterie doit pouvoir fournir ce courant : une batterie spécifiée à 10C « coefficient de décharge de la batterie » peut débiter un courant égal à 10 fois la valeur numérique de sa capacité, spécifiée à 15C ce sera 15 fois cette valeur numérique, et à 20C ... ainsi de suite.

Le courant prévu est de 90A, alors on prendra une batterie de type 20 C. la capacité minimale est $80/20 = 4$, on prendra 4.4 Ah, ou plus si on veut plus d'autonomie (mais la masse augmentera en conséquence).

- Une batterie Ni-MH actuelle ne devrait pas être utilisée au-delà de 12C pour une meilleure durée de vie donc ne conviendra pas pour l'application en cours.
- Le courant maximum utilisable d'une LiPo (lithium-polymère) est généralement indiqué sur la batterie.)

La valeur du courant maximum et la puissance d'alimentation demandée étant spécifiées, la tension que doit fournir la batterie est calculée par la formule :

$$V_{\text{batt}} = \frac{\text{Puissance}}{\text{Courant}}$$

Compte tenu de leurs résistances internes. A fort courant, on estime une tension externe à la batterie de l'ordre de 1.05 volt par élément Ni-MH et de 3.65 volts par élément Lithium-Polymère.

$$\text{Nombre d'éléments} = \frac{V_{\text{batt}}}{\text{tension par élément}}$$

- On prend le nombre entier plus proche.
- L'entier supérieur permettra un excédent de puissance, l'entier inférieur laissera un défaut de puissance. Faire un choix convenable d'après la manière dont on a choisi le nombre de watts/kilo du modèle.
- Il ne reste plus qu'à choisir dans le catalogue des constructeurs, le contrôleur qui accepte la tension de batterie requise et le courant maximum envisagé.

Note : Le calcul du courant optimum passe par :

$$I_{\text{CC}} = \frac{V_{\text{batt}}}{R_b + R_c}$$

Où R_c est la résistance du contrôleur auquel on peut ajouter une valeur de l'ordre de $5\text{m}\Omega$ pour les câbles et prises.

IV.13.4.5. Refaire l'évaluation de la masse :

Le moteur et le contrôleur sont choisis d'après catalogue de constructeurs, on a pu noter leur masse.

Le type et la capacité de la batterie sont aussi choisis. Il faut la choisir d'après catalogue et noter sa masse.

- Ajouter approximativement les masses du support moteur, du support d'hélice, de l'hélice, des câbles électriques et connecteurs.
- Faire le bilan de poids du dispositif complet.
- Voir si le nouveau résultat obtenu est compatible avec l'évaluation de départ pour ce type de modèle compte tenu de la tolérance possible sur le nombre de watts par kilo. Sinon on reprendre les évaluations précédentes jusqu'à obtenir un ensemble de valeurs homogènes.

IV.13.4.6. Choisir l'hélice :

Plusieurs étapes intermédiaires permettent de faire le bon choix.

1^{ère} étape : Puissance et vitesse de rotation.

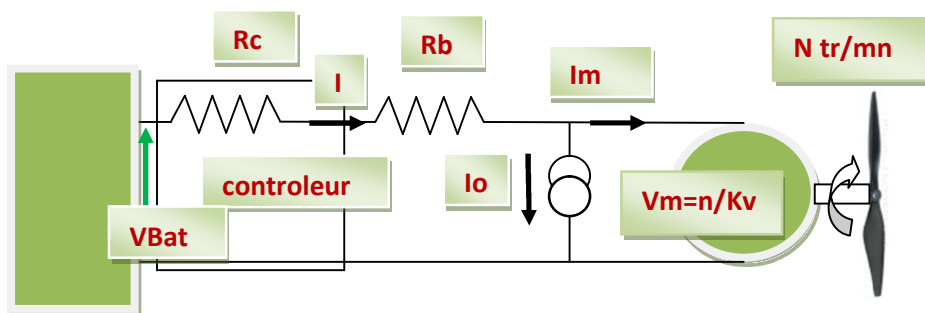


Figure IV.20: Les paramètres électriques servants aux calculs de la puissance délivrée à chaque hélice.

D'après le schéma électrique global de la propulsion, en portant les valeurs numériques précédente au maximum de puissance,

Calculer V_m :

$$V_m = V - (R_c + R_b) \times I \quad .$$

Et en déduire n :

$n = V_m \times K_v$, vitesse de rotation prévue en tours/minute. (V du schéma est la tension de batterie sous fort courant).

2^{ème} étape : Trouver l'hélice adaptée.

Trouver une hélice qui recevant la puissance utile maximum demandée (on la note Ph) tourne à la vitesse n tr/mn.

$$Ph = V_m \times I_m.$$

$$Ph = [V - (R_c + R_b) \times I] \times (I - I_o).$$

Une façon possible de déterminer ce choix réside dans l'utilisation d'une formule aérodynamique (déduite de la formule de Boucher) qui relie vitesse et puissance.

$$Ph = 1.8 \times 10^{-7} \times n^3 \times Pa \times N_p \times K_p \times D^4 \quad .$$

Ces 2 expressions doivent être égales.

A noter :

- D est le diamètre de l'hélice en mètre,
- Pa est le pas en mètre (le pas de l'hélice),
- n est la vitesse de rotation en tr/mn,
- N_p un coefficient qui vaut 2 pour une bipale. (hélice a 2 pale d'un coté et de l'autre), 3.2 pour une tripale, 4.4 pour une quadri-pale,

- K_p est un coefficient de forme des hélices, permettant d'ajuster cette formule issue d'un certain nombre de simplifications. On peut utiliser des valeurs approximatives de $K_p = 1,06$. selon le type d'hélice, son poids ...etc.

A partir de ces formules un simple tableau de type Excel qu'on construit pour des hélices bipales avec deux entrées : le nombre de tr/mn, et les caractéristiques (diamètre et pas) des hélices les plus courantes. Le coefficient K_p est à modifier suivant le type d'hélice.

Pour terminer :

- On équipe le modèle avec les éléments choisis.
- Il faut s'assurer que toutes les imprécisions qui entrent dans les estimations ne conduisent pas à une valeur de courant trop forte ou trop faible.
- Dans le cas d'un courant trop fort on choisi une nouvelle hélice un peu plus faible.
- Dans le cas d'un courant trop faible (la puissance obtenue est plus faible que prévue) alors on choisi une nouvelle hélice un peu plus forte.
- il ne reste qu'à tester le model et voir comment la propulsion donnera ce que l'on attendait.

A)-Applications numériques :

Type vole : voltige douce.

Largeur Maximale 70cm masse sans motorisation 1100g.

Mais chaque moteur (parmi quatre) supporter un quatrième de cette masse. Soit $1100g / 4 = 275g$.

1°) le critère de puissance : 110watts/kilo (voltige douce).

2°) Estimation de la masse totale (cela sous entend pour chaque moteur):

Type de batterie choisie : li-po.

Masse totale estimée :

Masse à vide x 2.

AN :

$$1.1 \times 2 = 2.2 \text{kg.}$$

Puissance d'alimentation :

$$2.2 \times 110 = 242 \text{W.}$$

3°) Choisir le moteur qui correspond à la puissance :

On prend le moteur le plus proche en termes de caractéristiques à celui qu'on a fabriqué.

La puissance du moteur est :

$$P_m = 60.5 \text{W.}$$

Masse du moteur = 85g.

Courant max permanent $I_{max} = 25 \text{A}$.

Courant de pointe $I_p = 35 \text{A}$.

$$K_v = 900 \text{tr/mn.}$$

$I_o = 1.5 \text{A}$; (caractéristiques des transistors Mosfets employés).

$R_b = 30 \text{m}\Omega$, en "Timing low".

4°) Choisir la batterie et le contrôleur :

On commence par choisir la valeur de courant voulue à la puissance maximale retenue. On décide une valeur de départ pour l'analyse de 95% de maximum du courant permanent (rendement atteint par ce type de moteurs) :

$$I_{\text{Voulu}} = 25 \times 0.95 = 23.75 \text{A.}$$

(Ce choix économise le moteur, il restera à une température plus basse. Mais si l'on estime que le maximum de courant ne sera pas fréquemment utilisé, on peut viser 90% du courant de pointe).

Pour disposer des 60.5W demandés il faut une tension de batterie :

$$V_{\text{batt}} = \frac{\text{Puissance}}{\text{Courant}} .$$

AN:

$$\frac{242}{24} = 10.1 \text{ volts} .$$

On a choisi des éléments LIPO, (tension 3.75volts par éléments sous fort courant).

Nombre d'éléments :

$$\frac{10.1}{3.70} = 2.73 .$$

Le choix se porte sur une batterie de 3 éléments, pour une tension sous fort courant estimée à $3 \times 3,70 = 11.1$ Volts.

Avant de poursuivre comparons le courant choisi au courant de rendement optimum. Il faut calculer le courant de court-circuit :

$$I_{\text{cc}} = \frac{V_{\text{batt}}}{(R_b + R_c)}$$

La résistance de notre contrôleur est de $12\text{m}\Omega$, valeur statistique moyenne pour $5\text{m}\Omega$ de câbles et connexions et $7\text{m}\Omega$ pour les contrôleurs de cette gamme.

Soit :

$$I_{\text{cc}} = \frac{11,1}{0.030 + 0.012} = 264\text{A} , \text{ avec } I_o = 1.5\text{A}$$

Le rendement maximum :

$$R_{\text{max}} = \sqrt{264 \times 1,5} = 19,9\text{A} .$$

Le rapport courant choisi / rendement max = $(23.75 \times 4) / 19.9 = 4.77$.

Cette valeur est très proche de la zone préconisée entre 3.5 et 4.9.

Ainsi nous conservons ce choix.

Le pack de batteries sera donc un pack de 3 LIPO 3300mAh. Le courant demandé sera de :

$$\frac{23.75 \times 4}{4400} = 20C \quad , \text{ tout à fait raisonnable pour la durée de vie de la batterie.}$$

IV.13.5. Refaire l'évaluation de masse :

Une cellule LIPO pèse 100g environ.

- Le modèle sans motorisation à une masse de 1100g.
- La masse du contrôleur est de 80.
- La masse du moteur est de 85g.
- Pour câbles, hélice, fixation hélice on ajoute 65g.

On arrive à une estimation de masse totale du modèle : 1825g

IV.13.5.1. Choisir l'hélice :

Calculer V_m :

$$V_m = V_{\text{batt}} - (R_c + R_b) \times I$$

AN :

$$V_m = 11.1 - (0.012 + 0.030) \times 23.75 = 10.1 \text{Vots}$$

Vitesse du moteur et de l'hélice :

$$N = K_v \times V_m .$$

AN:

$$900 \times 10.1 = 9090 \text{tr/mn.}$$

2^{ème} étape: Puissance utile du moteur transmise à l'hélice :

$$P_h = V_m \times (I_e \times I_o) .$$

AN :

$$P_h = 10.1 \times (23.75 - 1.5) .$$

$$P_h = 224 \text{Watts} .$$

On va rechercher les hélices consommant une puissance comprise entre 120 et 310 watts, correspondant aux vitesses entre 7000 et 11000 tr/mn qui encadrent la valeur calculée de 9090tr/mn.

IV.14. Conclusion :

Après cette étude on prouve l'évidence de la supériorité des moteurs sans balais sur les autres types de moteurs. En effet malgré les quelques inconvénients liés notamment à la complexité des contrôleurs de ce type de moteurs, mais les avantages prennent rapidement le dessus et les résultats des calculs le confirment sans équivoques.

Chapitre V:

**Communications et interfaces de
commande à distance.**

V.1.Introduction :

Le drone reste et jusqu'à maintenant un engin radiocommandé. En fait malgré sont automatisation et quel que soit sont degré d'autonomie en terme de contrôle intelligent ; des liaisons radio dans un sens ou dans l'autre sont à tout moment nécessaires pour des raisons diverses. Cette liaison de données une chaine complexe qui dépend étroitement de l'évolution technologique notamment dans le domaine des télécommunications.

V.2.Composants essentielle d'une chaine de communication et contrôle à distance :

V.2.1.L'Émetteur:

L'émetteur simplifié de la *figure ci-dessous* comprend les trois sous ensembles suivants :

- Un circuit de traitement en bande de base ;
- Un modulateur;
- Un amplificateur de puissance.

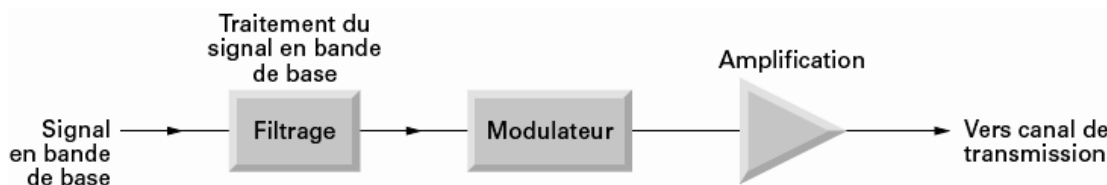


Figure V.1: Schéma synoptique d'un émetteur.

V.2.2.1.Génération de la fréquence pilote :

Avant tout on doit réfléchir à la structure qu'il doit adopter pour générer la porteuse. Cette porteuse est générée par un oscillateur local. Celui-ci doit être stable en fonction du temps, de la température et de la tension d'alimentation. Aussi Il devra avoir un faible bruit de phase ou de fréquence au voisinage de la fréquence centrale. Il délivrera une puissance de sortie compatible avec les étages suivants.

Dans notre application la fréquence porteuse est de 72Mhz, donc on à recours à un étage oscillateur suivi d'étages multiplicateurs. Cette solution sera évidemment mise en concurrence avec un oscillateur asservi par une boucle à verrouillage de phase, qui donne les mêmes résultats en termes de précision et stabilité qu'un oscillateur à quartz. En effet c'est une solution qui s'adapte à tous les types démodulation, amplitude ou fréquence.

En modulation de fréquence(le type de modulation qu'on a adopté) , le signal de sortie de l'oscillateur est envoyé directement vers les étages de sortie. le choix est restreint et se limite à donner une réponse à la question de modulation directe ou modulation indirecte. Finalement, et ceci quel que soit le type de modulation, il se peut que l'on ne puisse moduler directement la fréquence porteuse. Dans ce cas, on a recours à une ultime étape de transposition en fréquence qui met en jeu oscillateurs locaux, mélangeurs et filtres. La structure très générale de l'émetteur devient alors celle de la *figure ci dessous* et cette structure reste applicable quel que soit le type de modulation envisagé.

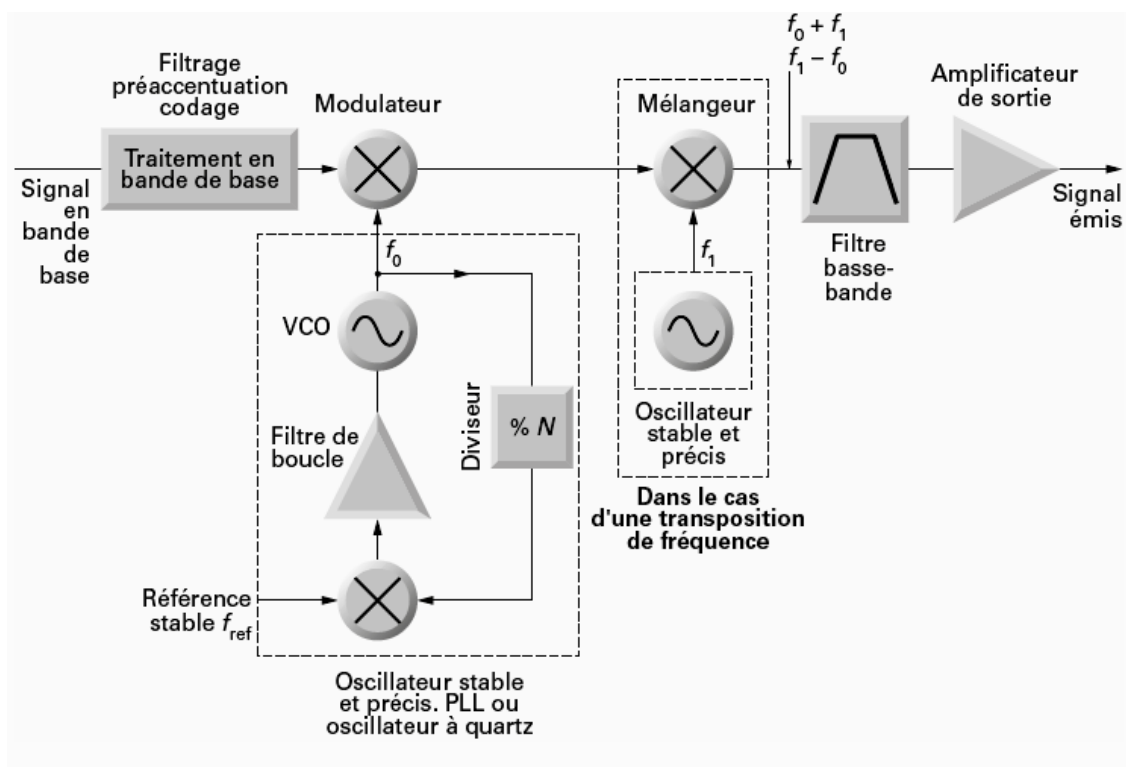


Figure V.2: structure générale de l'émetteur.

V.2.2.2. Amplificateur de sortie :

Pour concevoir le ou les étages amplificateurs de sortie il faut pouvoir répondre à deux questions simples :

- Quel est le type de modulation ?
- Quelle puissance de sortie doit-on envisager ?

a)-Classe de fonctionnement de l'amplificateur de sortie :

La réponse aux deux questions sélectionne instantanément la classe de fonctionnement de l'amplificateur. On travaille en modulation de fréquence alors la classe de fonctionnement peut être A, B, AB ou C. On préférera la classe C qui donne les meilleurs résultats en terme de rendement (donc de puissance). La figure suivante montre les différences fondamentales de fonctionnement entre ces quatre types de classes.

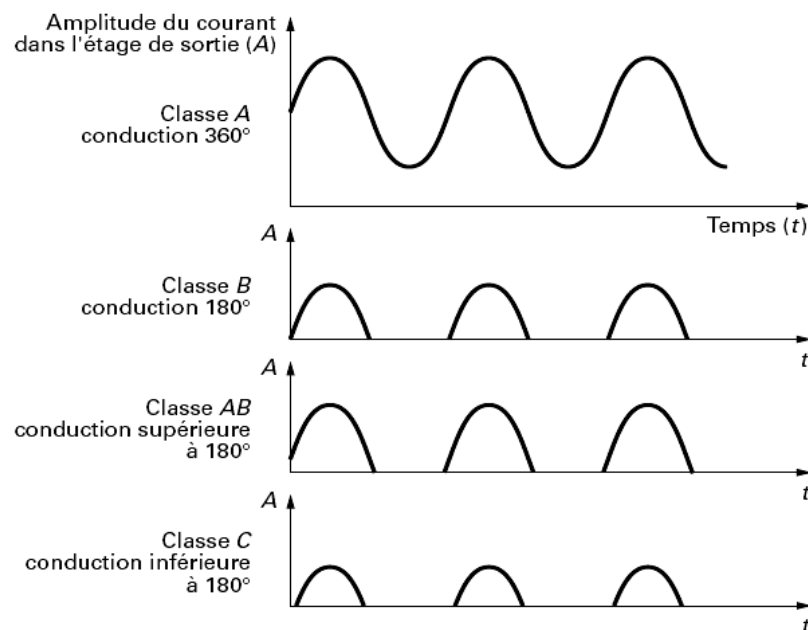


Figure V.3: Différence de fonctionnement des amplificateurs de puissance classe A, B, AB et C. $I_{collecteur} = f(t)$.

- ❖ En classe B le rendement théorique maximal est 78 %.
- ❖ En classe C on peut espérer avoir des rendements élevés de l'ordre de 90 %.

V.2.2.3. Adaptation d'impédance :

Finalement il s'agit d'envoyer la puissance délivrée par l'étage de sortie au médium de transmission. Un circuit d'adaptation d'impédance sera donc intercalé entre la sortie de l'amplificateur et la charge. La charge peut être une antenne, un câble coaxial, un câble bifilaire ou un guide d'onde.

Pour conclure :

Lors de la conception de l'émetteur les points essentiels sont :

- ❖ La stabilisation de la porteuse;
- ❖ Le principe adopté pour la modulation ;
- ❖ L'amplification, l'adaptation et le rendement des étages finaux.
- ❖ Les notions de coût ne seront pas écartées et intégrées avec le coefficient de pondération correspondant à l'application.

V.2.3. Le Récepteur :

Le récepteur est très souvent beaucoup plus complexe que l'émetteur. La structure finale d'un récepteur découle d'une suite de compromis entre les différents paramètres influant sur les performances. Tous ces paramètres sont étroitement imbriqués, il n'y a ni solution idéale ni solution universelle.

V.2.3.1. Rôle du récepteur :

Le récepteur reçoit une fraction de la porteuse modulée émise en présence de bruit et de multiples autres signaux de puissance et de fréquences diverses et inconnues.

Le rôle fondamental du récepteur est de démoduler la porteuse et de restituer le signal modulant original (analogique ou numérique) après avoir amplifié le signal reçu à travers un ou plusieurs étages préamplificateurs.

V.2.3.2. Récepteurs à un changement de fréquence :

Un filtre d'entrée, couplé aux circuits du démodulateur permettrait de sélectionner le canal souhaité. Ceci sous-entend un synchronisme parfait entre les filtres et les circuits du démodulateur.

Par ailleurs, plus la fréquence d'entrée est élevée, plus la largeur de bande du filtre d'entrée est importante, si l'on admet que le coefficient de surtension est fixe. Toute l'amplification est reportée sur les étages d'entrée, et ceci pose d'autant plus de problèmes (voir plus loin dans le récepteur à double changement de fréquences) que la fréquence d'entrée est élevée. Les raisons d'abandonner cette structure sont donc nombreuses. On opte alors pour une transposition de fréquence et le schéma synoptique du récepteur devient celui de la figure suivante :

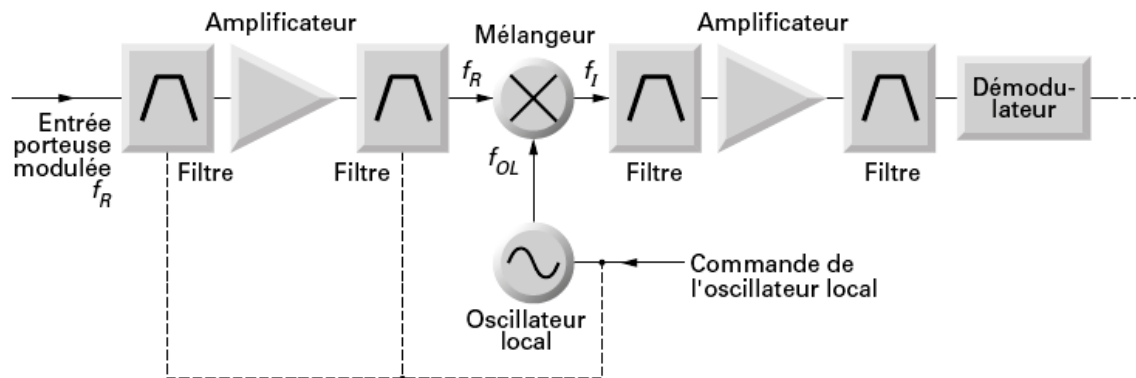


Figure V.4: Schéma synoptique du récepteur à un changement de fréquence.

V.2.3.3. Récepteurs à double changement de fréquence :

Un double changement de fréquence résout simultanément les problèmes lors du changement de fréquence unique.

Supposons que l'on souhaite recevoir des canaux espacés de 5 kHz (suffisant pour notre application) ayant, pour l'exposé une largeur *BP* (bande passante) de 5 kHz. Le coefficient de surtension Q du filtre passe bande à la fréquence intermédiaire f_i vaut f_i / BP . Pour que cette valeur reste raisonnable, f_i ne doit pas être trop élevée, mais doit être importante pour faciliter la réjection de la fréquence image. On va donc

opérer en deux étapes; un premier changement de fréquence facilitant l'élimination de la fréquence image et un second changement de fréquence sélectionnant le canal étroit.

Le synoptique résultant est celui de la figure suivante. Comme précédemment les signaux d'entrée sont envoyés au port RF du premier mélangeur. Le premier oscillateur local est variable, la sélection de la fréquence f_{OL1} permet de recevoir un canal parmi les n canaux présents et sélectionnés par les filtres d'entrée. Le canal sélectionné est alors transposé dans une fréquence f_{i1} ; cette première fréquence intermédiaire f_{i1} est transposée en une fréquence f_{i2} telle que f_{i2} inférieur à f_{i1} .

Notre récepteur ayant les valeurs de fréquence intermédiaire suivantes :

$$f_{i1} = 10.7\text{Mhz} \text{ } 21,4 \text{ MHz}$$

$$f_{i2} = 455 \text{ kHz}$$

$$B_p \text{ canal} = 5 \text{ kHz}$$

$$B_{p f_{i2}} = 5 \text{ kHz}$$

La fréquence du second oscillateur local peut être choisie par l'une ou l'autre des relations :

$$f_{OL2} = f_{i1} - f_{i2} = 10,7 - 0,455 = 10,245 \text{ Mhz.}$$

$$f_{OL2} = f_{i1} + f_{i2} = 10,7 + 0,455 = 11,155\text{Mhz.}$$

Supposons que le premier canal se situe à 72Mhz, pour sélectionner ce canal l'oscillateur local est à 82,7Mhz. Supposons que, dans l'objectif de supprimer les fréquences image, le filtre d'entrée sélectionne la bande 70 à 100MHz. Le récepteur est alors capable de recevoir tous les canaux présents dans cette bande de 30 40MHz. Il s'agit maintenant de comparer ce récepteur à double changement à un récepteur unique ayant une fréquence intermédiaire égale à f_{i2} soit 455 kHz. Soit comme précédemment le premier canal à 70 MHz, la fréquence image est :

$$70 + (2 \times 0,455) = 70,91 \text{ MHz.}$$

Même avec l'hypothèse optimiste que l'on puisse sélectionner la bande d'entrée de 70 à 70,5MHz, il est clair que la couverture du récepteur n'est que de 500 kHz alors qu'elle était de 30MHz dans le cas précédent.

Les avantages du récepteur à double changement de fréquence sont alors évidents. La première fréquence intermédiaire facilite la réjection de la fréquence image à l'entrée. Le deuxième changement de fréquence facilite le filtrage pour des canaux pouvant être très étroits.

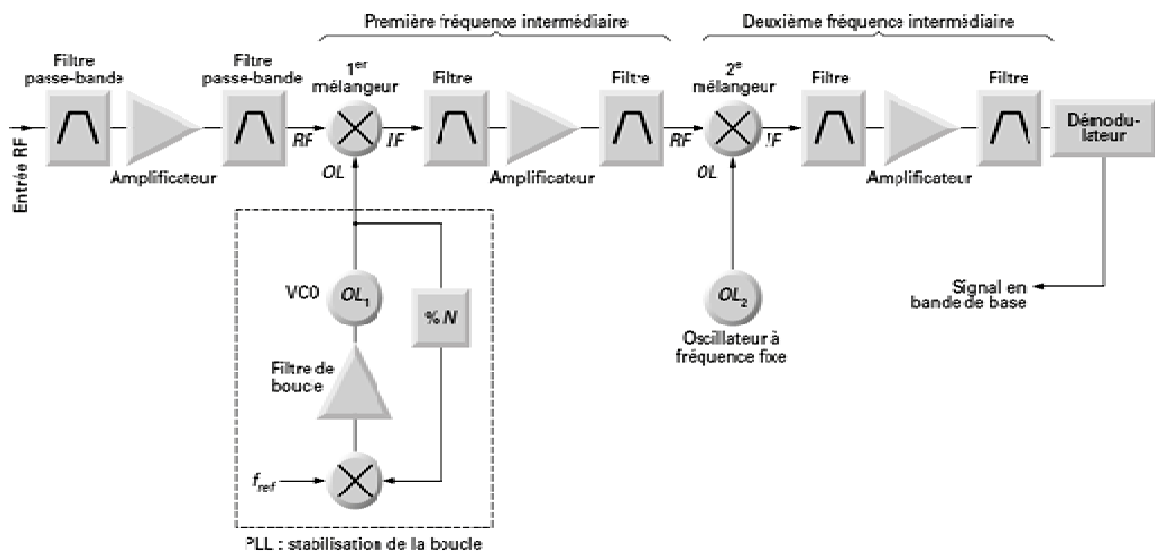


Figure V.5: Récepteur à double changement de fréquences.

V.2.3.3.1. Filtres passe-bande d'entrée :

Le filtre passe-bande d'entrée a pour rôle principal la sélection d'un canal ou un groupe de canaux et la réjection des fréquences image. Sa perte d'insertion est un facteur important puisqu'elle est égale à son facteur de bruit. Le premier filtre doit avoir une perte d'insertion minimale pour un facteur de bruit minimal. Le second filtre passe-bande a un rôle plus complexe. Il pallie l'isolation insuffisante du premier mélangeur.

Ce filtre doit éviter la propagation du signal d'oscillateur local vers l'entrée du récepteur. Les deux filtres d'entrée ont une influence sur la réjection d'un éventuel signal à la fréquence FI présent à l'entrée du récepteur.

V.2.3.3.2. Amplificateurs d'entrée :

Cet amplificateur est le maillon le plus délicat. On lui demande les performances suivantes :

- ❖ Faible bruit, puisque sa participation au facteur de bruit global est importante;
- ❖ Grand gain, pour les mêmes raisons ;
- ❖ Point d'interception élevée.

Ces performances influent sur la réjection des réponses parasites et sur la sensibilité du récepteur (facteur de bruit et gain).

V.2.3.3.3. Mélangeurs :

Les performances du mélangeur agissent sur la réjection des réponses parasites. La perte de conversion est de moindre importance si l'amplificateur d'entrée a un gain important. L'isolation entre les ports RF-OL et OL-IF est importante pour éviter la propagation du signal OL ayant un fort niveau, et susceptible de générer des problèmes de distorsion d'intermodulation dans l'étage d'entrée et dans les étages à fréquence intermédiaire.

V.2.3.3.4. Oscillateur local :

L'oscillateur local doit être stable. Le bruit de phase et les raies parasites au voisinage de la porteuse limiteront les performances. Le niveau des harmoniques a une influence directe sur le nombre et l'importance des réponses parasites.

V.2.3.3.5. Filtres à la fréquence intermédiaire :

Le filtre placé immédiatement en sortie du mélangeur, s'il existe, aura pour rôle principal la réjection de l'oscillateur local et de ses éventuels harmoniques dans la chaîne d'amplification FI. Ses performances ont une influence sur le nombre de réponses parasites et leur réjection. La largeur de bande du filtre FI a une incidence directe sur la sensibilité du récepteur.

V.2.3.3.6. Amplificateurs à la fréquence intermédiaire :

En modulation d'amplitude, les amplificateurs sont du type à commande automatique de gain. Leur performance a une répercussion sur les distorsions pour le signal en bande de base. Pour les modulations angulaires (comme en fréquence, phase), les limiteurs ont une incidence sur le rapport signal sur bruit après démodulation.

V.2.3.3.7. Signaux perturbateurs externes :

Les boucles à verrouillage de phase sont omniprésentes dans tous les systèmes de transmission. Chacune de ces boucles reçoit une fréquence de référence stable issue en général d'un oscillateur à quartz. Il est important que ce signal, nécessaire, ne se transforme pas en un signal perturbateur majeur.

V.2.3.3.8. Démodulateur :

Le paramètre le plus important du démodulateur est sa linéarité qui influe directement les distorsions du signal reçu. Le choix d'une structure particulière peut aussi avoir une influence sur le rapport signal sur bruit, comme l'emploi d'un démodulateur à PLL en FM par exemple.

A noter que tous les récepteurs actuels sont vendus sous forme d'un circuit intégré unique intégrant toutes les fonctions précédentes associées à d'autres modules de traitement numériques (microcontrôleurs) pour la sélection numérique des canaux (à l'aide d'emploi de diodes varicaps), le scanning, l'enregistrement et la numérotation des canaux...etc.

V.3. La radiocommande :

La radiocommande sous-entend un système radio permettant de commander des appareils divers à distance. Dans notre cas, ce système permettra de faire le lien entre le pilote et le système de commande du quadri-rotor. Il est intéressant de connaître ce que le récepteur transmet comme ordre sur ses différentes voies en fonction des positions des commandes de la radio. Un ordre est en fait un signal PWM avec un rapport cyclique qui varie de 0.05 à 0.1, une tension d'amplitude 5 V et une période

d'environ 20ms. De plus, le récepteur ne transmet pas tous les créneaux en même temps sur ses différentes voies, il s'occupe de chacune des voies alternativement en suivant une horloge. La figure suivante présente une visualisation des ordres du récepteur.

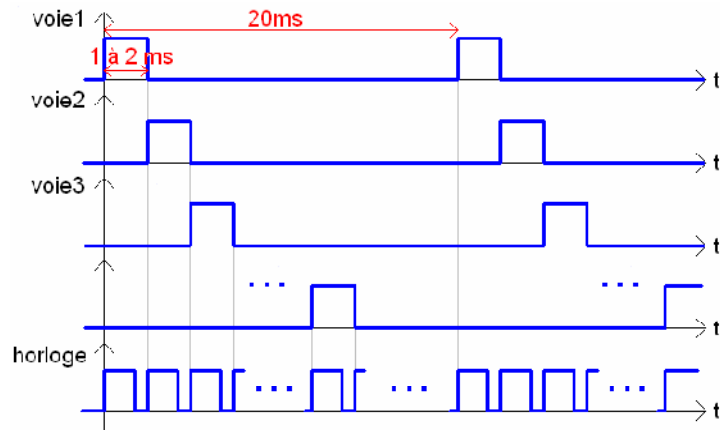


Figure V.6: codage PPM/PCM

La radiocommande utilise une transmission (émission-réception) radio de 8 voies, permettant le codage de données en PPM (pulse position/phase modulation) ou PCM (pulse code modulation). Seules 5 voies seront utilisées dans notre cas. La figure suivante représente la structure de récepteur de radiocommande et fait apparaître les sens de variation des différentes commandes.

Voie 1 : commande de roulis.

Voie 2 : commande de tangage.

Voie 3 : commande de puissance.

Voie 4 : commande de lacet.

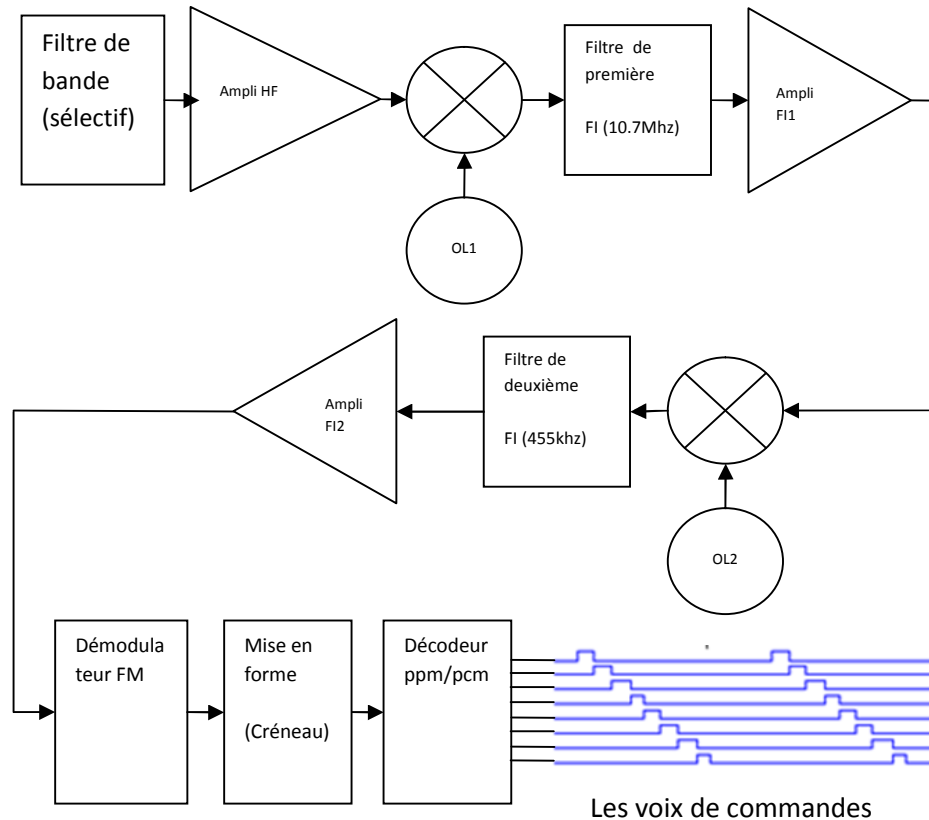


Figure V.7: structure d'un récepteur de télécommande 8voix.

Les voies indiquées correspondent à celles des récepteurs couramment utilisés et vendus. Concernant les commandes de tangage et de roulis, l'utilisation de toute la plage de 0% à 100% peut ne pas être judicieuse. Afin de faciliter le pilotage, on peut améliorer la résolution autour de 50% en programmant les valeurs minimales et maximales avec la radiocommande (par exemple : 25% et 75%).

A noter que ces deux types de codages à savoir (ppm et pcm) qui sont très anciens mais restent très employés actuellement dans la radio commande, le contrôle à distance d'engins, robots ...etc.

La figure suivante montre la chaîne de contrôle de drone par un transmetteur à plusieurs voix PPM/PCM à l'aide d'une manette ; cette méthode de commande se montre très maniable et souple et convient à notre application.

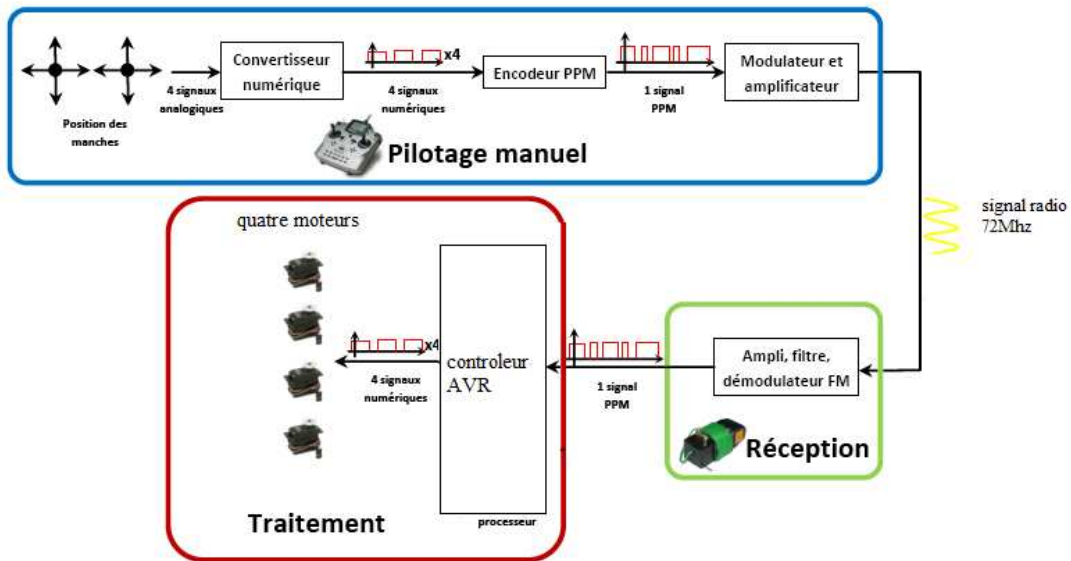


Figure V.8: structure générale de la radiocommande (manuelle) pour le contrôle de drone.

V.4.Communications a base de modules xbee :

Notre projet consiste à réaliser une liaison entre un module xbee émetteur (lié à un PC) et un autre liés à la carte de control de vol (embarquée) à l'aide de protocole zigbee. Le logiciel de contrôle étant une interface java de commande ou bien labview .ce dernier (pour les versions à partir 2009 et supérieur) dispose d'outils très évoluées notamment des bibliothèques dédiées à la communication et le contrôle des cartes arduino.

V.4.1.Présentation module xbee :

Les modules Xbee sont des transcrives (émetteur/récepteur) radio au standard (protocole) Zigbee / IEEE 802.15.4, lesquels sont spécialement conçus pour la réalisation de systèmes de communication au sein de réseaux de capteurs sans fil. De petites dimensions, ces modules se distinguent par leur grande simplicité d'utilisation et leur coût très compétitifs qui les prédestinent également à de très nombreuses autres applications.

Pouvant être utilisés sans aucune configuration préalable (à la manière d'une liaison série pouvant s'apparenter à un câble série radio virtuel) ou bien avec une configuration a l'aide d'un l'hyper terminal comme la liaison série et pour notre projet

on va utiliser X-CTU pour la configuration des deux module(Emetteur et Récepteur), les modules disposent également d'une multitudes de modes de fonctionnement très facilement accessibles via des commandes AT ou API (Application Programming Interface).



Figure V.9: Modules Xbee en communications.

Les modules Xbee se caractérisent (pour le module standard) par une portée de 30m à 100m, une puissance d'émission sélectionnable de -10dBm à 0dBm, d'une vitesse de transmission de 250000 bps et d'une consommation de 50mA. Ils travaillent sur la bande de fréquence ISM (Industrial Scientific and Medical) 2.4 GHz où ils peuvent sélectionner le canal de transmission parmi les 16 disponible.

Ces canaux se caractérisent par une largeur théorique de 5MHz, dont seulement 2MHz sont réellement utilisés.

V.4.2.Structure matérielle :

Le module Xbee peut être connecté à n'importe quel circuit présentant une liaison UART pour autant que les niveaux de tension ne doivent pas être adaptés.

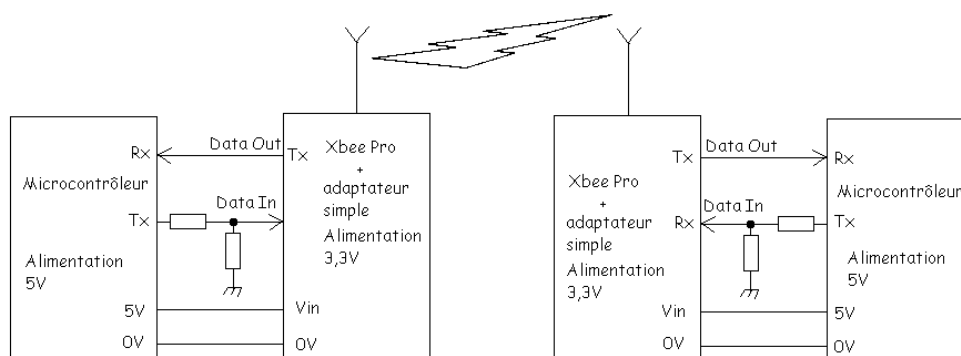


Figure V.10: illustration de la communication via Xbee.

La communication est de type asynchrone avec le module et l'interface l'UART de microcontrôleur. Et se fait selon la figure suivante. Chaque paquet est constitué d'un bit Start de niveau bas, suivi de 8 bits de données avec le bit de poids faible en premier, et enfin un bit stop de niveau haut. La synchronisation des données et la vérification de la parité sont assurées par le module UART.

Etant connecté à un circuit présentant une liaison série asynchrone, le module utilise un buffer (mémoire tampon) pour stocker les données transmises par l'UART via la patte DIN. Ce flux de données est contrôlé par le signal CTS. Lorsque le buffer ne dispose que d'un espace libre de 138 bits.

V.4.3. Trames API :

Les modules Xbee peuvent communiquer soit au travers de commandes AT, soit au travers de trames API (Application Programming Interface). Ces trames permettent de transmettre des données ainsi que des messages permettant de vérifier la réussite de la communication. Le type de transmission API est donc préférable dans le cas où on travaille en mode télémétrique ou des données des capteurs embarquées sur le drone par exemple ou bien l'état des accumulateurs sont transmis vers le PC de contrôle pour les afficher.

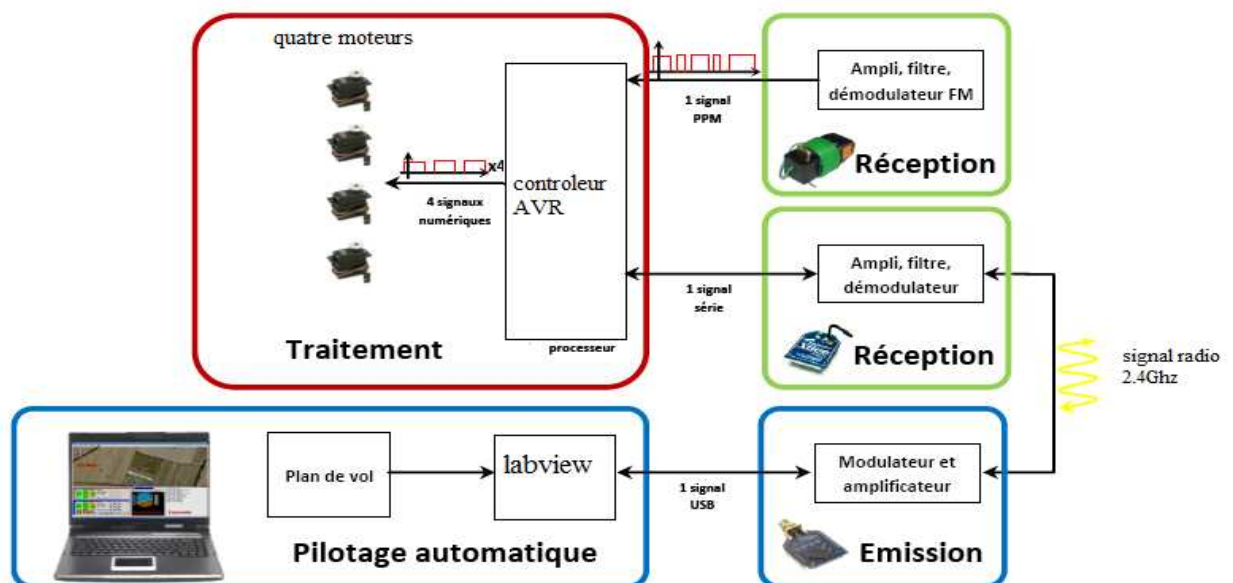


Figure V.11: communications à base de modules xbee à travers l'interface labview.

Etant donné que la liaison réalisée se fait par le module xbee encasté dans la carte, il est prévu dans le programme embarqué de se communiquer avec le xbee en mettant le Switch a la position RF. L'Arduino envoie les données issues du PC vers le module xbee émetteur qui les transmet a son tour via les ondes radio a travers le protocole zigbee vers le module xbee récepteur. A la réception la carte (arduino) de contrôleur de vole interprète les ces données et les exécute en temps réel.

V.4.4. Configuration de module xbee :

Une fois le module Xbee sur place, et en utilisant le logiciel X-CTU (logiciel dédié spécialement a la configuration et le contrôle des modules xbee), le programme peut être lancé. Il s'ouvre sur la fenêtre suivante :

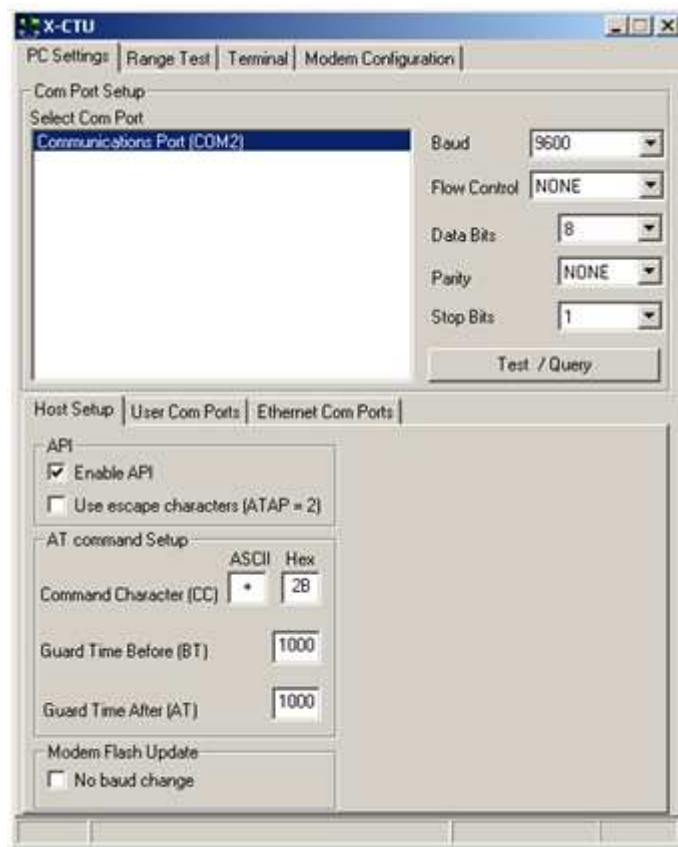


Figure V.12: configuration a l'aide de X-CTU.

Les paramètres Flow Control, Data Bits, Parity et Stop Bits doivent prendre les valeurs indiquées sur la fenêtre. La vitesse, 115200 bauds (le maximum) pour notre application, ce mode de transmission peuvent être modifiés selon les besoins.

Une fois les paramètres entrés, un test de communication peut-être effectué ce qui ouvre la fenêtre suivante:



Figure V.13: Fenêtre Test X-CTU

Après avoir cliqué sur ok, l'accès aux paramètres du module se fait en sélectionnant l'onglet «Modem Configuration». Un click sur «read» permet d'afficher la configuration du Xbee se trouvant sur la carte de test.

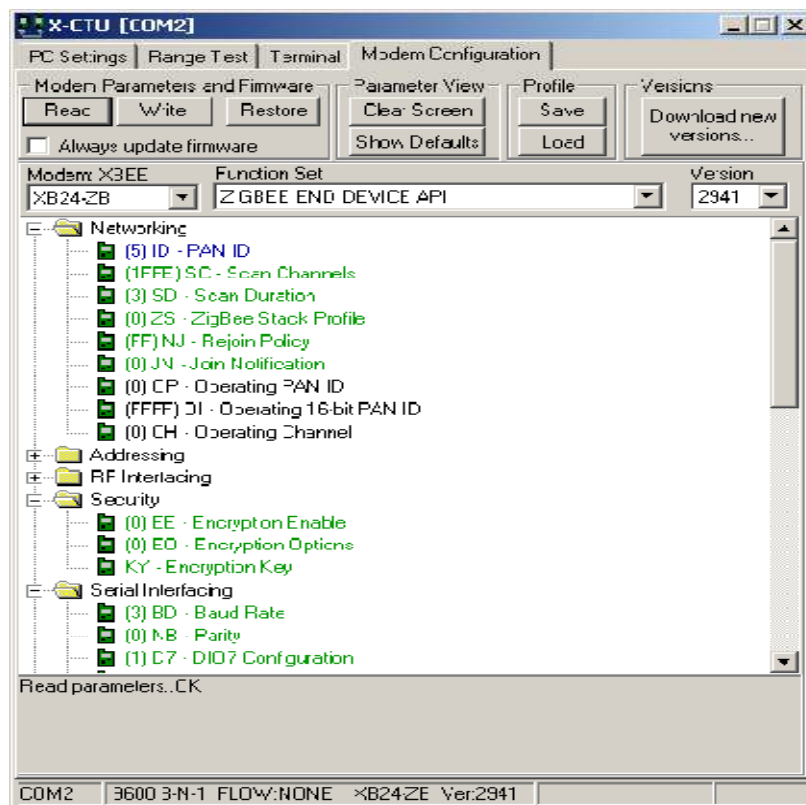


Figure V.14: Configuration du module xbee

Pour paramétrer le module en coordinateur ou en « end device », il suffit de le sélectionner dans la liste se trouvant dans la barre de recherche « Function Set ». Les paramètres importants pour l'application qui nous intéressent sont:

ID - PAN ID : Ce nombre doit être commun aux deux modules utilisés.

BD - Baud Rate : Définition de la vitesse de communication série du module qui est de 115200 bit/s.

SM -Sleep Mode : Définition du fonctionnement du sleep mode. Ce paramètre doit être mis à '1' pour les modules « End Device ». Le coordinateur n'a pas ce paramètre dans sa configuration.

Les autres paramètres n'ont pas besoin d'être changés. Une fois la configuration établie, le module peut-être programmé en cliquant sur « write ».

Le Xbee peut ensuite être retiré de la plaque après avoir coupé l'alimentation comme sur la figure suivante.

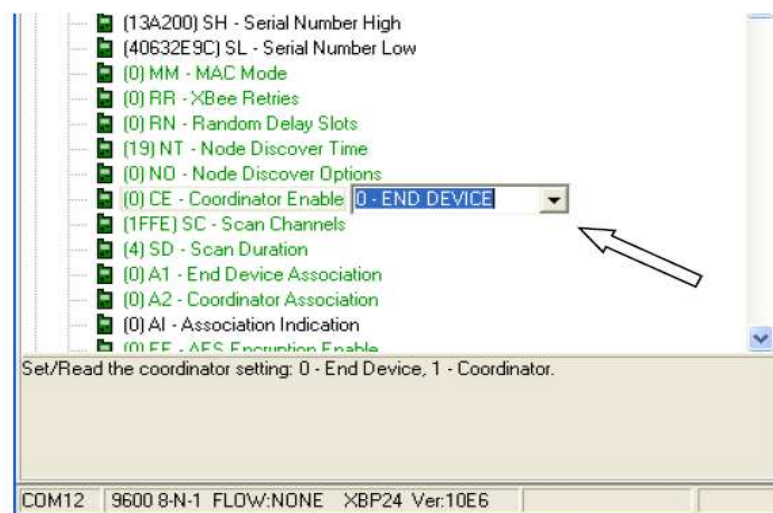


Figure V.15: Comment couper l'alimentation pour retirer le module xbee.

Le paramètre suivant est l'adressage de module de destination :

L'adresse du module en cours de configuration est par exemple \$0001 comme représenté sur la figure suivante.

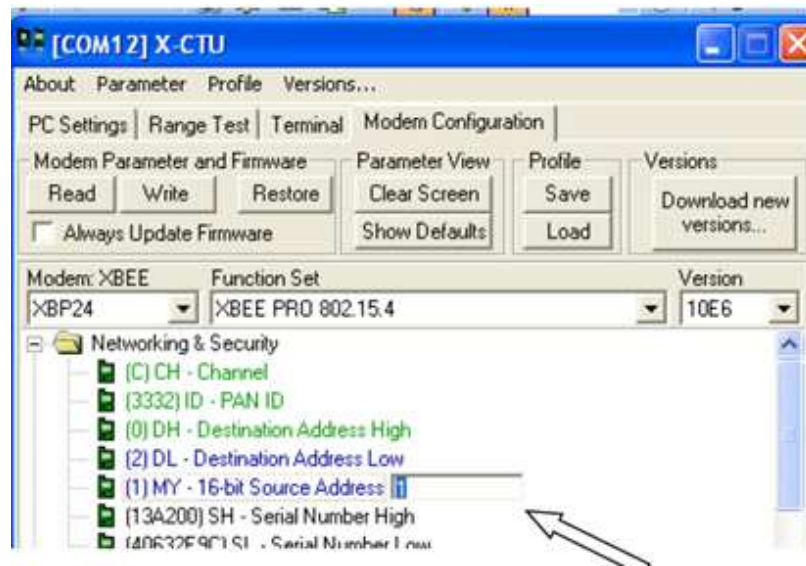


Figure V.16: L'adresse haute de module xbee émetteur.

- La partie basse de l'adressage court du module de destination :

L'adresse du module de destination est donc \$0002 comme il est donnée par la figure suivante

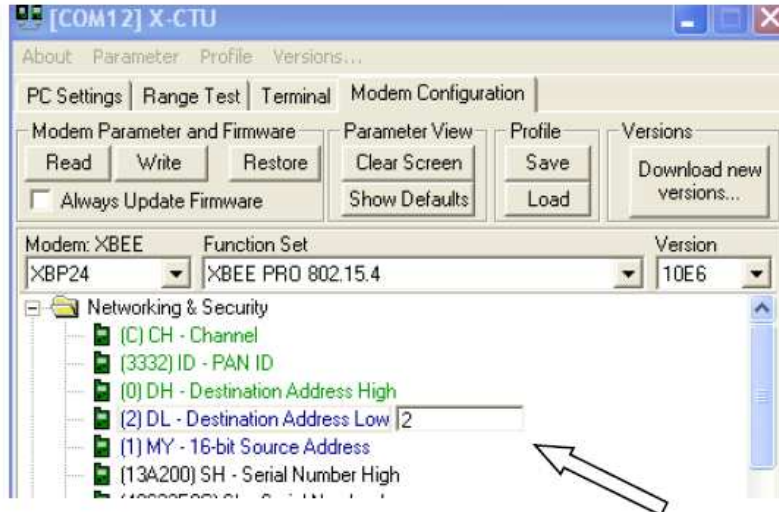


Figure V.17: L'adresse de module de destination.

V.5. Commande de drone via l'interface labview :

Afin d'ouvrir le maximum de perspective au projet, on a décidé d'étudier la possibilité de piloter le drone par le logiciel labview via le clavier. Mais il existe une méthode plus évoluée pour le faire .en effet on peut commander le drone à l'aide de la Wiimote (manette de jeu wii motion de Nintendo qui n'est rien d'autre qu'un détecteur d'accélération à 3 axes) grâce au mode accéléromètre. En effet, la gestion des accéléromètres serait parfaitement faisable.



Figure V.18: manette de jeu wiimote de Nintendo.

Il suffit de récupérer les informations de roulis et tangage qui correspondent à l'inclinaison de la manette tenue aux mains grâce au logiciel labview ; ensuite il suffit de borner ces valeurs et provoquer l'envoi de commande de déplacements suivant l'intervalle dans lequel on se situe. Cette technique est très faisable et convient très bien à notre application dans la mesure où les moteurs sont capables de réagir aussi vite que celui de la main humaine qui tient cette manette.

V.5.1. Arduino et LIFA:

LIFA signifie (Labview Interface For Arduino). C'est une extension de Labview (compatible à partir de la version 2009) qui permet de piloter une carte compatible Arduino depuis Labview.

V.5.2. Matériels et logiciels nécessaire :**a)-Matériels :**

- Un ordinateur de type compatible PC avec Microsoft Windows7 ou WindowsXP.
- Une carte compatible Arduino : dans notre cas une carte arduino UNO.
- Divers shield (carte d'extension) dans notre cas un shield bluetooth ou Xbee.dans notre cas c'est le shield Bluetooth qui est le protocole de communication de la wiimote.
- Un détecteur triaxial wiimote (manette de jeu Nintendo)ou equivalent.

b)-Logiciels :

- L'environnement de développement d'arduino
- NI LabVIEW (version 2011 ou supérieur avec les pilotes « VISA » qui gèrent la voie série.
- VI Packet Manager : logiciel qui gère l'installation des extensions de LabVIEW.
- Le paquet LIFA pour arduino à installer.la bibliothèque qui controle la wii qui est wiimotelib écrit en C.

V.5.3.Installation du firmware LIFA sur la carte Arduino :

Le firmware LIFA est disponible après installation dans le répertoire "C:\Program Files\National Instruments\LabVIEW 2011\vi.lib\LabVIEW Interface for Arduino\Firmware\LVIFA_Base" (par exemple). On obtiendra alors trois fichiers qui sont :

- Le sketch « LVIFA_Base.ino » : c'est ce sketch qu'on compile et transfère dans la platine Arduino.il sert de bootloader en quelque sorte.
- Le sketch « LabVIEWInterface.ino » :il contient l'implémentation de toutes les fonctions utilisées par LabVIEW pour dialoguer avec la carte Arduino.

- Le fichier d'inclusion «LabVIEWInterface.h»: ce fichier d'en-tête est particulièrement intéressant puisque c'est lui qui contient les "define" qu'on modifie pour adapter LIFA. Trois "define" en particulier seront à modifier suivant le projet :
- ❖ DEFAULTBAUDRATE : la vitesse de transmission de la voie série. En effet LabVIEW communique avec la carte Arduino par une voie série, par défaut la vitesse est de 115200 bauds/s qu'on pourra modifier mais pas pour ce projet.
 - ❖ COMMANDLENGTH : le nombre d'octets transmis en une seule fois par une commande LabVIEW. Par défaut cette valeur est à 15.
 - ❖ STEPPER_SUPPORT : permet ou non d'intégrer le support des moteurs pas à pas dans le firmware, n'est pas nécessaire dans notre cas.

On chargera donc "LVIFA_Base.ino" dans l'environnement Arduino, puis on le compile et on programme notre carte Arduino le comme d'habitude. Ainsi la carte Arduino est prête à fonctionner avec LabVIEW.

V.5.4. Configuration de la communication Bluetooth entre le PC et la wiimote:

1. Confirmer que le PC active la communication Bluetooth.
2. On rentre dans control panel>hardware and sound>Bluetooth device, et sous les options tab dans la section nommée "discovery." On active la case "allow Bluetooth device to find this computer."
3. Dans la même section, sur "devices". On click "add Bluetooth device wizard."
4. On active les boutons 1 et 2 de la wiimote tel que les quatre LED's commencent à flasher. Cela veut dire que la wiimote est détecté et entre en contacte avec le PC.
5. On click suivant dans "Bluetooth wizard".
6. On selectionne le materiel "Nintendo RVL-CNT-01".

7. Le PC, une fenaitre s'ouvrira et demandera "windows needs to install driver software for your Bluetooth peripheral device." On choisit "locate and install drivers."
8. Maintenant on doit voir la wiimote apparaitre avec les autres composant "tab » ; ce qui signifie que la manette est maintenant connectée au PC.

V.5.5. Calibration des données de l'accéléromètre:

1. On ouvre le " NI Example Finder" (a partir de LabVIEW 2011 dans le fichier Exemples) et on cherche le « WiiMote ». On ouvre « WiiMote Graph Accelerometer.vi ».
2. Le graphe sur l'interface frontale de labview doit afficher les déplacements d'un barre graphe de la wiimote de +2g to -2g correspondants aux déplacements de la manette.

V.5.5.1. Le code labview de graphe « Accelerometer.vi » :

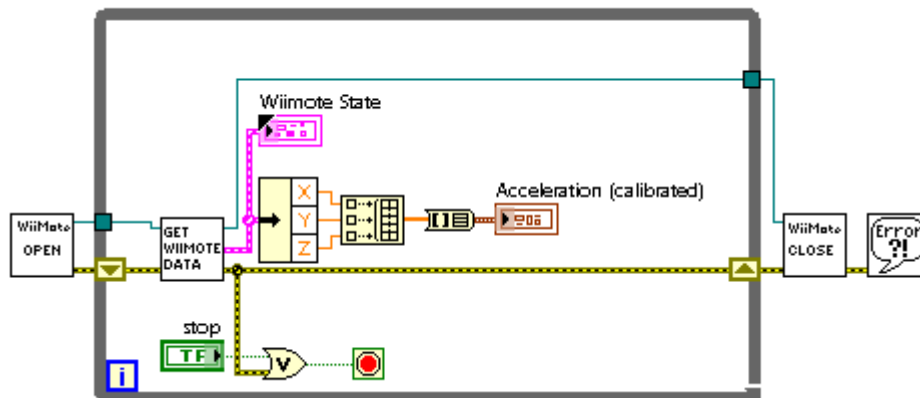


Figure V.19: diagramme (block-diagram) de l'interface virtuelle de contrôle via la wiimote.

1. "WiiMote Initialize" créer une référence à l'objet Wiimote, qui inclue l'état de tout les boutons.
2. "WiiMote Read Accelerometer" reçoit les données calibrées de détecteur accélérations tenue à la main.

3. "Wiimote Close" fait clore la référence avec les objets Wiimote et établit les rapports d'erreurs.

V.6.Conclusion :

Malgré les performances et la rapidité de la liaison des systèmes de radiocommandes basées sur les anciens protocoles tels que ppm/pcm, d'autres systèmes très évolués associant des technologies variés (communication xbee, Bluetooth, logiciel labview, la manette wiimote...) rendant les possibilités de contrôle presque illimitées.

Chapitre VI:

Partie pratique.

VI.1.Introduction :

Bien qu'une étude théorique portant sur plusieurs aspects d'un drone quadri-rotor étant menée, mais le but de projet est la réalisation pratique de celui-ci en partant des éléments les plus élémentaires réalisables, que se soit sur le plan matériel ou logiciel. Ainsi ce chapitre va traiter la réalisation des outils et composants autres que ceux vus dans les précédents chapitres.

VI.2.Fabrication des moteurs sans balais :

Le diamètre extérieur de rotor est de 42 mm et le diamètre intérieur est de 37mm, le stator lui en fer doux de diamètre extérieur égal a 32mm laissant ainsi la place aux aimants en néodymium (métaux rare de performances en terme d'attraction magnétique et de légèreté inégalées).



Figure VI.1: le rotor avec les aimants fixés a sa surface intérieure.

La taille des aimants est de 10 mm x 6 mm x 2 mm de nombre 14 par moteur .ces aimants sont collées sur la partie intérieur de rotor et éloignées de 1mm les uns des autres.



Figure VI.2: plaquettes d'aimant néodyme.

Un entre fer d'à peine 1mm est laissé entre le périmètre extérieur de stator et la surface de l'aimant, celui-ci permet au rotor de tourner librement tout en étant soumis au maximum de force magnétique exercé par ces aimants. Le stator en fer est entouré de 20 spires par bobine avec $d = 0,71$ mm de diamètre des fils ont été appliqué, l'objectif étant de permettre le maximum de courant les traversant nécessaire pour les vitesses exorbitantes attendes par ces moteurs.



Figure VI.3: stator avec enroulement en triangle.

Les rotors ainsi que le support du stator sont fabriquées en alliage d'aluminium, par usinage mécanique des pièces de révolution (tournage).

Le principe de l'usinage est d'enlever à l'aide d'une machine outil de la matière, de manière à donner à la pièce brute la forme voulue. Les pièces obtenues sont d'une grande précision.



Figure VI.4: image d'un tour universel pour l'usinage des pieces rondes.

VI.2.1 Quelques opérations réalisées :

- Dresser/charrioter de surfaces intérieures de rotor (alésage).
- perçage des trous de passage de vis porte hélice.

- alésage des capitées des emplacements roulements.
- chanfreinage des extrémités notamment celles des supports stator.



Figure VI.5: pièce en aluminium brute qui deviendra le rotor, fixé sur le mandrin d'un tour.



Figure VI.6: usinage de la surface extérieur de support-stator.



Figure VI.7: Alésage de l'emplacement de roulements

VI.2.2. Le moteur fini :

La réalisation des pièces étant terminée ; il reste l'assemblage de celles-ci. Le moteur fini pèse 120 g, avec de grandes ouvertures sur la partie supérieure de rotor en vue de permettre l'aération des bobines de stator. A noter seulement, à propos d'une propriété importante sur la conception de moteur qui est celle d'être facilement démontable en ses différentes parties et de pouvoir apporter des modifications à la structure, c'est-à-dire de remplacer les pièces endommagées au besoin.

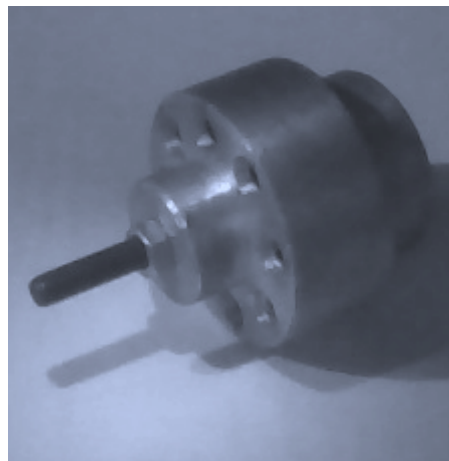


Figure VI.8: moteur sans balais finalisé.

En fin on entame le test de ce dernier en le connectant au contrôleur de vitesse. L'opération des tests ne doit pas s'effectuer avec l'hélice en place.

VI.3. Choix du microcontrôleur pour la centrale inertielle (contrôleur de vol) :

Afin de réaliser les tâches d'acquisition des tensions des capteurs, de filtrage numérique, de correction PID, et de commande des moteurs ; l'utilisation d'un microcontrôleur s'imposait. Aussi, il se devait de répondre à certains critères de performances et de fonctionnalités.

Tout d'abord dans le cas où on utilise des capteurs analogiques, alors il serait nécessaire de convertir six tensions analogiques délivrées par trois capteurs, représentant les accélérations et les vitesses de rotation dans les six axes de liberté. Cette opération requiert un convertisseur analogique/numérique doté de six canaux d'acquisition. La résolution de l'ADC n'est pas à négliger, les capteurs ayant au mieux une sensibilité de

2mV, le convertisseur doit fournir un résultat sur 12 bits. En effet avec une amplitude de 5V, cela permettra de mesurer une variation minimale de 1,22mV. Cependant notre choix est porté sur des modules de capteurs numérique communicant en bus I2C, donc le microcontrôleur doit disposer de ce type de bus.

Ensuite, les signaux seront à traiter par un filtre de Kalman, gourmand en calcul numérique. L'utilisation d'un calculateur avec une architecture 16bits est conseillée pour réaliser ces opérations. Concernant l'horloge, celle-ci doit être de 16 Mhz ou plus pour avoir une rapidité de calcul rependant au contrainte liées a la rapidité de réaction dans ce type d'appareils.

D'autres parts le quadri-rotor disposant par définition de quatre moteurs, le microcontrôleur à retenir doit avoir la capacité à les commander indépendamment. La commande de moteur s'opérant au moyen d'un signal à modulation de largeur d'impulsion (sauf dans le cas de commande a l'aide de module xbee, ou WIFI ayants des protocoles séries en sortie), le calculateur employé sera équipé de quatre modules de génération de signaux PWM, logiquement accompagnés de quatre compteurs.

Enfin, avec pour souci de faciliter le développement du contrôleur (en pouvant agir en temps réel sur les valeurs contenues en mémoire par exemple), il semble essentiel que le microcontrôleur à retenir dispose d'une liaison série afin qu'il puisse communiquer avec un poste de travail équipé d'une interface RS232 ou USB.

VI.3.1. Branchement de la liaison carte / ordinateur :

Lors de la configuration (paramétrage de PID, paramétrage de récepteur...etc.), ce branchement se fait simplement par l'intermédiaire du fil COM (RS 232 ou de la carte qu'il faut relier à un câble série ou bien une liaison USB (UART dans notre cas) toutes sont intégrées au microcontrôleur employé. Par ailleurs il existe la possibilité de rapporter et modifier ces paramètres de configuration en vol grâce aux système de communication à distance a savoir, le radio modem 2.4Ghz+codage PPM, soit le module xbee, WIFI...etc. embarqué pour les communications en vol.

VI.4.Algorithme général :

L'idée générale du programme embarqué est relativement simple. Il faut dans l'ordre :

- *L'initialisation du code*
- *La déclaration des variables entrées / sorties*
- *Initialisation des variables + variables internes*
- *L'attribution des entrées/ sorties du programme*
- *Convertir les tensions analogiques des capteurs*
- *Mixer les données par le filtre de Kalman.*
- *Calculer la commande par le correcteur PID.*
- *Mettre à jour le rapport cyclique des moteurs.*

Le filtre de Kalman et le correcteur PID devant s'exécuter à une période fixe (choisie ici à 10ms), il est indispensable de les exécuter dans une fonction d'interruption, comme précisé précédemment. Il faut ajouter à cela une période d'initialisation des capteurs au démarrage du programme, afin d'enregistrer la position horizontale.

VI.4.1.Le programme de stabilisation :

Nous allons décrire dans cette partie le code utilisé pour le quadri rotor, morceau par morceau afin de bien l'appréhender dans son ensemble et surtout de voir ce qui peut être modifié dans le but d'une éventuelle amélioration du programme.

Ce code est en langage arduino utilisant le C, et peut donc être lu à l'aide d'un éditeur Arduino IDE destiné spécialement au matériel AVR. En effet il peut être adapté sans modifications majeurs à n'importe quel microcontrôleur supporté par Arduino.

VI.4.2.Explication de code source :

Comme tout les programmes arduino, il ya les deux routines «setup» et «main». Setup initialise toutes les variables et la boucle « main » performe l'acquisition des entrées, effectue les calculs puis active les sorties. L'acquisition des signaux PWM

issues de récepteur est réalisée grâce au "pin change interrupts". Lorsque la tension a l'entrée change, la routine d'interruption est ainsi appelée et le calcul de délais commence. La durée de l'impulsion est calculée à partir des instants où se sont produits les interruptions. Les signaux PWM en sorties des contrôleurs de vitesse est effectuée à l'aide de plusieurs TIMER's associées aux comparateurs, ainsi le contrôleur envoie les impulsions en sortie exactement à l'instant prévu, pendant le temps mort la sortie conserve le niveau de tension haut ou bas. Les détecteurs communiquent avec le contrôleur à l'aide de bus I2C, qui transmet des données synchrones et qui est un excellent support pour de multiples dispositifs communicants via uniquement deux fils.

Le programme principale « main » est divisée en plusieurs tâches, chacune est exécutée à une fréquence différente ; ce qui donne à chaque tâche une priorité. Le programme qui s'occupe de la stabilisation de l'appareil reçoit la plus grande priorité. Alors que la tâche s'occupant de la communication par exemple est moins prioritaire.

VI.4.3.Organigramme de gestion de stabilisation :

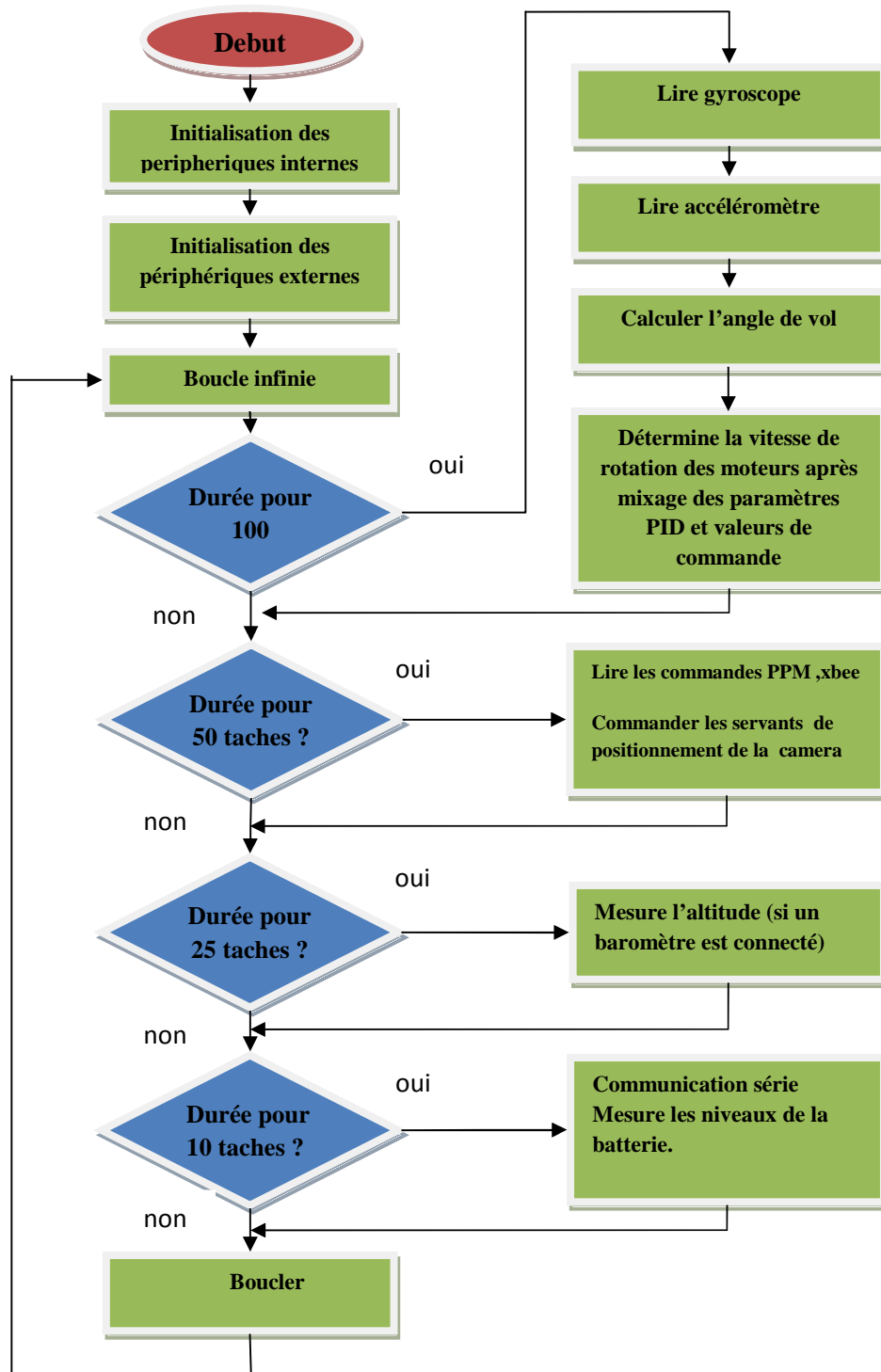


Figure VI.9: Sous programme qui effectue les fonctions de stabilisation de drone.

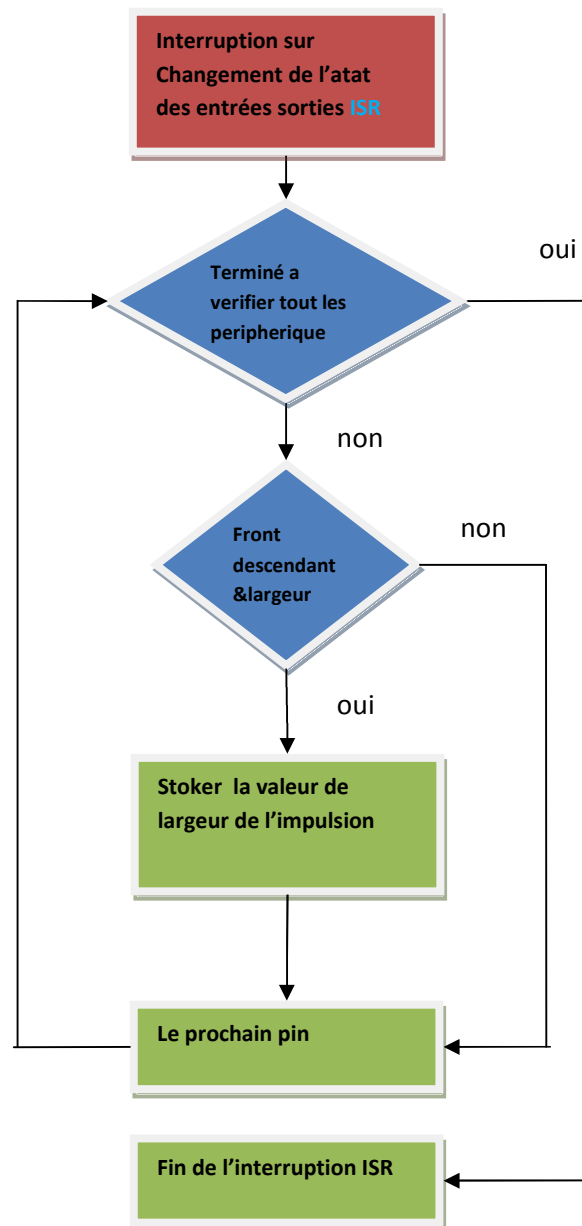
VI.4.4. Programme d'interruption :

Figure VI.10: la routine d'interruption gérant les communications.

VI.4.5. Routine d'initialisation :

```
void setup()  
{  
    DDRC = 0; // pin en entrée
```

```
//active PCINT 18 a 23
PCICR |= (1 << PCIE2);
PCMSK2 = 0xFC;

Serial.begin(115200) ; //débit de transmission série préconfiguré
}

typedef struct {           //structure qui gère les variables employées
    byte edge;
    unsigned long riseTime;
    unsigned long fallTime;
    unsigned int lastGoodWidth;
} tPinTimingData;
volatile static tPinTimingData pinData[6 + 1];
volatile static uint8_t PCintLast;

ISR(PCINT2_vect)
{
    uint8_t bit;
    uint8_t curr;
    uint8_t mask;
    uint32_t currentTime;
    uint32_t time;

    // recherché l'état des pattes
    curr = PINC & 0xFC;
    mask = curr ^ PCintLast;
    PCintLast = curr;

    currentTime = micros();
```



```

Serial.print(pinData[i].lastGoodWidth);
Serial.print(", ");
}

delay(500);
}

```

VI.4.6. Signal de commande des contrôleurs de vitesse PWM :

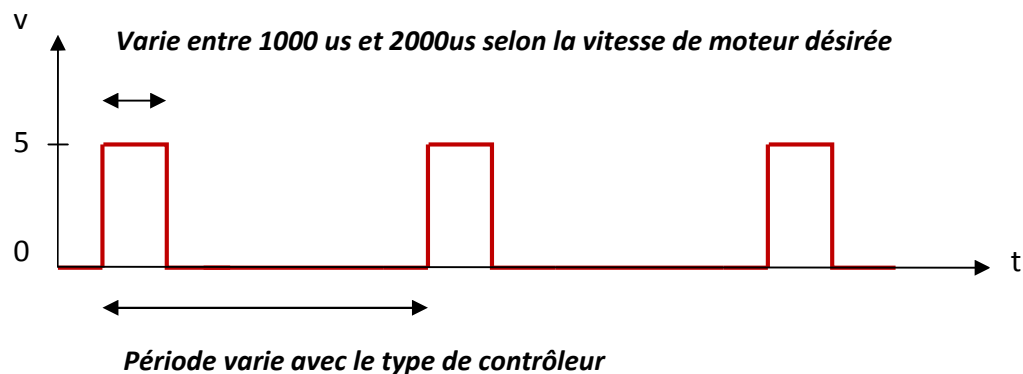


Figure VI.11: signaux de commande des moteurs issus de contrôleur principale.

Les fonctions arduino qui s'occupe de génération des signaux PWM de tel sorte que le contrôleur de vitesse connaîtra a tout moment le niveau auquel il doit pousser le moteur .La fonctionnalité « output -compare » de timer est mise a profit en vue de produire les signaux en question .lorsque le timer atteint 0, il active la sortie a l'état HAUT plus tard lorsqu'il atteint une certaine valeur (qui est spécifié..) il inverse la pin. Ce qui au final réalise une onde carrée avec un rapport cyclique variable .

Ainsi ,les signaux de commande des servos a usuellement une période de 20ms donc un fréquence de 50Hz.la sortie de contrôleur a une période plus petite donc un fréquence plus faible entre 250 a 300Hz .notre contrôleurs de vitesse peuvent êtres alors programmés pour accepter des fréquences plus grandes, bénéfiques pour une maniabilité de l'appareil élevé (donc une réaction rapide aux commandes).La conséquence sur la stabilité est aussi importante.

Ci-dessous le code qui traite le décodage des signaux pwm reçus :

```

#define PWM_FREQUENCY 300 // en Hz
#define PWM_PRESCALER 8
#define PWM_COUNTER_PERIOD
(F_CPU/PWM_PRESCALER/PWM_FREQUENCY)

void setup()
{
  // pins en sorties
  DDRD |= (1 << 4) | (1 << 5) | (1 << 6) | (1 << 7);

  // par défaut 1000 microsecondes de largeur d'impulsion.

  OCR1A = 1000 * 2 ;
  OCR1B = 1000 * 2 ;
  OCR2A = 1000 / 16 ;
  OCR2B = 1000 / 16 ;

  // tla config est:
  // mise à 0 de OCnA/OCnB si équivalent, mise à 1 de OCnA/OCnB si BOTTOM
  (mode non inversé).

  // config de timer 1
  TCCR1A = (1<<WGM11)/(1<<COM1A1)/(1<<COM1B1);
  TCCR1B = (1<<WGM13)/(1<<WGM12)/(1<<CS11);
  ICR1 = PWM_COUNTER_PERIOD;

  // setup timer 2
  TCCR2A = (1<<WGM20)/(1<<WGM21)/(1<<COM2A1)/(1<<COM2B1);
  TCCR2B = (1<<CS22)/(1<<CS21);
  // periode fixé pour timer 2, environ 244 Hz

  // Anoter que timer1 est de 16-bit et timer2 est un 8-bit .

```

```
}

void loop()
{
  int pw;
  for (pw = 1000; pw <= 2000; pw += 20)
  {
    OCR1A = pw * 2 ;
    OCR1B = pw * 2 ;
    OCR2A = pw / 16 ;
    OCR2B = pw / 16 ;

    delay(10);
  }

  for (pw = 2000; pw >= 1000; pw -= 20)
  {
    OCR1A = pw * 2 ;
    OCR1B = pw * 2 ;
    OCR2A = pw / 16 ;
    OCR2B = pw / 16 ;

    delay(10);
  }
}
```

VI.4.7. Acquisition des données des capteurs :

Cette étape contient le sketch arduino qui s'occupe de lecture des données des détecteurs (accéléromètre BMA180 et le gyro ITG-3200).

Ces deux détecteurs communiquent avec le microcontrôleur à l'aide de bus I2C ce type de bus permet à plusieurs capteurs d'être reliés parallèlement via seulement

deux fils donc un gain important en terme de nombre de pattes nécessaires du contrôleur. Sans oublier également sa vitesse de transmission qui est très élevée 100Khz à 400Khz.

La détection des accélérations linéaires et des vitesses angulaires se fait sur trois axes orthonormés. Donc toutes les valeurs issues des capteurs sont partagées suivent ces trois axes (x, y, z) dont la résolution est de 16 bits pour chaque'un des axes.

Le contrôleur de vole principale (ou centrale inertielle) va initialiser l'accéléromètre en le mettant juste en mode reset. Puis après il effectua un filtrage de 10Hz et programme le lecteur à +/-2G. La valeur lue à partir de l'adresse «0x20» des registres de capteur étant stockée dans un registre spécifique «0x40» dans la mémoire flash du contrôleur.

A noter que le code est réalisée en ayant consulté le catalogue de BMA 180 qui explique la structure de procédé de communication I2C et toutes les informations nécessaires a l'acquisition des données qu'il envoi et/ou reçoit. Aussi le bma180 contient un détecteur de température qui ne sera pas utilisé dans notre cas.

De même que pour l'accéléromètre, le programme de contrôleur va initialiser le gyro (détecteur de la vitesse angulaire.) puis active son filtre passe bas a 10Hz .il lui ordonne d'utiliser son oscillateur interne intégré .la lecture des données peuvent êtres réalisé après avoir spécifié les adresse de maitre et d'esclave comme pour l'accéléromètre vu en haut.

le code permettant l'acquisition des données des précédents capteurs est utilise la bibliothèque <Wire.h> de logiciel Arduino_IDE:

```
#include <Wire.h>
```

```
void setup()
```

```
{
```

```
  Serial.begin(115200); //débit de la liaison série.
```

```
  Wire.begin();
```

```
Serial.println("initialiser les détecteurs");

AccelerometerInit();
GyroInit();

Serial.println("les détecteurs sont initialisés");
}

void AccelerometerInit()
{
  Wire.beginTransmission(0x40); // Adresse de l'accéléromètre
  // reset the accelerometer
  Wire.send(0x10);
  Wire.send(0xB6);
  Wire.endTransmission();
  delay(10);

  Wire.beginTransmission(0x40);
  // config de largeur de bande de filtre passe bas
  Wire.send(0x0D);
  Wire.send(0x10);
  Wire.endTransmission();

  Wire.beginTransmission(0x40);

  Wire.send(0x20); // lire a partir d'ici
  Wire.endTransmission();
  Wire.requestFrom(0x40, 1);
  byte data = Wire.receive();
  Wire.beginTransmission(0x40);
  Wire.send(0x20);
  Wire.send(data & 0x0F); // filtre passe bas 10 Hz
```

```
Wire.endTransmission();

Wire.beginTransmission(0x40);
Wire.send(0x35); // lire a partir d'ici
Wire.endTransmission();
Wire.requestFrom(0x40, 1);
data = Wire.receive();
Wire.beginTransmission(0x40);
Wire.send(0x35);
Wire.send((data & 0xF1) | 0x04); // amplitude +/- 2g
Wire.endTransmission();
}

void AccelerometerRead()
{
Wire.beginTransmission(0x40);
Wire.send(0x02); // etablire le pointeur vers la donnée à lire.
Wire.endTransmission();
Wire.requestFrom(0x40, 6); // lire les données à 3 axes, chaqu'un //de 16 bits
// afficher les données
Serial.print("Accelerometer: X = ");
short data = Wire.receive();
data += Wire.receive() << 8;
Serial.print(data);
Serial.print(" , Y = ");
data = Wire.receive();
data += Wire.receive() << 8;
Serial.print(data);
Serial.print(" , Z = ");
data = Wire.receive();
data += Wire.receive() << 8;
```

```
Serial.print(data);
Serial.println();
}

void GyroInit()
{
  Wire.beginTransmission(0x69); // Adresse de gyro
  // reset the gyro
  Wire.send(0x3E);
  Wire.send(0x80);
  Wire.endTransmission();

  Wire.beginTransmission(0x69);
  // low pass filter 10 Hz
  Wire.send(0x16);
  Wire.send(0x1D);
  Wire.endTransmission();

  Wire.beginTransmission(0x69);
  // use internal oscillator
  Wire.send(0x3E);
  Wire.send(0x01);
  Wire.endTransmission();
}

void GyroRead()
{
  Wire.beginTransmission(0x69);
  Wire.send(0x1D); // etablir le pointeur pour lire la donnée.
  Wire.endTransmission();
```

```
Wire.requestFrom(0x69, 6);

// lire 3 axes de données ,affichez les.
short data = Wire.receive() << 8;
data += Wire.receive();
Serial.print("Gyro: X = ");
Serial.print(data);
Serial.print(" , Y = ");
data = Wire.receive() << 8;
data += Wire.receive();
Serial.print(data);
Serial.print(" , Z = ");
data = Wire.receive() << 8;
data += Wire.receive();
Serial.print(data);
Serial.println();
}

void loop()
{
  AccelerometerRead();
  GyroRead();

  delay(500); // attendre l'accusé de réception de la part de l'esclave //(detecteur)
}
```

La figure suivante montre Control Principale de stabilisation mis en avant dans le programme précédent:

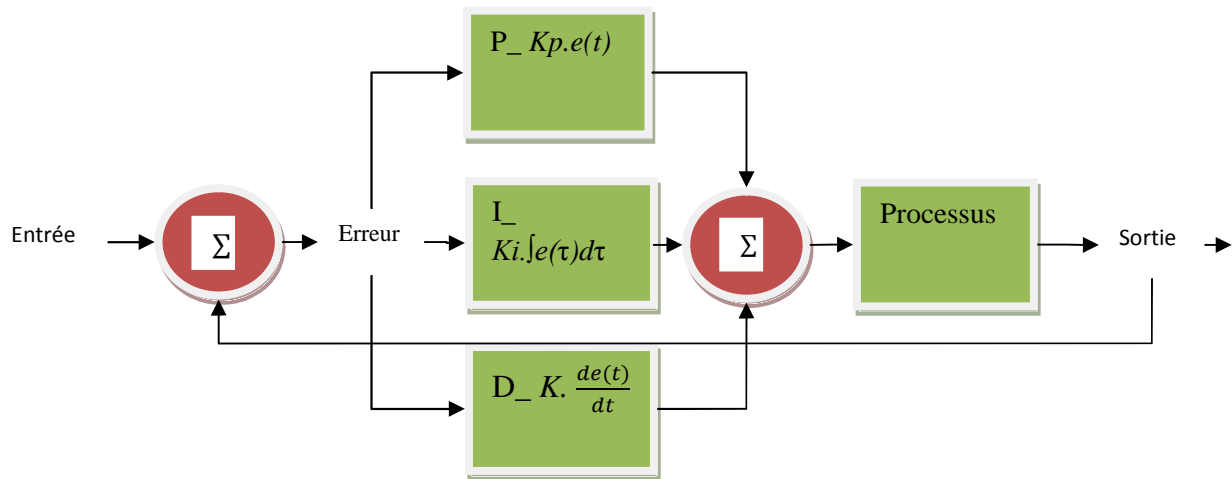


Figure VI.12: Fonction de stabilisation en boucle fermée.

VI.5. Les modes de pilotage :

Le code nous permet voler selon deux modes différents :

- a) Le premier est lorsque l'accéléromètre est ignoré, ainsi on contrôle uniquement la vitesse angulaire de drone à l'aide de la manette des commande (joystick ou autre.) ce mode est appelé le mode acrobatique. Dans ce mode le la fonction de code est de déplacer le drone dès que l'ordre lui est reçu. et arrêter aussitôt que l'ordre est cessé. L'avantage de ce mode de pouvoir effectuer des cabrioles et d'autres mouvements acrobatiques semblables.
- b) Le deuxième mode est celui où l'accéléromètre est utilisé quote à quote avec le détecteur gyroscopique pour calculer l'angle instantané de drone par rapport à l'horizon. Dans ce mode, si on pose la manette de commande ou bien lorsque le signal est brouillé ou qu'il n'atteint pas l'appareil, le drone revient automatiquement à la position horizontale et garde sa position angulaire telle quelle. Ce mode est appelé le «mode stable», "stable" mode ou "attitude" mode en anglais. Ce mode ne permet pas de cabrioles.

Pour calculer l'angle courante de l'appareil, la vitesse angulaire est lue grâce au capteur gyro .la vitesse angulaire multiplié par le temps nous donne l'angle de rotation pendant cette même période de temps .il en résulte un grand nombre d'erreurs durant

ce processus a cause de bruit dans le gyro. Pour corriger cette erreur, l'accéléromètre est utilisé pour mesurer la gravité qui sera prise en compte dans les calculs.

Le contrôleur de vole ajoute ses informations avec ce de contrôleur PID (proportionnel, intégral, dérivateur.) pour contrôler les moteurs .si on utilise notre contrôleur PID pour faire pivoter le drone d'un angle vers un autre en effectuent la différence des vitesses ainsi, par ajustement des valeurs des terme PID, le contrôleur pourra réaliser en conséquence de grand ou petit écarts de vitesse en concordance avec ces paramètres.

VI.6.Réalisation des différentes la cartes électroniques :

L'ensemble de l'électronique embarquée est découpé en trois parties :

- Une carte avec le microcontrôleur et l'alimentation
- Une carte où sont situés les trois capteurs
- Un montage de puissance situé au plus près de chaque moteur (contrôleur de vitesse).

VI.6.1.Carte du contrôleur à base d'Atmega328 :

Le microcontrôleur Atmega328 ne nécessite pas un câblage particulièrement complexe pour l'alimenter. Il dispose de trois pins à relier à la masse et trois autres à relier au 5V. L'alimentation du convertisseur analogique/numérique représentée par les pattes VSS et VDD sont reliées elles aussi à la masse et au 5V respectivement. Un condensateur découple l'ensemble de ces quatre entrées d'alimentation. Ensuite, l'oscillateur du microcontrôleur est cadencé par le quartz à 16MHz associé à deux condensateurs de 22pF, comme recommandé dans le catalogue. Sans oublier la connexion de la broche de reset qui se connecte au 5V par l'intermédiaire d'une résistance de 10K pour limiter le courant à l'allumage.

La carte du microcontrôleur permet également de générer les trois tensions nécessaires pour alimenter l'ensemble des composants: 3,3V, 5V et 12V. Les composants à logique TTL (microcontrôleur, drivers...) requièrent une tension de 5V,

les capteurs fonctionnent à 3,3V, et les moteurs tournent à une tension nominale de 12V. Il a été retenu le choix d'utiliser des batteries de 11.1V qui polariseront directement les contrôleurs de vitesse, tandis que les tensions de 3,3V et 5V seront obtenues par des régulateurs.

Le catalogue de ces régulateurs préconise la mise en place de condensateurs de filtrage de 10 μ F de part et d'autre de chaque régulateur. De plus le courant nécessité par les composants logiques n'étant pas particulièrement élevé, la présence de dissipateurs thermiques n'est pas obligatoire.

VI.6.2. Commande des moteurs par MOSFETs :

Pour pouvoir stabiliser en assiette le quadri-rotor, nous devons contrôler indépendamment la vitesse des 4 hélices, et donc des 4 moteurs à courant continu sans balais. Cependant, la commande des moteurs directement à partir du microcontrôleur de vitesse «atmega 8» est impossible car le courant tiré par un moteur quel qu'il soit, est bien supérieur au courant admis par une patte d'un microcontrôleur, qui est de 25mA. Pour remédier à cela, on fait appel à une interface de puissance.

Le transistor bipolaire est largement utilisé en électronique pour commander un circuit à partir d'un composant faible puissance. Cependant, un transistor bipolaire est commandé assez fortement en courant, et étant donné leur gain moyen, les 25mA disponibles en entrée vont être atteints rapidement si on désire quelques ampères en commutation. Le deuxième inconvénient des transistors bipolaires est leur dissipation excessive de chaleur, nécessitant alors des refroidisseurs imposants.

Le MOSFET (Metal-Oxide-Semiconductor Field-Effect Transistor) est le transistor convenant à nos besoins, car contrairement au bipolaire, il ne nécessite quasiment aucun courant de base pour commuter et laisse passer plusieurs ampères. Voyons plus en détails ses principales caractéristiques :

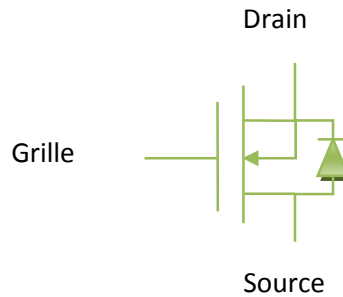


Figure VI.13: Représentation électrique d'un transistor mosfet de puissance avec sa diode de protection.

- $R_{DS(ON)}$: Résistance entre la source et le drain lorsque le MOSFET est ON (interrupteur fermé).
- $I_{D(MAX)}$: Intensité maximale que le transistor peut laisser passer de la source au drain.
- P_D : Puissance dissipée maximale que le transistor peut supporter.
- $V_{GS(TH)}$: Tension à appliquer en entrée du MOSFET pour pouvoir le commuter.
- C_{ISS} : Capacitance entre la grille et l'ensemble source-drain.

La puissance dissipée est le premier élément à prendre en compte pour le choix du MOSFET, car cela va déterminer si un dissipateur sera nécessaire. Pour cela, il faut savoir qu'un MOSFET fonctionnant en commutation dissipe plus lors de transition bas vers haut et inversement. En effet, lorsque l'interrupteur est fermé (MOSFET ON), V_{DS} est très faible et I_{DS} est élevé, et lorsque l'interrupteur est ouvert (MOSFET OFF), V_{DS} est élevé et I_{DS} est très faible. Ce n'est donc pas ces deux cas que le MOSFET va dissiper de la chaleur. En revanche, lors du passage de l'état ON à l'état OFF, on peut calculer :

$$P = I^2 \times R_{DS(ON)} = I \times U$$

Cependant, lors du passage d'un état à l'autre, le courant et la tension aux bornes du MOSFET n'est pas maximale, elle est à peu près divisée par 2. On aurait plutôt :

$$P_{commutation} = \left(\frac{P}{2}\right) \times \left(\frac{U}{2}\right)$$

Si l'on continue dans ce raisonnement, la dissipation moyenne va dépendre du temps relatif de commutation par rapport au temps où le MOSFET reste dans un état. Il suffit d'additionner les temps de montée et de descente du MOSFET, et de diviser le tout par la période utilisée pour commuter le MOSFET. La formule devient alors :

$$P_{Moyenne} = \frac{\left(\frac{I}{2}\right) \times \left(\frac{U}{2}\right) \times (T_{DON} + t_R + t_{Roff} + t_F + t_C)}{\left(\frac{T}{2}\right)} ; \text{ Avec P en Watt.}$$

Un autre point à prendre en compte pour choisir un MOSFET est la tension appliquée en entrée VGS qui doit se situer en général entre 5 et 10 volts de plus que la tension VGS(TH). De plus, CISS entre en jeu ici, car avant d'atteindre la tension désirée, il va falloir charger cette capacité. Cette charge nécessite alors un certain courant, qu'il vaut mieux éviter d'amener directement du microcontrôleur. Le pilote de MOSFETs va jouer ce rôle. Ce driver est simplement un buffer à plusieurs entrées et sorties schématiser ci-après :

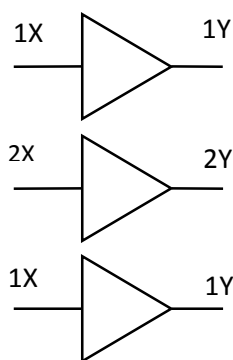


Figure VI.14: Schéma montrant plusieurs buffers en parallèles.

La sortie étant en collecteur ouvert, il suffit de tirer une résistance en sortie du composant, à la tension désirée, pour avoir une commande identique au signal d'entrée, mais de tension plus élevée. Enfin, c'est le driver qui va fournir le courant demandé par le MOSFET, donc le microcontrôleur se trouve protégé.

VI.6.3. Les tensions PWM de commande des moteurs:

La PWM (Pulse Width Modulation) ou MLI (Modulation de Largeur d'Impulsion) en français, est une méthode qui comme son nom l'indique consiste à faire varier un signal électrique non pas en amplitude ni en fréquence, mais en largeur d'impulsion. Explicitons ceci à l'aide d'un schéma :

Les signaux ont une fréquence de 20kHz soit une période de 50µs. Les temps T_{haut} et T_{bas} sont les temps que l'on fait varier pour réaliser cette modulation. Par convention, on n'utilise que le temps T_{haut} pour définir le rapport cyclique, ou Duty Cycle (D.C.) en anglais, qui est donné par la formule suivante :

$$\text{Rapport Cyclique} = \frac{T_{Haut}}{T} \times 100$$

Dans cet exemple, le premier signal a un D.C. égal à 50%, le second à 75%, et à 25% pour le troisième. L'intérêt de cette technique est évident pour la commande d'un moteur à travers un MOSFET. En effet, la PWM envoyée au transistor va faire varier la tension moyenne aux bornes du moteur à courant continu. Si on utilise un moteur de 5V par exemple, un cycle de travail de 0% arrêtera le moteur (0V), tandis qu'un cycle de 50% fera tourner le moteur à la moitié de sa puissance maximale (2,5V), et qu'un cycle de 100% le fera tourner à plein régime (5V).

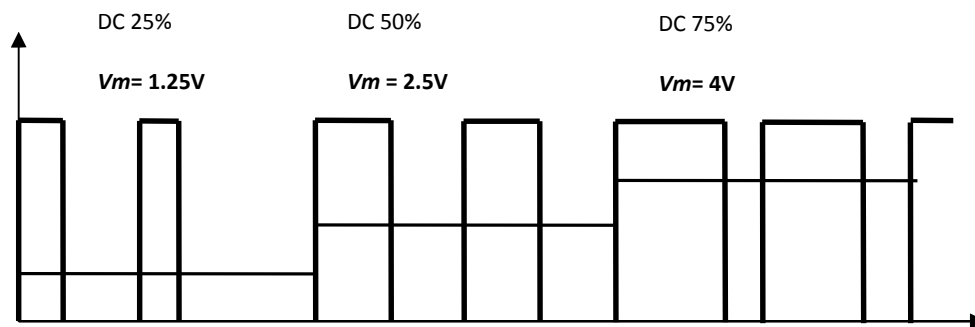


Figure VI.15: le rapport entre le rapport cyclique de signal PWM et la tension moyenne de sortie.

La fréquence de la PWM est choisie de telle manière à éviter la surcharge de bruit lors du fonctionnement de l'appareil. L'oreille humaine peut idéalement capter les sons de 50Hz à 20kHz. En dessous de la barre des 50Hz, la résolution de commande devient trop faible, on ne peut commander correctement le moteur. Avec une fréquence de 20kHz, le sifflement provoqué par les frottements internes du moteur sont quasi inaudibles pour un humain. Aussi il faut garder à l'esprit que le moteur peut être apparenté à une charge RL, impliquant des temps de charge et de décharge non négligeables; donc plus la fréquence sera élevée, et moins le courant oscillera autour de sa valeur moyenne. Cependant, il ne faut pas augmenter trop fortement la fréquence de la PWM, car la puissance dissipée par le MOSFET dépend du ratio entre le temps de commutation et la fréquence de la PWM, ainsi plus cette dernière sera élevée, plus le MOSFET chauffera.

VI.6.4. Carte des capteurs :

Afin de permettre leur remplacement plus aisé, il est possible de placer les capteurs sur une carte à part entière. Les composants situés sur cette carte sont des plus réduits; uniquement les capteurs y seront soudés, ainsi qu'une barrette, afin d'y placer un connecteur pour transmettre les tensions des capteurs vers la carte du contrôle et amener l'alimentation aux composants. Mais dans le souci de diminuer le poids et l'encombrement et dans la mesure où les capteurs actuels sont déjà placés sur carte

intégré ; alors cela sont seulement disposés convenablement « parallèlement » sur la carte mère principale en respectant les axes X, Y et Z de l'appareil.

VI.6.5.1. Catre de puissance :

Les moteurs consommant un courant très élevé allant jusqu'à 25A chaqu'un, soit 100A pour les quarts moteurs. Alors le montage doit permettre de fournir le courant nécessaire à leur fonctionnement. 56 Mosfets IRFB3507 en tous sont utilisés pour amplifier les courants alimentant ces quatre moteurs. Les Mosfet ont été choisis car ils étaient entre autre disponibles en magasin. Ils sont un peu surdimensionnés car, si on regarde d'un peu plus près le catalogue, ils peuvent supporter 100A chacun.

On a :

- ❖ $R_{DS(ON)} = 15\text{mA}\Omega$.
- ❖ $I_{D(MAX)} = 97\text{A}$ (en continu), 120A (en pulsations) à 25°C .
- ❖ $P_D = 190\text{W}$.
- ❖ $V_{GS(TH)} = xx\text{V}$ pour $xx\ \mu\text{A}$.
- ❖ Boitier TO-220AB.

Dans les applications commerciales ce sont des composants CMS qui sont employées, y compris les Mosfets. De ce fait le poids des 56 Mosfet serait largement diminué.

VI.6.6.1. Schéma de la carte de control de vole :

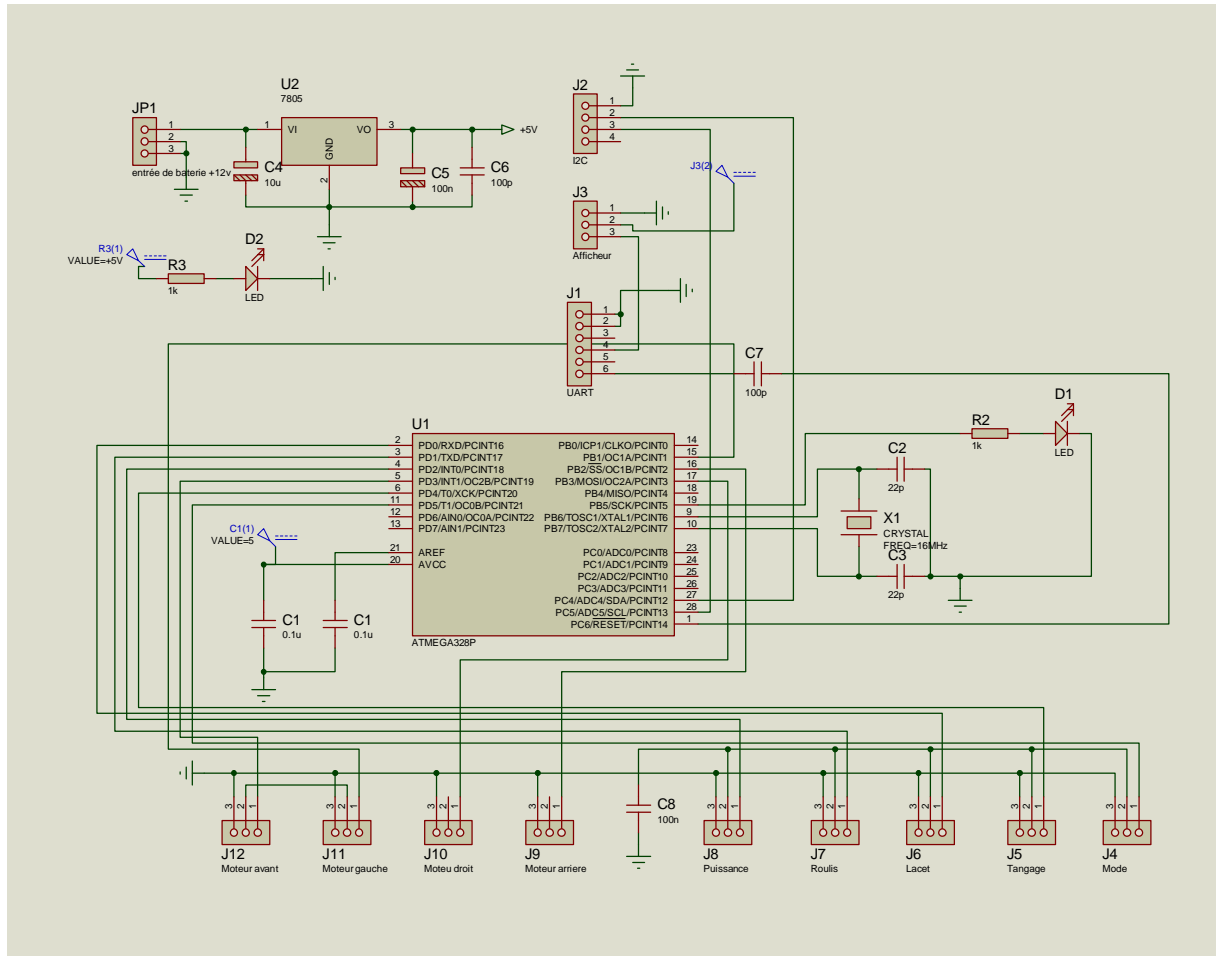


Figure VI.16: schéma électrique de la carte constituant la centrale inertielle (contrôleur de vol).

VI.6.6.2. schéma électrique de la carte de control de vitesse :

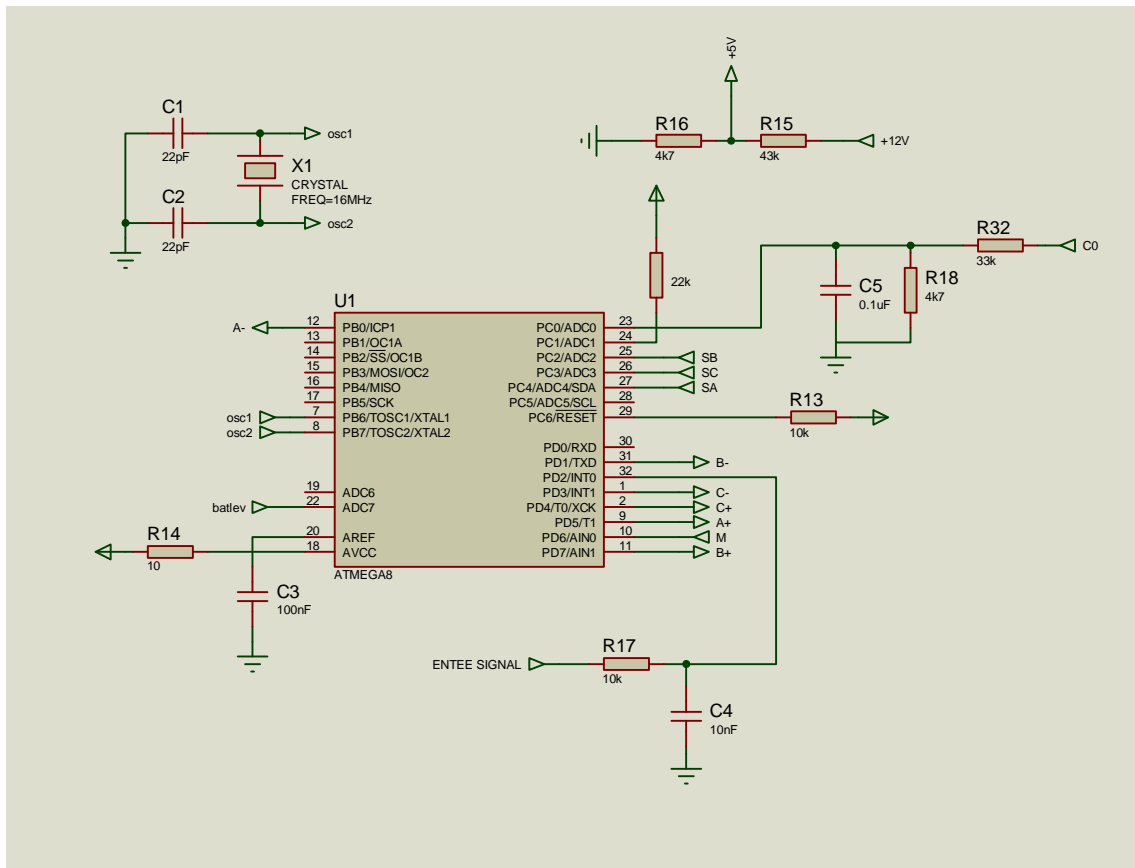


Figure VI.17: schéma électrique de cœur de contrôleur de vitesse.

VI.6.7. Le module de puissance :

La nouveauté de ce type de montage de puissance par rapport à la plupart des contrôleurs de commerce est l'utilisation de mosfet uniquement de type NMOS dans les compartiments demi ponts attaquant les moteurs. Cette configuration permet de gagner considérablement en puissance et en prix. En effet les transistors PMOS sont moins performants et plus coûteux.

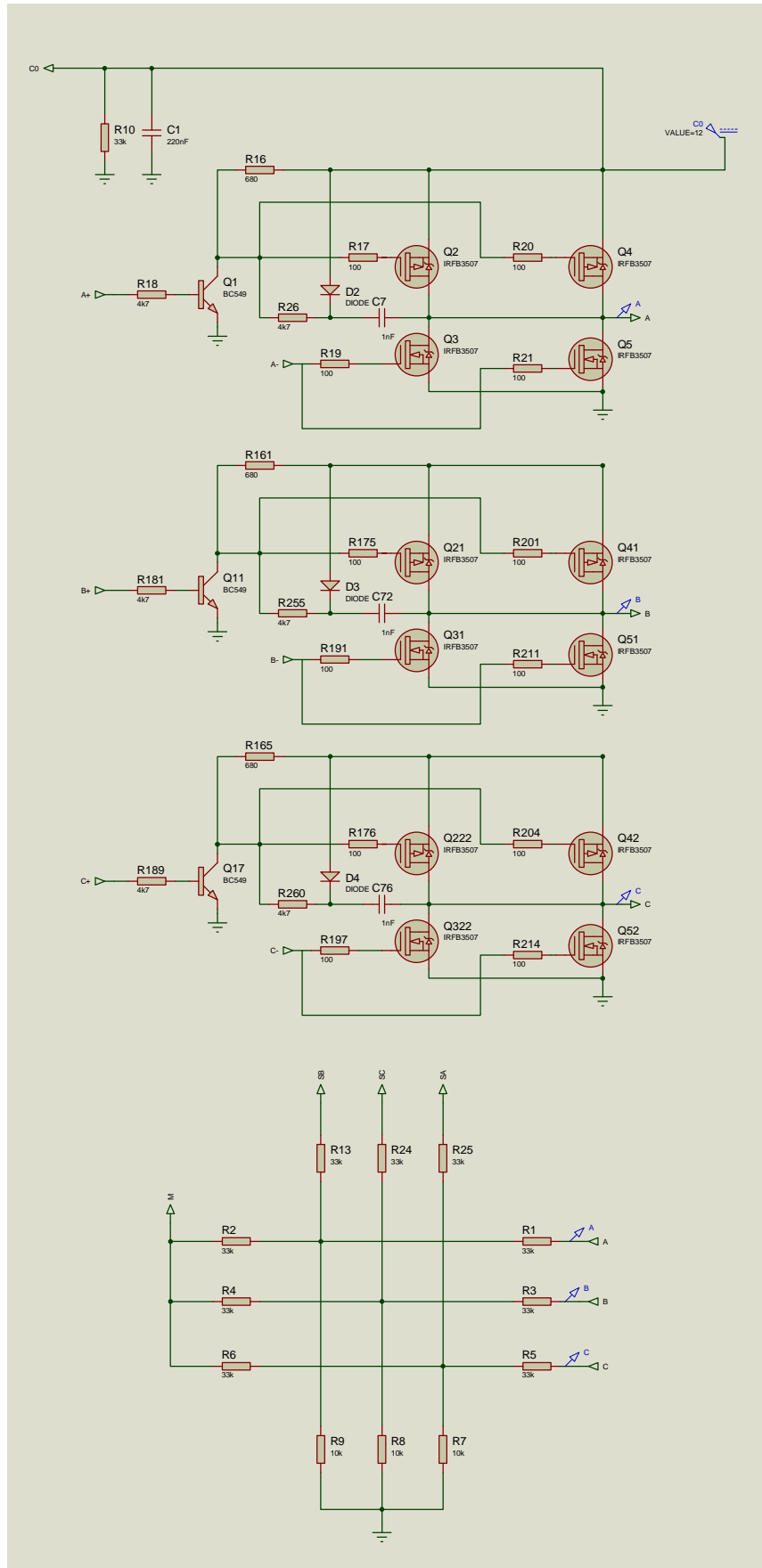


Figure VI.18: schéma électrique de la partie de puissance de contrôleur de vitesse.

VI.6.8.Routage :

La suite de logiciels ARES de proteus répond parfaitement aux besoins de conception du projet. Elle est composée d'un éditeur de schémas pour réaliser graphiquement les circuits et d'un éditeur de circuits imprimés « Layout » couplé à un routeur automatique.

Après un placement précis des composants et la définition des contraintes en taille des cartes, le routeur procède à la réalisation de circuit imprimé.

Ci-dessous, la représentation de la carte du contrôleur :

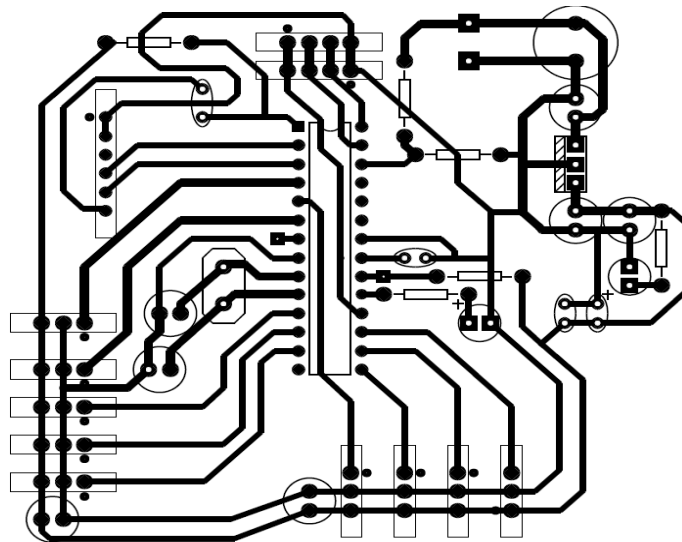


Figure VI.19: schema d'implantation des composants de controleur de vol sur ARES.

De même l'implantation des composants pour le contrôleur électronique de vitesse sur ARES est représentée ci après :

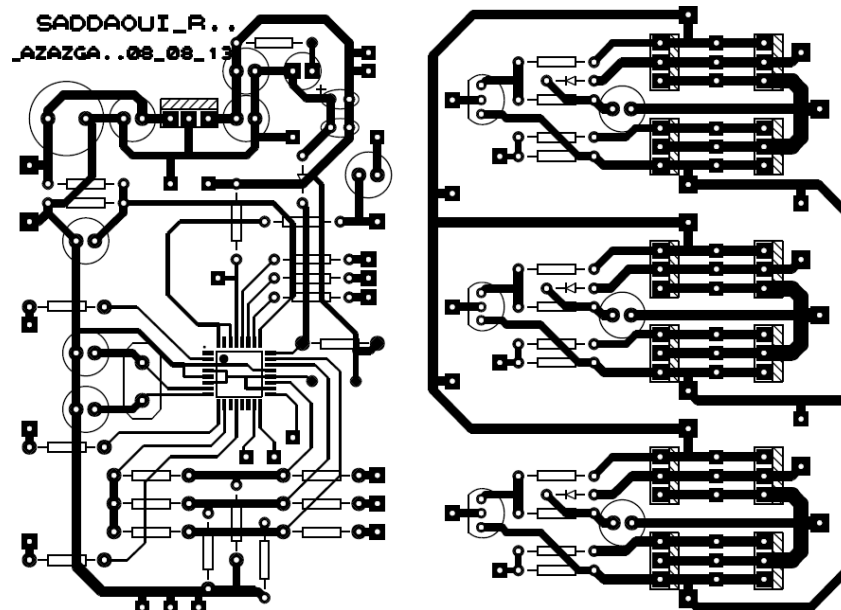


Figure VI.20: typons et emplacement des composants sur la carte de contrôleur de vitesse.

VI.6.9. Le châssis :

Le châssis est le squelette du drone. Il peut prendre n'importe quelle forme. La distance entre les moteurs peut varier entre 35 et 60cm, suivant la taille des moteurs et des hélices. On utilise un cadre standard de 50cm en profilés carrés en aluminium. 50cm est la distance entre le centre de deux moteurs montés face à face, ce qui veut donc dire que les bras en aluminium sont un peu plus longs (environ 52cm).

VI.6.10. Conception de la structure:

La structure de quadri-rotor est une réalisation à base de tubes de section carrée en aluminium. Ce choix est dû d'une part à sa facilité de fabrication et d'autre part pour sa grande légèreté qui permettra de gagner en masse tout en conservant la rigidité en flexion du drone. Le centre du croisillon a été étudié afin de pouvoir aménager deux espaces pouvant contenir la carte de contrôleur principale et les batteries. Ainsi, nous avons prévu deux compartiments : l'un au niveau des moteurs et l'autre en dessous.

Nous avons choisi de fixer tout le système de commande (récepteur + batterie récepteur) dans la partie inférieure dans le but d'abaisser légèrement le centre de gravité. En effet, un tel abaissement va permettre au drone d'avoir un meilleur retour à

sa position de stabilité horizontale, sans pour autant l'empêcher de se pencher autour de ses axes et de maintenir cette position.

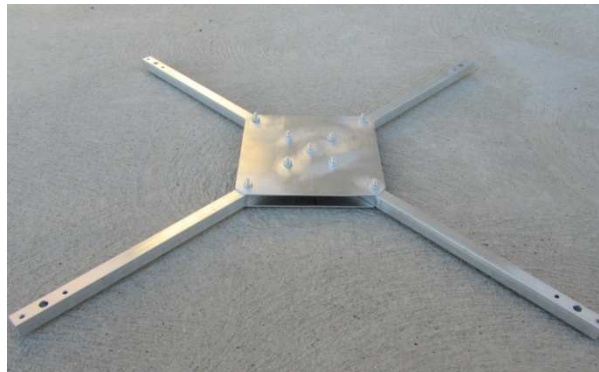


Figure VI.21: Photographie de châssis en croix.

Cette conception montre une très grande résistance en flexion et en torsion, ce qui était le but principal afin de résister aux forces des moteurs perpendiculaires à la structure. Le résultat étant une structure (châssis) ayant un poids total d'environ 1Kg.

VI.6.10.1. Avantages de cette conception :

Cette conception a l'avantage majeur d'être très rigide tout en étant relativement légère (400g de structure ce qui fait plus de 200g avec les supports moteurs).

La rigidité en flexion offerte par les barres en aluminium est très bonne et la torsion est quasiment inexistante grâce à la fixation par plusieurs boulons.

La structure avec en particulier le support avionique supérieur est de plus très modulaire. En effet, la plaque supérieure en contre-plaqué sur lequel l'avionique peut être attachée est amovible et tient en place dans le drone par simple emboîtement. Ainsi, il est possible de travailler sur la carte de contrôleur de vol ou sur le système gyroscopes/mixeurs en dehors du drone, ce qui constitue un gros avantage.

La construction va débuter par la réalisation d'un châssis sur lequel viendront se monter les moteurs, les régulateurs de vitesse, les gyroscopes, le récepteur radio et les accumulateurs.

Les deux profilés en aluminium proviennent du magasin comme on peut les extraire des vieux antennes UHF de TV. Le profilé anodisé, tube carré, fait 10 x 10 x 1, et sa longueur de 100 cm non définitive peut être ajustée par la suite. Il va constituer l'ensemble des bras 1 et 4 de la croix formant le châssis.

Le second et le troisième profilé sont découpés de façon à avoir 1cm de moins de longueur que ceux des 2 premiers pour former un périmètre carré après l'assemblage finale.

L'image suivante montre les bras constituant les châssis ainsi que le support moteur.



Figure VI.22: image de l'emplacement moteur à l'extrémité des bras.

Le support moteur est la partie où viendra se fixer le moteur, Il est important que le plan de fixation de chaque moteur soit parfaitement parallèle au plan que forme la partie haute ou basse du tube carré. Ensuite il ne reste qu'à réaliser trois autres supports identiques au premier.

Les quatre bras avec leur support moteur sont réalisés, la prochaine étape consiste à monter les quatre moteurs sur leurs supports respectifs. Cependant Il est évident que le montage ne doit comporter aucun jeu pouvant favoriser des vibrations.

Son poids actuel est d'environ 980g. A cela va s'ajouter les quatre contrôleurs de vitesse 70 g le récepteur 25g et surtout l'accu de 80g. En gros on arrive à un poids total minimum de 1430g plus le câblage, les 3 hélices de 20 g et la camera.

Il est à noter que tout excédent de poids serait mauvais. Il faut réduire au maximum le poids car, chaque gramme économisé est un gain en temps de vol.



Figure VI.23: image de drone finalisé.

La carte noire, c'est le fameux cerveau de l'appareil, elle est fixée solidement sur des petites entretoises. par ailleurs les variateurs de vitesses sont tenus verticalement autours de la carte principale sur le châssis et il sera de même pour le récepteur.

Afin de protéger toute l'électronique une protection en époxy de 1 mm couvre le tout.

On peut voir sous la carte des trous, qui vont servir pour le passage des câbles d'alimentation des variateurs. Ces câbles vont transporter le courant issue de l'accu, ce dernier sera placé sous le châssis.

Maintenant il reste à faire le calibrage de tous les modules, carte principale, régulateurs, moteurs, récepteur, et émetteur.

VI.6.11. Les hélices :

On utilisera 2 types d'hélices différentes, tournant en sens opposés, et soufflant toutes vers le bas. Les hélices avant et arrière tournent dans le sens horaire, alors que celles de droite et gauche tournent dans le sens inverse.

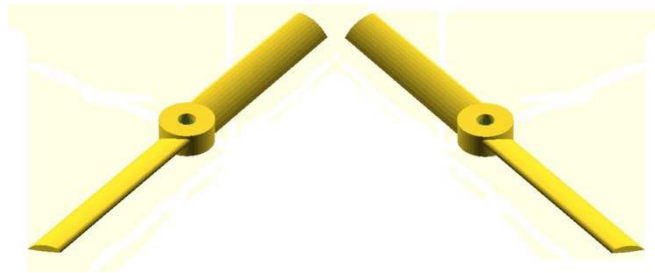


Figure VI.23: deux hélices avec sens des pales opposées.

Pour le vol stationnaire, les hélices tournent théoriquement (pour un vol stationnaire et en supposant que le centre de gravité est très proche de centre du châssis) à la même vitesse. Pour cela, les forces créées par les hélices tournant dans le sens horaire et celles des hélices tournant en sens inverse s'équilibrent. Cela permet au quadri-rotor de rester stable dans les airs.

VI.6.12.Autonomie :

Pour fournir une puissance suffisante, on utilise généralement des batteries Lithium-Polymère (ou Li-po). Ce type de batteries diffère des autres par un faible poids associé à une capacité élevée. Manipuler ces accus n'est pas sans danger, ce qui veut dire qu'il faut toujours garder à l'esprit certaines règles. Ces accus nécessitent un type de chargeur adapté, du aux processus de charge qui diffèrent beaucoup des autres types d'accus (comme les NiCad et NiMh). Les cellules ne doivent pas être surchargées, ni trop déchargées.

Pour garder un écart minimum entre les différents éléments de l'accu, un équilibreur doit être utilisé. Cet équilibreur peut être intégré au chargeur, ou un équilibreur séparé peut être relié au chargeur. Tous les accus Li-Po ont une connectique pour l'équilibrage de la charge.

On utilise pour notre application des accus Li-Po avec trois éléments. Ils ont une tension de 11,1 V. Couramment conseillé : Li-Po, 3s 11.1V ,2100mAh, 20C.

VI.6.13.Aspects de sécurité dans les drones :

Il nous a paru important d'ajouter quelques lignes sur des aspects de sécurité dans les drones basés sur notre expérience dans le domaine. Premièrement il est pertinent de

rappeler que la loi ne permet pas le vol de drones dans des zones urbaines. Le fait de rendre autonome un modèle réduit nous oblige à étudier les aspects de sécurité de manière prioritaire. Nous devons premièrement éviter le démarrage intempestif des moteurs. Aussi il est indispensable de couper l'alimentation des moteurs en cas de panne. Les pannes les plus courantes sont la perte de liaison radio bien entendue et le mauvais fonctionnement des capteurs (gyromètres, accéléromètres,). Il est donc impératif de développer des méthodes pour détecter ce type de pannes. Pour réduire le risque d'accidents il est important d'utiliser des hélices carénées et de réduire au minimum le poids total de l'appareil. L'utilisation de lunettes de protection des yeux est fortement conseillée pendant les travaux de mise au point.

VI.6.14.Conclusion :

Ce dernier chapitre à été consacré à la présentation de l'étape réalisation de notre application en mettant en pratiques les connaissances et déductions des autres chapitres.

Ainsi, nous avons présenté les différentes opérations de fabrication des moteurs, puis le développement des différentes cartes électroniques, enfin le montage et les testes.

CONCLUSION GÉNÉRALE :

L'objectif final de ce mémoire étant de réaliser un drone de type quadri-rotor ,si bien que nous avons débuté par une étude théorique de principe de vol de l'appareil puis songer a la manière de réaliser le contrôleur d'un vole autonome (stabilisation automatique) et les éléments concourants à cette application tel les microcontrôleurs gestion principale et ceux de control de vitesse suivis de l'étude des détecteurs de navigation aérienne et le choix de ces derniers.

Dans l'étape suivante nous avons débutés la construction des moteurs et les éléments indispensables à leurs fonctionnements, tel les contrôleurs de vitesse en veillant à avoir suffisamment de puissance pour faire décoller l'engin.

Autre l'étude et la réalisation des moteurs et le contrôleur de vol ; nous avons étudié différents systèmes de communication et de contrôle a distance.tel le codage ppm/ pcm, la communication xbee ...

Ensuite nous avons montrés la programmation des différents microcontrôleurs suivis de l'élaboration de quelques méthodes de contrôles évoluées via le logiciel labview.

Enfin il ne restait que l'étape consistant a effectuer des teste de vol et de comparer les résultats théorique avec les mesures effectuées sur le terrain et des améliorations seront apportés en conséquences.

Ce projet a été à la fois très intéressant et très formateur. Nous avons pu concevoir et réaliser un drone, mettant en application nos connaissances acquises durant les années d'études.

Nous avons choisi notre matériel selon les fonctionnalités souhaitées tout en prenant en compte le temps de développement, le coût du matériel, les contraintes techniques et la complexité de mise en place. La réalisation de ce drone nous a également donné l'opportunité de travailler sur plusieurs domaines à la fois

notamment la programmation, l'électronique, la gestion de l'énergie et la mécanique. Nous avons donc au final pu mener un vrai travail de conception de système, ce qui s'apparente à un vrai travail d'ingénieur.

De manière générale ce projet nous a ouvert l'esprit à toutes ses perspectives, notamment professionnelles.

Mots clés :

Drone, microcontrôleur, Arduino, AVR, capteurs de navigation aérienne, gyroscope, accéléromètre, circuit électronique embarqué, boucle d'asservissement PID, xbee...

Les références

[1]:

Titre: Flight Dynamics of Aeroelastic Vehicles.

Auteur : Martin, R., David, K, 1988.

[2]:

Titre : *Stabilisation de véhicules aériens à décollage vertical :
Théorie et application* Heudiasyc, UMR CNRS

Auteur : R. Lozan – P, Castillo – S, Salazar - D.Lara,1999.

[3]:

Titre: Multi-layered Control of a Four-Degree-of-Freedom Mobile Robot With Compliant Linkage.

Auteur : Borenstein, J.1997.

[4]:

Titre : *Synthèse et Implémentation d'un contrôleur pour micro
hélicoptère 4 rotors.*

Auteur : André, North, 1995.

Sites web:

www.DIYdrone.com.

www.flyingrobot.com.

Autopilot Quadcopter - Toradex Challengeyu.htm/.

quadcopter matlab simulation - Free Open Source Codes - CodeForge.com.htm/.

Forum de discussions consacré en grande partie aux drones quadri- rotor :

<http://forum.xbird.org/>.

Budget et dépenses :

Le total des dépenses est approximative et avoisine le 25800,00 DA ce qui inclus les composants passifs et des transistors bipolaires servant de driver des Mosfets de puissance, A noter que ces derniers (les Mosfets) sont les éléments les plus chers globalement de part la quantité demandée par les contrôleurs de vitesse. Nous avons eu également besoin de barres en aluminium et de contre-plaqué pour construire la structure du drone quadri rotor.

Le tableau ci-dessous présente les prix des différents composants utilisés.

Composants	Quantité	Prix (unitaire) DA
Microcontrôleur AVR atmega328	1	600,00
Microcontrôleur AVR atmega8	4	450,00
Emetteur 2.4Ghz FM-10mw +récepteur 2.4Ghz a double conversion de fréquences.	1	x
Encodeur et décodeur PPM 6 voix	1	<i>Fait maison...</i>
Moteur sans balais	4	2100,00 (<i>fait maison</i>)
Mosfet	56	160,00
Les stators	4	<i>Récupération</i>
Roulement	8	400,00
Aimants	14x4	<i>récupération</i>
Batteries (éléments li-ion de 3.7V).	6	300,00
Camera sans fils (analogique).	1	3000,00
Carte de développement de contrôleur.	1	400,00

principal.		
cartes de développements de contrôleurs de vitesse.	4	450,00
Châssis en aluminium.	3	<i>récupération</i>
Boulons+ vis	x	x
hélices fait maison.	4	x
Gyroscope ITG3200	1	2000,00
Accéléromètre « bma180 »	1	-
Programmateur AVR (<i>USBASP à base d'Atmega8</i>)	1	700,00 (<i>Fait maison</i>)
Les composants électroniques (transistors, résistances, capacités...)	x	x
UART	1	x
Total	-	25800,00

Quelques définitions :

Coefficient de traînée :

Un mobile se déplaçant dans un fluide pesant (comme l'air) subit de la part de ce fluide une distribution de pression dont la résultante s'oppose à sa marche. La composante de cette résultante selon la direction du mobile est appelée traînée. L'intensité de la force de traînée est exprimée en fonction de la vitesse, de la forme et de la taille du mobile, du fluide dans lequel il interagit. Le coefficient de traînée usuellement désigné C_x est un nombre sans dimension caractéristique de la forme du mobile uniquement. C'est le rapport entre la force de traînée F_x , le produit de la pression dynamique q , par la surface de référence S du mobile :

$$C_x = \frac{F_x}{q \times S}$$

où la pression dynamique est donnée par : $q = \frac{1}{2} \times \rho \times V^2$.

Avec : ρ masse volumique du fluide et V : la vitesse du mobile par rapport au fluide.

Aérodynamique :

L'aérodynamique est une branche de la dynamique des fluides qui porte principalement sur la compréhension et l'analyse des écoulements d'air, ainsi qu'éventuellement sur leurs effets sur des éléments solides qu'ils environnent. L'aérodynamisme (terme non scientifique) qualifie l'apparence d'un corps en mouvement dans l'air et sa résistance à l'avancement. L'aérodynamique s'applique aux véhicules en mouvement dans l'air (aérodynes, automobiles, trains), aux systèmes de propulsion (hélices, rotors, turbines, turboréacteurs), aux installations fixes dans un air en mouvement subissant les effets du vent (bâtiments, tours, ponts) ou destinés à la production d'énergie (éoliennes), aux systèmes mécaniques transformant une énergie aérodynamique en énergie mécanique et vice-versa (turbines, compresseurs).

Couple (physique) :

De manière générale, en physique, on appelle **couple** tout système d'actions mécaniques dont la résultante \vec{R} est nulle et le moment résultant \vec{M} par rapport à un point O est non nul. Ce moment est alors *indépendant* du point O . Par abus de langage, on appelle aussi couple la valeur de ce moment. En mécanique, un couple est l'effort en rotation appliqué à un axe. Il est ainsi nommé en raison de la façon caractéristique dont on obtient ce type d'action : un bras qui tire, un bras qui pousse, les deux forces étant égales et opposées. Lorsque le couple ne s'exerce pas rigoureusement dans l'axe, il se produit une rotation de cet axe (précession). On mesure le couple en newton-mètre (N·m). L'unité de travail, le joule (J), est homogène au newton-mètre : un couple de 1 N·m appliqué à un axe qui tourne d'un tour représente un ajout d'énergie de 2π J. On le représente par un vecteur dans l'axe de rotation, vers le haut pour une rotation dans le sens trigonométrique (qui est l'inverse du sens des aiguilles d'une montre), comme la vitesse de rotation.

Inertie :

En physique, l'inertie d'un corps dans un référentiel galiléen (dit *inertiel*) est sa résistance à une variation de vitesse. L'inertie est fonction de la masse du corps : plus celle-ci est grande, plus la force requise pour modifier son mouvement sera importante. La notion d'inertie est à la base du principe d'inertie, première loi de Newton. L'inertie est une propriété précisée par le principe d'inertie, valable dans les référentiels inertiels (plus exactement : il définit ces référentiels). C'est une notion qualitative, est quantifiée par la masse inerte qui intervient dès la définition de la force, qui elle-même est une quantification de l'idée d'effort nécessaire à influencer sur le mouvement inertiel tel que l'expose le principe d'inertie.

Hélice aérienne :

Schématiquement, une hélice d'avion ou d'hélicoptère est une aile en rotation (les ailes et les pales ont des profils voisins), générant une force propulsant l'avion vers l'avant. L'hélice en rotation prend l'air en avant de l'avion et le projette vers l'arrière à une vitesse supérieure à celle de l'avion : la masse d'air qui passe à travers l'hélice est accélérée. Selon le principe de la conservation de la quantité de mouvement une force de réaction s'applique sur l'hélice et la pousse vers l'avant. Une explication courante attribue cette poussée au principe de Bernoulli.

Mandrin :

Le **mandrin** est une pièce mécanique fixée au bout de l'arbre d'une machine rotative ; il permet la fixation rapide d'un outil (comme le mandrin de perçage) ou d'une pièce (comme le mandrin de tour). Un mandrin peut serrer un foret, une fraise qui vont enlever de la matière dans une pièce à usiner, ou un outil pour le serrage de vis ou écrou : empreinte ou douille dans le cas d'une visseuse.

Fonction de transfert :

C'est une représentation mathématique de la relation entre l'entrée et la sortie d'un système linéaire, le plus souvent invariant. Elle est utilisée, notamment, en traitement du signal, en théorie des communications, en automatique, et dans toutes les sciences de l'ingénieur qui font appel à cette discipline (électronique, mécanique, mécatronique, etc.). Dans les systèmes multidimensionnels on représente la matrice de transfert plutôt que la fonction de transfert.

Le filtre de Kalman :

Le filtre de Kalman est un filtre à réponse impulsionnelle infinie qui estime les états d'un système dynamique à partir d'une série de mesures incomplètes ou bruitées. Le filtre de Kalman est un estimateur récursif. Cela signifie que pour estimer l'état courant, seuls l'estimation de l'état précédent et les mesures actuelles sont nécessaires. L'historique des observations et des estimations n'est ainsi pas requis. Le filtre de Kalman est utilisé dans une large gamme de domaines technologiques (radar, vision électronique, communication ...). C'est un thème majeur de l'automatique et du traitement du signal. Un exemple d'utilisation peut être la mise à disposition, en continu, d'informations telles que la position ou la vitesse d'un objet à partir d'une série d'observations relative à sa position, incluant éventuellement des erreurs de mesures.

Par exemple, pour le cas des radars où l'on désire suivre une cible, des données sur sa position, sa vitesse et son accélération sont mesurées à chaque instant mais avec énormément de perturbations dues au bruit ou aux erreurs de mesure. Le filtre de Kalman fait appel à la dynamique de la cible qui définit son évolution dans le temps pour obtenir de meilleures données, éliminant ainsi l'effet du bruit. Ces données peuvent être calculées pour l'instant présent (filtrage), dans le passé (lissage), ou sur un horizon futur (prédiction).

Filtre impulsionnel infinie :

Un **filtre à réponse impulsionnelle infinie** ou **filtre RII** est un type de filtre électronique caractérisé par une réponse basée sur les valeurs du signal d'entrée ainsi que les valeurs antérieures de cette même réponse. Il est nommé ainsi parce que dans la majorité des cas la réponse impulsionnelle de ce type de filtre est de durée théoriquement infinie. Il est aussi désigné par l'appellation de filtre récursif. Ce filtre est l'un des deux types de filtre numérique linéaire. L'autre type possible est le filtre à réponse impulsionnelle finie (filtre RIF). Contrairement au filtre RII la réponse du filtre RIF ne dépend que des valeurs du signal d'entrée. Par conséquent, la réponse impulsionnelle d'un filtre RIF est toujours de durée finie.

La plupart des filtres analogiques peuvent également être considérés comme des filtres à réponse impulsionnelle infinie.

L'effet Hall :

Lorsqu'un courant traverse un barreau en matériau semi-conducteur (ou conducteur), et si un champ magnétique d'induction **B** est appliqué perpendiculairement au sens de passage du courant, une tension, appelée tension Hall, proportionnelle au champ magnétique et au courant apparaît sur les faces latérales du barreau. Cette tension est proportionnelle à la vitesse de déplacement des porteurs de charge qui est considérablement plus grande dans les matériaux semi-conducteurs que dans les conducteurs métalliques.

Les capteurs à effet Hall permettent de mesurer :

- les champs magnétiques (tesla-mètres) .
- l'intensité des courants électriques : capteur de courant à effet Hall. (exemple : pince ampérométrique à effet Hall).
- ils permettent également la réalisation de capteurs ou des détecteurs de position sans contact, utilisés notamment dans l'automobile, pour la détection de position d'un arbre tournant (boîte à vitesses, cardans...) .
- les compteurs électriques domestiques (électroniques, les mécaniques utilisant le courant de Foucault).

Régulateur PID :

Un régulateur PID ou correcteur PID (pour « proportionnel intégral dérivé ») est un organe de contrôle permettant d'effectuer une régulation en boucle fermée d'une grandeur physique d'un système industriel ou "procédé" .Il permet de régler un grand nombre de grandeurs physiques.

Néodyme :

Le **néodyme** est un élément chimique, de symbole Nd et de numéro atomique 60. C'est un métal gris argent du groupe des terres rares. Il fait partie de la famille des lanthanides. À température ambiante, il est ductile, malléable et s'oxyde rapidement à l'air. Son nom vient des mots grecs *nêos* et *dídymos* signifiant respectivement « nouveau » et « jumeau ». En effet, les chimistes ont longtemps cru que le mélange d'oxyde de praséodyme-néodyme était un corps simple jusqu'à ce que Carl Auer von Welsbach les sépare en 1885.

Dans les aimants permanents, ils sont utilisés en alliage avec le fer $Nd_2Fe_{14}B$ car ils sont extrêmement puissants mais assez fragiles. Ils sont utilisés :

- dans les générateurs des éoliennes.
- Dans les moteurs électriques et les générateurs de certaines voitures hybrides.
- Dans certains transducteurs des membranes de casques stéréo haut-de-gamme, ainsi que dans les nouvelles générations d'enceintes de sonorisation et de microphones dynamiques...etc.

CARACTERISTIQUES DES MODULES XBEE PRO :

Modulation RF : Modulation en quadrature de phase avec décalage

Bande d'émission : 2,4GHz

Largeurs des canaux de transmission : 5 MHz

SPECIFICATIONS		XBee	XBee-PRO
Performances	Portée intérieur / milieu urbain	Jusqu'à 30 m	Jusqu'à 100 m
	Portée extérieure champs libre	Jusqu'à 100 m	Jusqu'à 1,6 km
	Puissance d'émission RF	1 mW (0 dBm)	60 mW (18dBm)*, 100mW EIRP*
	Vitesse de transmission RF	250Kbps	250K bps
	Sensibilité de réception	-92 dBm (1% PFR)	-100 dBm (1% PFR)
Alimentation et consommation	Tension d'alimentation	2.8VDC – 3.4VDC	2.8VDC – 3.4VDC
	Courant en émission (typique)	45 mA (@ 3.3V)	215 mA (@ 3.3V, 18 dBm)
	Courant en réception (typique)	50 mA (@ 3.3V)	50 mA (@ 3.3V)
	Courant de repos	< 10 µA	< 10 µA
Autres caractéristiques générales	Fréquence	ISM 2,4 GHz	ISM 2,4 GHz
	Dimensions	24,4 mm x 27,6 mm	24,4 mm x 32,9 mm
	Température de fonctionnement	-10 to 85° C (classe industrielle)	-10 to 85° C (classe industrielle)
	Antennes optionnelles	Connecteur U.FL, antenne chip ou antenne Whip intégrée	Connecteur U.FL, antenne chip ou antenne Whip intégrée
Réseau et sécurité	Topologies réseaux supportés	Point à point, Point à Multipoint, Peer to Peer et Mesh	Point à point, Point à Multipoint, Peer to Peer et Mesh
	Nombre de canaux	16 canaux séquence directe (sélection par logiciel)	12 canaux séquence directe (sélection par logiciel)
	Options de filtrage	PAN ID, Canal & Source / Adresses de destination	PAN ID, Canal & Source / Adresses de destination
Certifications	FCC Part 15.247	OUR-XBEE	OUR-XBEEPRO
	Industry Canada (IC)	4214A-XBEE	4214A-XBEEPRO
	Europe (CE)	ETSI	ETSI (puissance TX max = 10mW)

* Pour l'Europe : les modules XBee-PRO doivent être configurés avec une puissance d'émission TX max de 10dBm (niveau de sortie configurable par la commande PL). La législation Européenne stipule une puissance max. EIRP de 12.86 dBm (19mW).

Figure : tableau montrant les caractéristiques des deux types de modules xbee (standards ou pro).

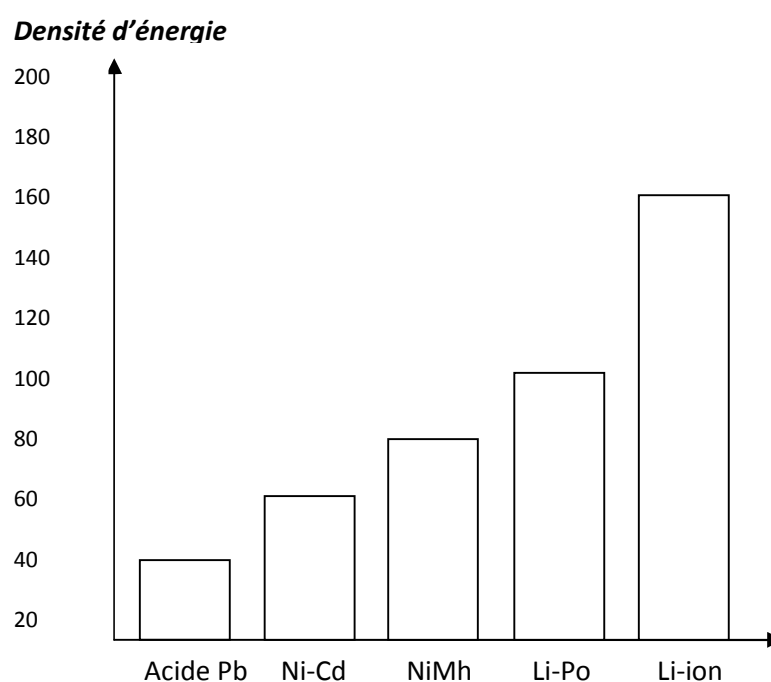


Figure : comparaison des performances des batteries suivant leurs types.

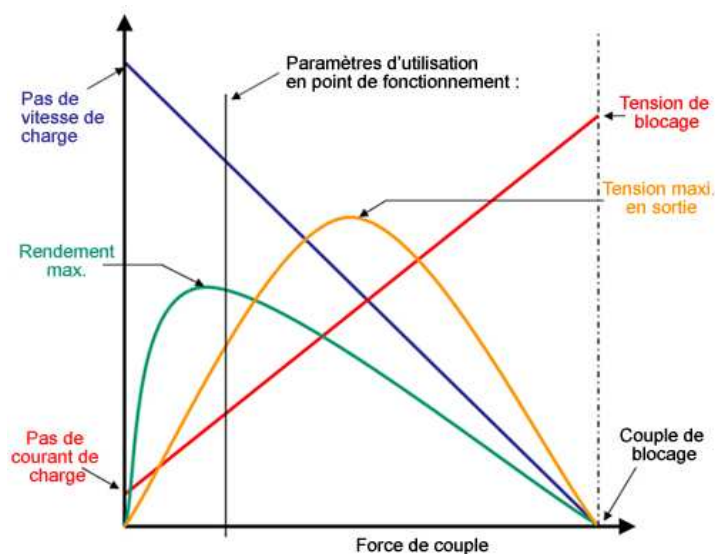


Figure : courbes montrant les variations des caractéristiques des moteurs sans balais.

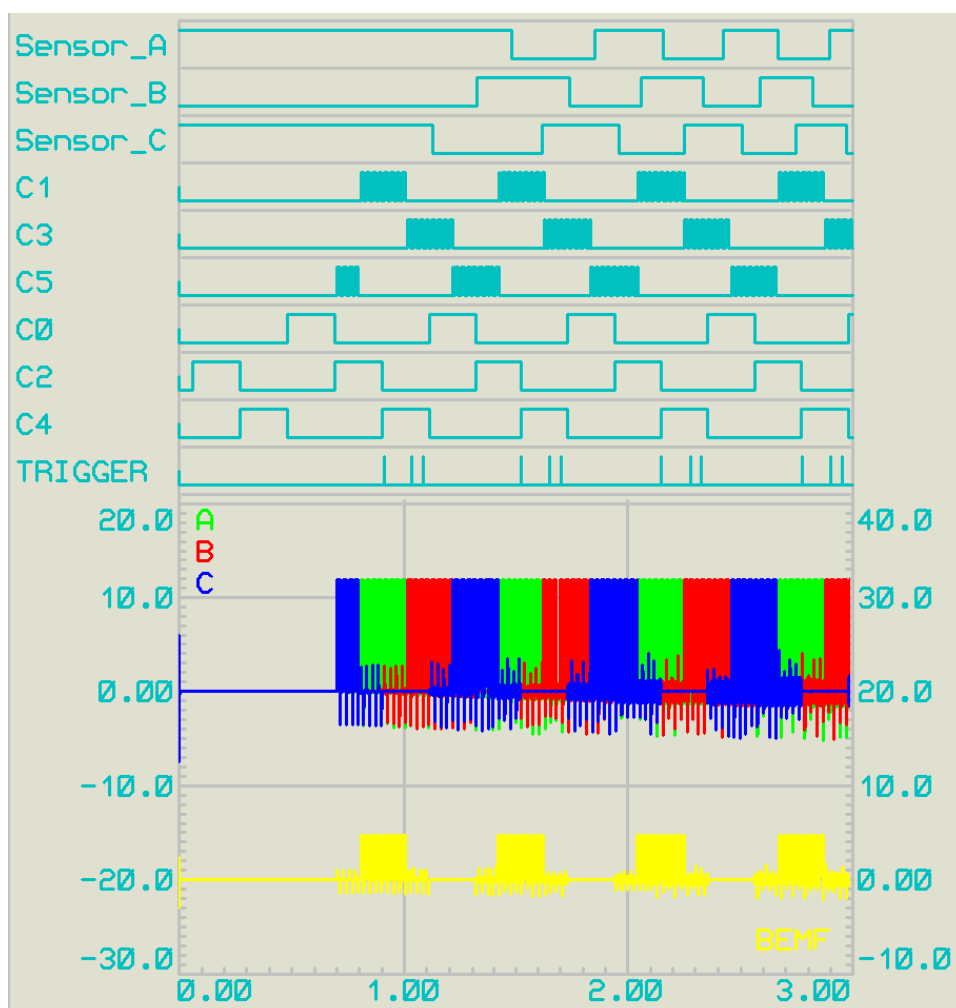


Figure : analyse des signaux d'un moteur sans balais

Angle (Radians)

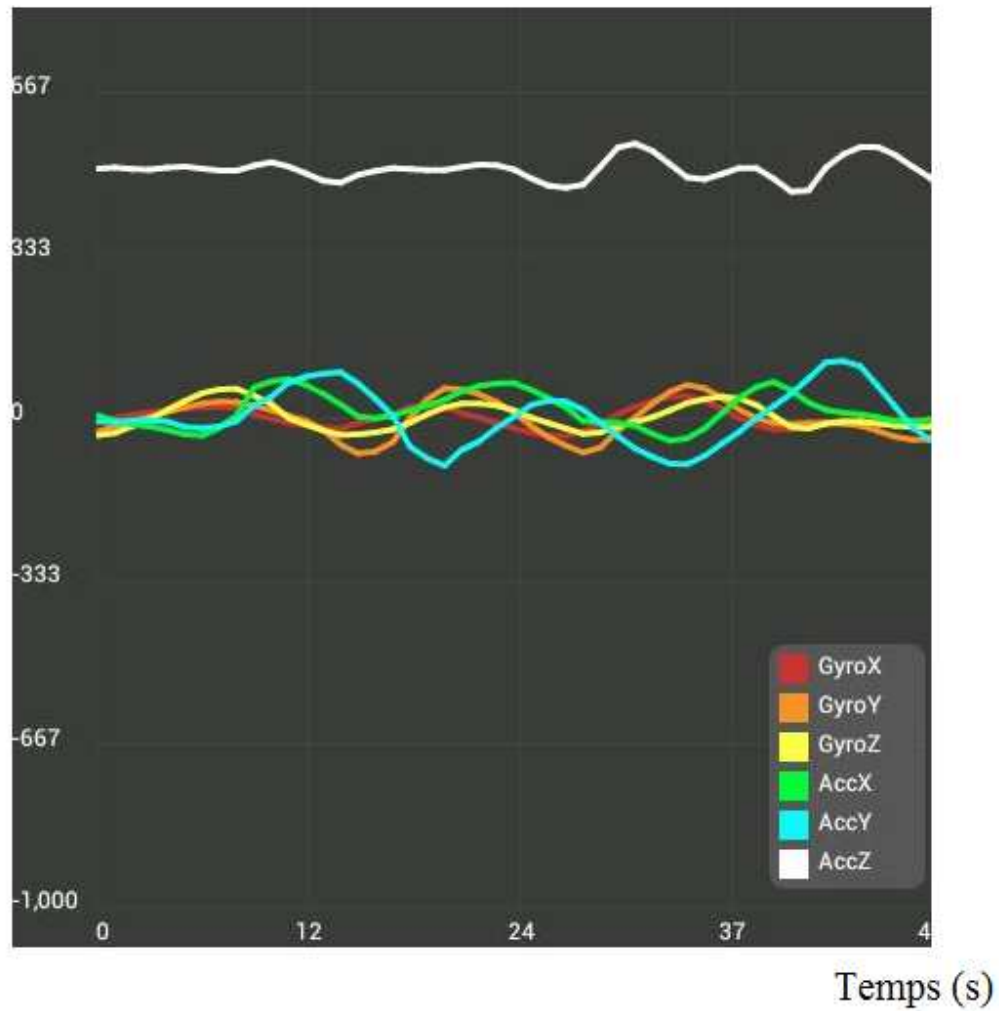


Figure : Variation d'accélération et de vitesse angulaire en fonction de temps.

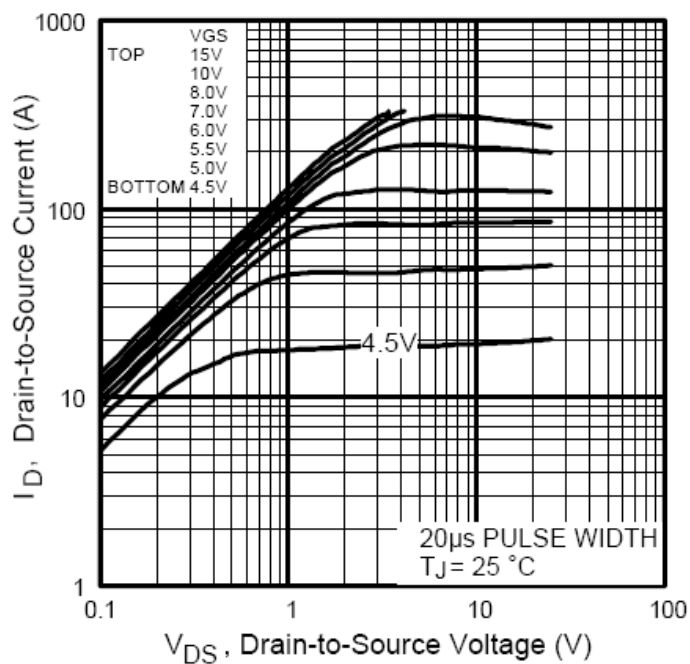


Figure : caractéristiques typiques de mosfet Irf1010 [de Irf1010 datasheet].

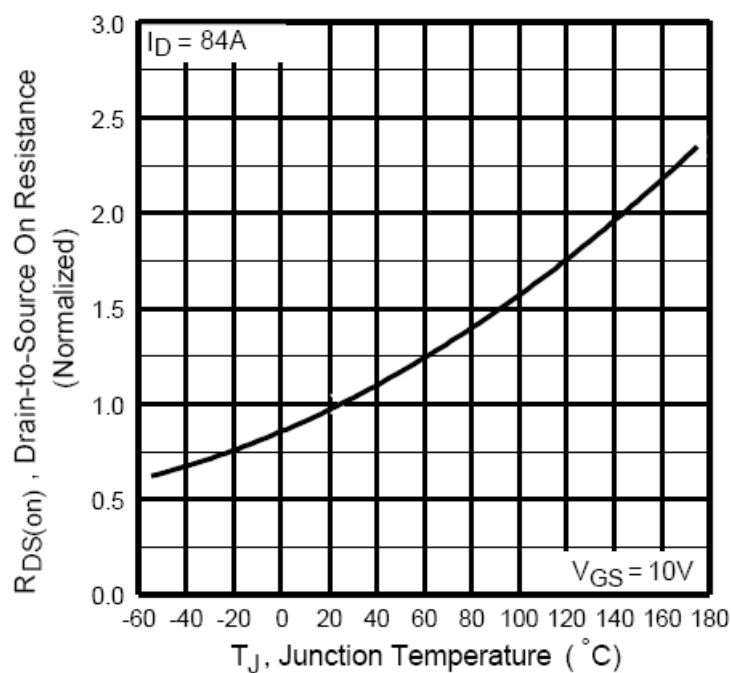


Figure : Résistance Drain/Source en fonction de température de jonction.

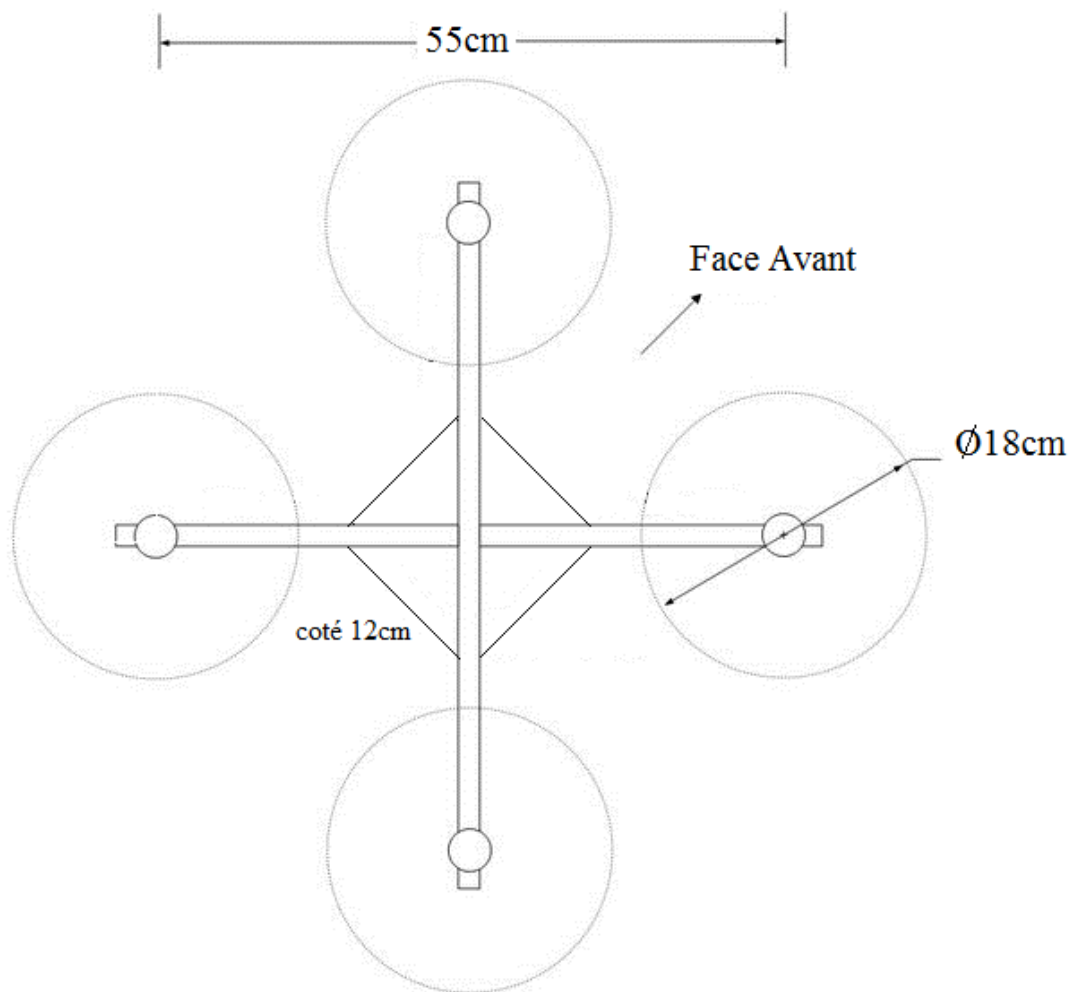
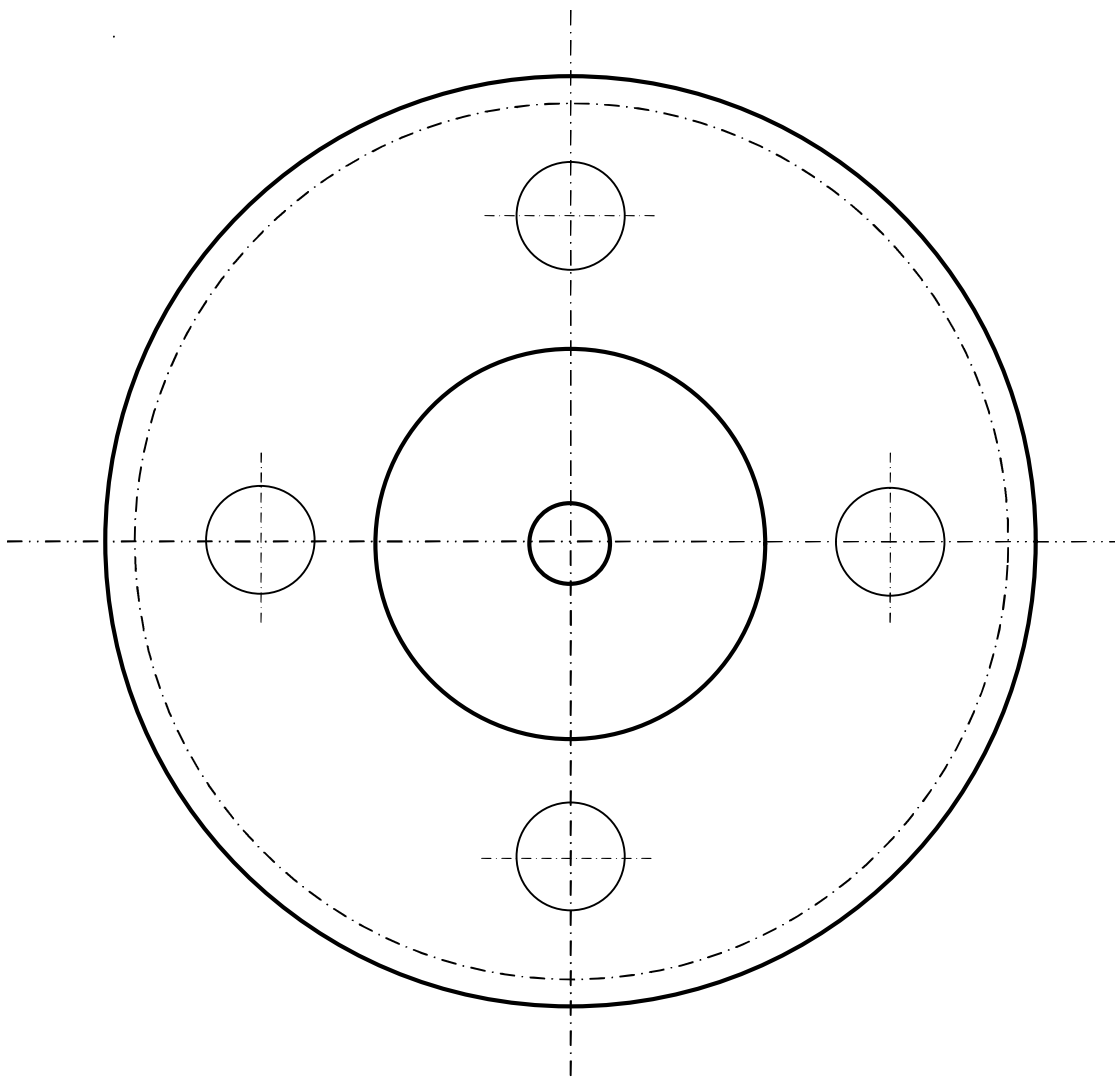
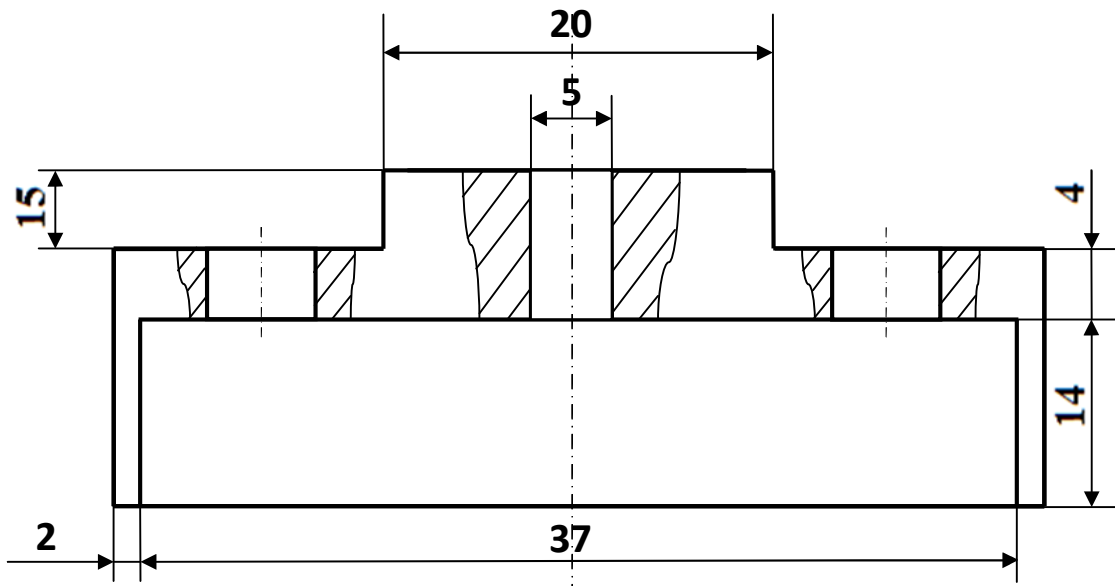
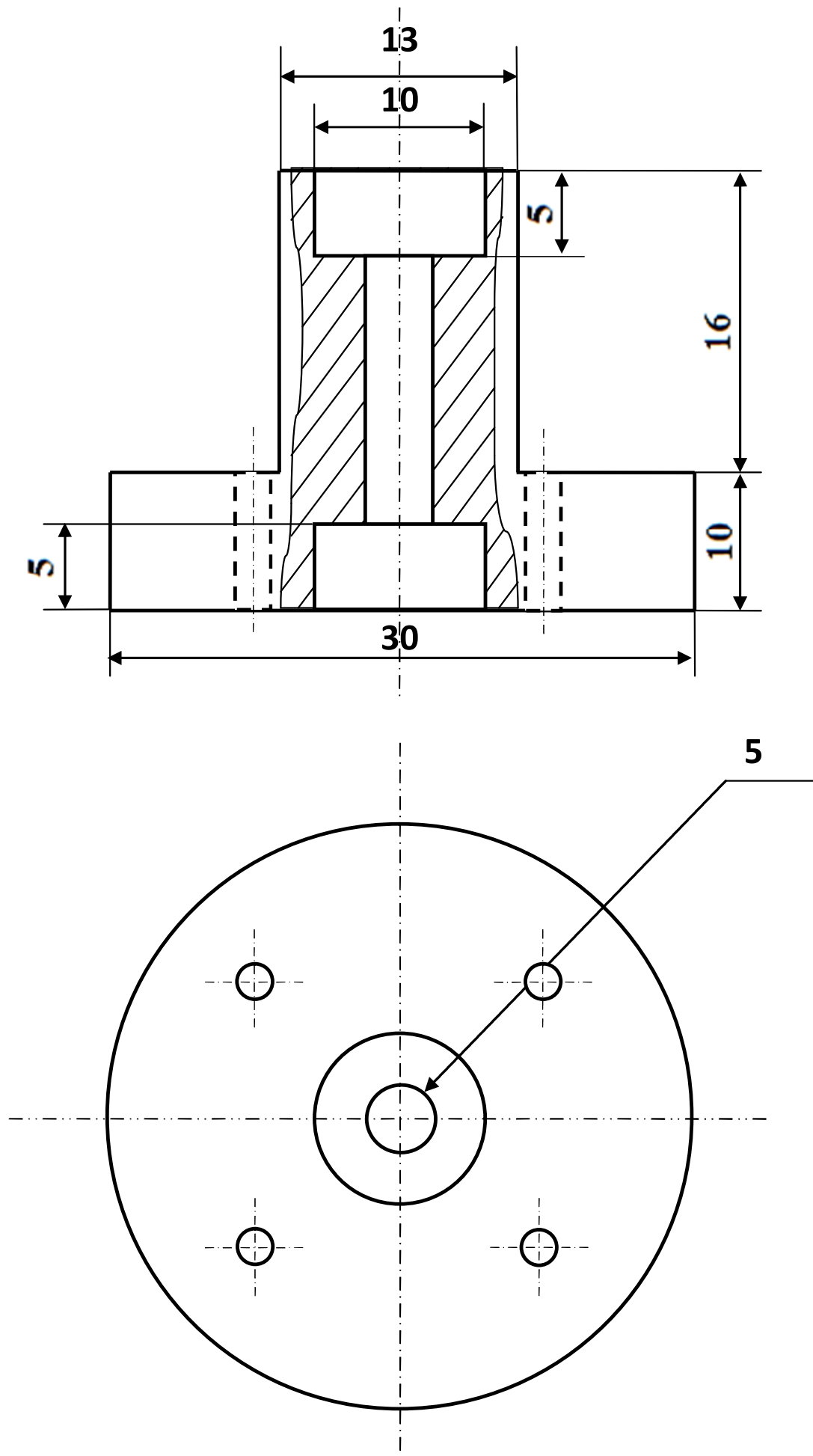
Vue de dessus de quadri-rotor :

Figure : vue de dessus de quadri-rotor.

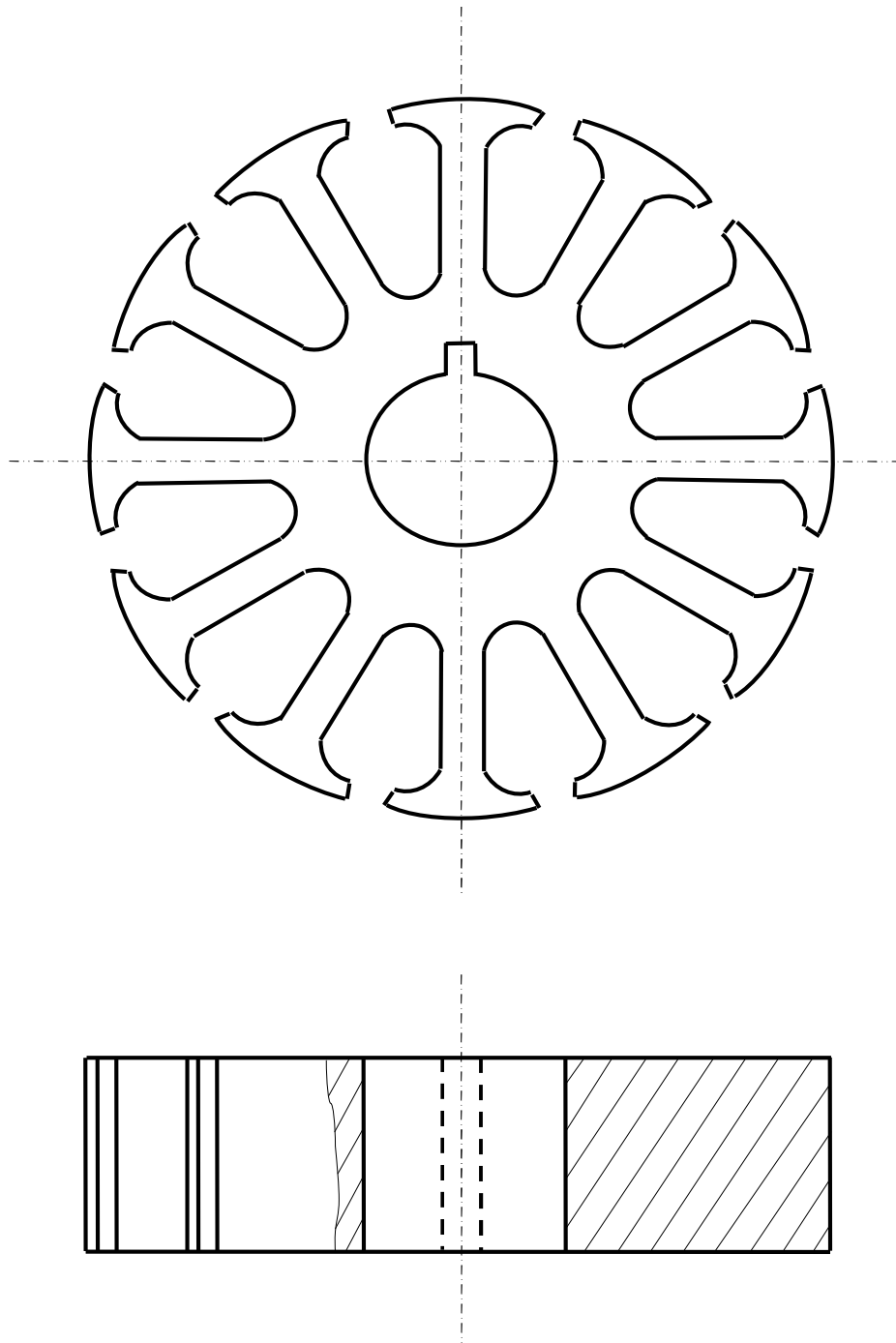
Les dessins des éléments fabriqués constituant les moteurs :**1)-Le rotor :**

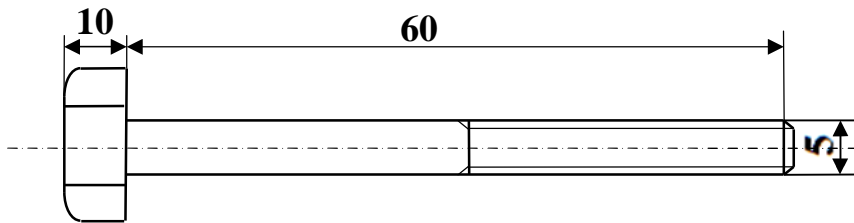


2)-Le support stator :



3)-Le stator :



4)-la vis port hélice :**5)-Schéma global de moteur :**