

## TABLE DES MATIERES

<i>REMERCIEMENTS</i> .....	i
RESUME .....	ii
ABSTRACT .....	iii
TABLE DES MATIERES .....	iv
LISTE DES ABREVIATIONS .....	vi
LISTE DES FIGURES .....	vii
LISTE DES TABLEAUX .....	viii
INTRODUCTION.....	1
CHAPITRE I : LES EQUIPEMENTS EMBARQUES DANS UN MINIDRONE QUADRIROTOR.....	2
I.1 – Equipements de navigation .....	2
I.1.1 – Centrale inertielle .....	2
I.1.2 – Localisation GPS .....	4
I.1.3 – DéTECTEUR d’obstacles.....	4
I.1.4 – Altitude .....	5
I.2 – Système d’alimentation.....	6
I.3 – Système de propulsion .....	7
I.4 – Capteurs de surveillance .....	8
I.4.1 – Température .....	9
I.4.2 – Tension .....	9
I.4.3 – Vitesse moteur.....	10
I.5 – Equipements de mission.....	11
I.5.1 – La Caméra .....	12
I.5.2 – Emetteur et récepteur embarqué.....	12

CHAPITRE II : ANALYSE ET TRAITEMENT DES DONNEES EXPLOITEES .....	16
II.1 - Système de drone .....	16
II.1.1 – Drone .....	16
II.1.2 – Opérateur .....	18
II.1.3 - Poste de commande .....	18
II.2 - Format des données .....	19
II.2.1 - Données ascendantes .....	19
II.2.2 - Données descendantes.....	20
II.3 - Traitement et Synchronisation des commandes.....	22
II.3.1 - Commande manuelle .....	23
II.3.2 - Autonomie décisionnelle.....	25
CHAPITRE III : INTERFACAGE DES EQUIPEMENTS EMBARQUES .....	27
III.1- Généralités .....	27
III.2- Interfaces utilisées.....	28
III.2.1- Interfaces de puissance .....	28
III.2.2- Contrôleur de vitesse .....	31
III.2.4- Interfaces pour les capteurs.....	33
III.3- Architectures logicielles .....	33
III.3.1- Gestion de la carte contrôleur .....	33
III.3.2- Gestion du microcontrôleur central .....	37
.....	41
CONCLUSION .....	43
ANNEXE 1 : Spécificité des Microcontrôleurs .....	44
ANNEXE 2 : Moteurs brushless de type outrunner .....	49
ANNEXE 3 : EXTRAIT CODE SOURCE .....	50

## LISTE DES ABREVIATIONS

ADC	: Analog to Digital Converter
AWGN	: Additive White Gaussian Noise
CAN	: Convertisseur Analogique Numérique
CNA	: Convertisseur Numérique Analogique
FH-SS	: Frequency-Hopping Spread Spectrum
FSK	: Frequency Shift Keying
GPS	: Global Positioning System
GUI	: Graphical User Interface
IDE	: Integrated Development Environment
LASER	: Light Amplification by Stimulated Emission of Radiation
LiPo	: Lithium Polymère
MSB	: Most Significant Bit
NRZ	: Non Return to Zero
OSI	: Open Systems Interconnection
PIC	: Peripheral Interface Controller
PWM	: Pulse with Modulation
QAM	: Quadrature Amplitude Modulation
RISC	: Reduced Instruction Set Computer
ROS	: Rapport d'onde stationnaire
RZ	: Retour à Zero
SNR	: Signal Noise Ratio
SNR	: Signal to Noise Ratio
TCAS	: Traffic alert and Collision Avoidance System
UAV	: Unmanned Aerial Vehicle
UWB	: Ultra WideBand

## LISTE DES FIGURES

Figure 1. 1 : Gyromètre vibrant .....	3
Figure 1. 2 : Accélémètre .....	3
Figure 1. 3 : Détection par infrarouge .....	4
Figure 1. 4 : Oscillation du signal reçu.....	5
Figure 1. 5 : Capteur barométriques .....	6
Figure 1. 6 : configuration théorique du moteur .....	7
Figure 1. 7 : Montage du DS1820 .....	9
Figure 1. 8 : Détecteur de niveau Batterie faible .....	10
Figure 1. 9 : Fonctionnement caméra CCD .....	12
Figure 1. 10 : Schéma du module émetteur .....	13
Figure 1. 11 : Schéma du module récepteur .....	14
Figure 1. 12 : Architecture générale du mini drone .....	14
Figure 2. 1 : Type de drone quadrirotor.....	16
Figure 2. 2 : Les mouvements du drone .....	17
Figure 2. 3 : Format de données ascendantes .....	19
Figure 2. 4 : Données de navigation.....	20
Figure 2. 5 : Données vidéo émises depuis le drone .....	20
Figure 2. 6 : Boucle d'exécution intégrée dans le système de contrôle principal .....	22
Figure 2. 7 : Joystick de pilotage de drone .....	23
Figure 2. 8 : Enchaînement des dispositifs embarqués.....	25
Figure 3. 1 : Pilotage du gate des transistors montés en pont .....	29
Figure 3. 2 : Configuration du mosfet IRFZ44 à canal N.....	30
Figure 3. 3 : Schéma de principe de l'interface de puissance .....	30
Figure 3. 4 : Configuration des ports du PIC16F84A .....	31
Figure 3. 5 : Découpage du courant d'alimentation par modulation de fréquence .....	35
Figure 3. 6 : Circuit de détection de position du rotor.....	36
Figure 3. 7 : Commutations des phases du moteur [12] .....	36
Figure 3. 8 : Format des registres de température .....	40
Figure 3. 9 : Architecture générale des équipements .....	41

## **LISTE DES TABLEAUX**

Tableau I : Caractéristique des données ascendantes .....	19
Tableau II: Longueur des données .....	21
Tableau III : Combinaison des commandes.....	24
Tableau IV: Les 6 phases du moteur vues à la sortie de l'interface de puissance .....	29
Tableau V : Les ports du microcontrôleur .....	32
Tableau VI : les combinaisons des 6 phases de commutation du moteur .....	34
Tableau VII : Priorité des commandes à exécuter.....	37
Tableau VIII : correspondance entre mouvement et actionneurs.....	38
Tableau IX : résultat de traitement du module IR .....	39
Tableau X : Relation entre température et données .....	40

## INTRODUCTION

Explorer le monde inconnu d'antan, ça fait partie de la nature humaine. De ce fait, plusieurs inventions ont été mises au point par l'homme. Mais lorsque l'exploitation s'effectue dans un milieu hostile, le risque d'y laisser la vie est trop élevé. Et l'homme continu à inventer des machines pour aboutir à ses fins. Avec les avancées technologiques, l'aéronef non habité ou UAV a vu le jour. Contrairement aux appareils volants qui nécessitent la présence humaine à bord, l'UAV est téléguidé.

Pour mettre au point un tel système, plusieurs paramètres doivent être pris en considération tant au sol que dans les airs, dont la communication. Quel que soit la mission confiée au drone, il faut traiter les informations à transmettre et les ordres à exécuter.

Dans le présent mémoire intitulé « Traitement de données dans un drone », nous allons essayer de coordonner les actions des équipements embarqués pour accomplir une mission. Ce travail est divisé en trois chapitres. Le premier chapitre est consacré à l'étude des équipements embarqués. On y présente les principes de fonctionnement de chaque dispositif embarqué. Le deuxième chapitre est destiné à l'analyse et le traitement des données exploités par le drone. Enfin, le troisième chapitre est axé à la gestion des équipements embarqués.

# CHAPITRE I : LES EQUIPEMENTS EMBARQUES DANS UN MINIDRONE QUADRIROTOR

Pour accomplir une mission, le drone embarque plusieurs équipements. Selon leur spécificité, on peut grouper ces équipements en plusieurs éléments. Dans ce chapitre, nous allons présenter ces différents groupements.

## I.1 – Equipements de navigation

Dans les domaines aéronautiques, les systèmes de navigation nécessitent de connaître plus précisément la position et la vitesse de l'engin. Pour notre drone, on a essentiellement besoin de central inertiel, d'un système de localisation GPS ainsi qu'un détecteur d'obstacle.

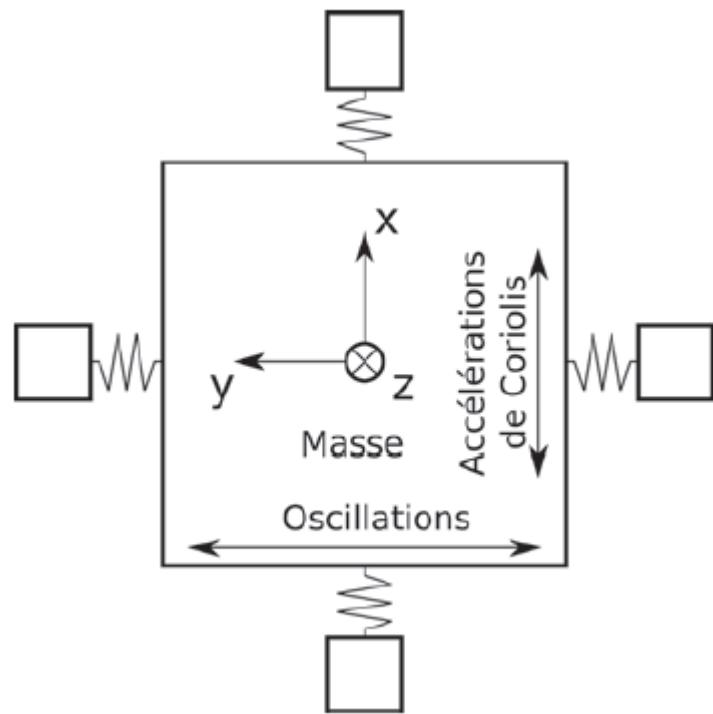
### I.1.1 – Centrale inertielle

La navigation inertuelle repose sur l'utilisation des centrales inertielles. Il s'agit d'un composant électronique de navigation de précision qui mesure l'accélération et la vitesse angulaire à l'aide de gyromètres et d'accéléromètres. Grâce à ces capteurs, la position et la vitesse peuvent être déterminées à chaque instant. Les centrales inertielles sont souvent composées de trois accéléromètres et trois gyromètres.

#### a – Gyromètre

Le gyromètre permet de déterminer les variations d'angle de rotation suivant un ou plusieurs axes. Il est composé de capteurs détectant un effet gyroscopique. Il possède une masse sismique correctement équilibré. Cette masse sismique possède deux degrés de liberté en rotation. L'un des degrés de liberté sert à entretenir la rotation de la masse tandis que l'autre sert à mesurer la vitesse angulaire.

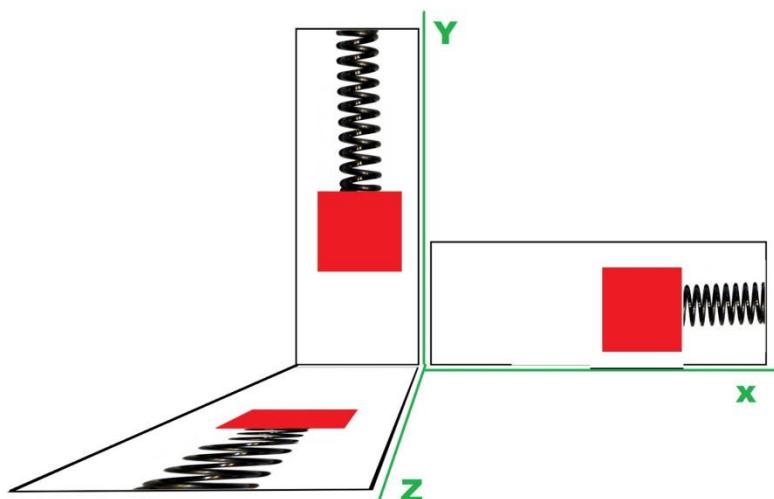
Le gyromètre vibrant représenté sur la Fig 1.1 est composé d'un élément sensible possédant la capacité de pouvoir osciller selon deux axes x et y. Lorsque l'élément sensible est en vibration le long de l'axe x, et si le gyromètre est soumis à une de rotation autour de l'axe z, une vibration apparaît le long de l'axe y. Cette vibration est excitée par la force de Coriolis et son amplitude est proportionnelle à la rotation.



**Figure 1. 1 : Gyromètre vibrant [1]**

### b – Accéléromètre

La mesure d'accélération d'un drone est réalisée par un accéléromètre à masse suspendue. Cette masse sismique, subit dans son référentiel une accélération qui peut être transposée au référentiel du drone. De ce fait, lors d'une accélération, la masse suspendue subit, selon le type de détection, un déplacement ou une déformation. La mesure du déplacement ou de la déformation permet de retrouver l'accélération.



**Figure 1. 2 : Accéléromètre [2]**

L'accéléromètre détermine une accélération sur un axe en G ( $9.806\text{m.s}^{-2}$ ). Cette valeur analogique est transformée en numérique par un CAN intégré au microcontrôleur.

### I.1.2 – Localisation GPS

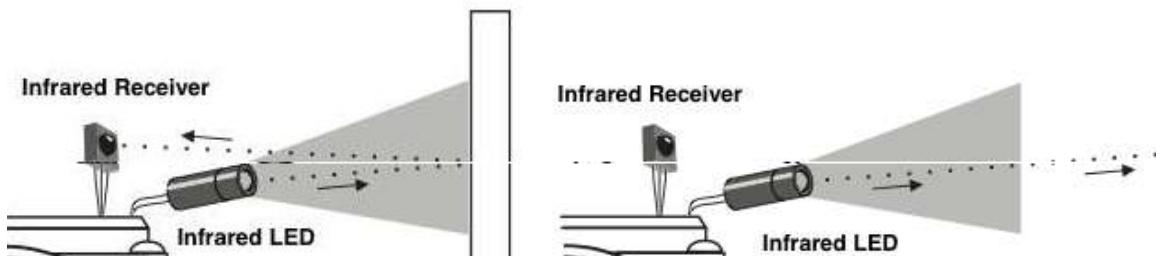
La fonction principale du module GPS est d'assurer la stabilité du quadrirotor lorsque celui-ci est éloigné du poste de commande. En effet, à 50 mètres d'altitude voire même à 100 mètres, le quadrirotor devient de plus en plus petit à l'œil, et donc la stabilisation devient un atout essentiel pour le contrôle de celui-ci. La stabilisation s'améliore encore grâce au récepteur GPS qui calcule la position dans l'espace et permet de ne pas dériver à cause du vent.

L'exploitation de ce module a fait apparaître de nouvelles fonctionnalités dernièrement. Pour les systèmes drones récents, un mode de pilotage automatique permet par exemple de choisir le lieu de destination du quadrirotor. Sur la carte géographique de l'interface du poste de commande, on peut juste indiquer le point vers lequel on veut que le quadrirotor se déplace.

### I.1.3 – DéTECTEUR d'obSTACLES

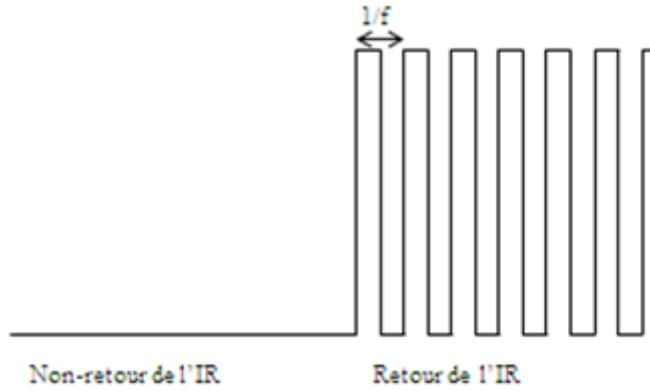
La distance séparant le drone des obstacles peut être mesurée à l'aide d'un télémètre ultrason, d'un télémètre infrarouge ou d'un télémètre laser. Mais pour une détection, l'utilisation d'un simple capteur infrarouge suffit largement. Ce dernier est de plus avantageux par rapport aux deux autres technologies du fait de sa faible consommation.

Le capteur mesure le retour ou le non-retour d'une lumière infrarouge. L'émission d'IR modulée se fait sur chacun des quatre extrémités. Si le rayon revient au récepteur, cela veut dire qu'un objet l'a renvoyé, sinon il n'y a rien.



**Figure 1. 3 : Détection par infrarouge [3]**

La diode émettrice envoie des impulsions de fréquence 38KHz. Elle s'allume et s'éteint à la fréquence demandée par le filtre du capteur en une période de  $\frac{1}{f} = 27\mu\text{s}$ , formant ainsi une oscillation comme le montre la Fig 1.4. Grâce à cette oscillation, le filtre fait la différence entre l'infrarouge de différentes sources de lumière et celle de l'émetteur. Lorsque le récepteur détecte une oscillation, cela signifie donc que le rayon a été réfléchi par un objet se trouvant devant le capteur.



**Figure 1. 4 : Oscillation du signal reçu**

Le récepteur utilise la fréquence 38KHz, l'émetteur doit donc aussi osciller à cette fréquence. Pour cela, le microcontrôleur doit commander l'émission de la diode suivant la période d'oscillation  $\frac{1}{f} = 27\mu\text{s}$ . La led s'allume donc pendant  $14\mu\text{s}$  puis s'éteint pendant  $13\mu\text{s}$ .

#### I.1.4 – Altitude

Pour déterminer l'altitude, les altimètres barométriques sont largement utilisés sur les avions. Ils dépendent de la variation des températures et des pressions et mesure l'altitude en partant du principe que la pression atmosphérique baisse de 1hPa (1bar = 1000hPa) chaque fois que l'on s'élève de 8mètres. L'altitude est donc donnée par la formule :

$$\text{Altitude} = [P_{\text{mesurée}} - P_0] \times 8$$

où  $P_{\text{mesurée}}$  est l'estimation de la pression donnée par le capteur et  $P_0$  la pression au niveau de la mer (101325Pa).

Pour obtenir l'altitude par rapport au sol, il faut faire une mesure de la pression avant chaque vol.  $P_0$  est ensuite remplacée par cette valeur. La formule devient :

$$\text{Altitude} = [P_{\text{mesurée}} - P_{\text{sol}}] \times 8$$

Le capteur de pression de la Fig 1.5 convertit la pression atmosphérique présente dans l'environnement de 0 à 2bar en une tension continue variant entre 0 et 5V. Une propriété de ce capteur est que la tension de sortie est proportionnelle à la pression mesurée :

$$U(V) = 0,0045 \times P - 51$$



**Figure 1. 5 : Capteur barométriques [4]**

## I.2 – Système d'alimentation

Utilisée correctement, une batterie Lithium Polymère n'est pas plus dangereuse qu'une autre batterie rechargeable. Les batteries de type « Lithium Polymère » ont une capacité poids/puissance beaucoup plus intéressante comparée aux autres types de batteries. Pour la même capacité, elles pèsent le tiers du poids d'un pack traditionnel. Toutefois les batteries LiPo nécessitent des régimes de charge différents et il y a un risque d'incendie en cas de non-respect des consignes.

Si une batterie Lithium Polymère subit un court-circuit ou une surcharge importante, les éléments internes peuvent être endommagés. Le lithium étant très réactif au contact de l'eau ou de l'humidité, si l'emballage de la batterie est défectueux, il peut se produire une émission de gaz毒ique et un incendie.

Si la batterie est manipulée avec soin et en respectant les consignes d'utilisation, il n'y a aucune probabilité d'explosion ou de feux et la durée de vie est de l'ordre de 2 ans. Les batteries Lithium Polymère ne doivent jamais être déchargées en dessous de 3 volts par élément

sous peine de les endommager. Si cela se produit, la charge suivante ne sera que partielle et les éléments ne retrouveront pas leur capacité nominale ce qui entraînera une réduction des performances.

D'autres batteries sont aussi disponibles : Lithium-phosphate (LiFePO4) dont la tension nominale est d'environ 3.3V et le rapport énergie/masse est moins favorable que sur les LiPo mais la durée de vie est supérieure.

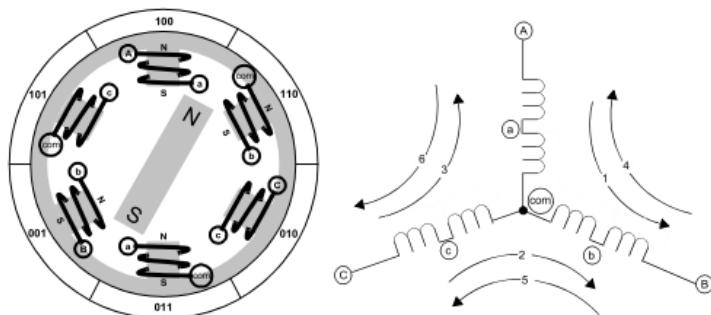
### I.3 – Système de propulsion

Les moteurs brushless sont les plus utilisés pour propulser des véhicules aériens de type minidrone. Le défaut principal des moteurs à courant continu est la présence des balais qui engendrent des frottements, des parasites et limitent la durée sa durée de vie par leur usure. Pour éviter tous ces problèmes, on a recours au moteur Brushless [5].

Ils ont l'avantage d'avoir un bon rendement, supérieur à 70% pour les petits et 90% pour les gros. L'absence de balais lui offre une durée plus longue par rapport aux moteurs continus ordinaires.

Cependant, le fait que le moteur n'a pas de balais exige d'intercaler une carte contrôleur entre l'accumulateur et les phases du moteur. La conception de ce circuit est très complexe, elle génère une tension triphasée synchrone avec la rotation du rotor. Ce dispositif est fiable mais plus onéreux que les variateurs de moteur classique.

On peut modéliser ce moteur vu sur la Fig 1.6 comme un moteur ayant un stator à 3 dents et 3 bobines connectées en étoile, et un rotor à 4 aimants permanent.



**Figure 1. 6 : configuration théorique du moteur**

### **I.3.1 - Phase de démarrage**

-On applique une tension entre deux pôles des bobines et le troisième en l'air. On attend que le rotor vienne au repos, dite la position d'équilibre.

-On commute la tension d'une des bornes vers l'autre qui était en l'air, changeant la polarité magnétique d'une dent et on attend la nouvelle position d'équilibre. On dit que 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 suites de pas sont caractérisées par les forces d'attraction et de répulsion qui apparaissent au cours de la commutation pour faire tourner le rotor.

Dans ce mode de fonctionnement, le moteur ne peut pas tourner vite car il faut assurer un temps minimum entre chaque commutation pour que le rotor finisse son mouvement. Cette phase est surtout nécessaire pour démarrer le moteur afin de créer un courant par effet dynamo qui sera ensuite capturé par un circuit combinatoire sur chaque phase.

### **I.3.2 - Phase d'accélération**

Elle consiste à mesurer la tension induite présente sur la phase en l'air lorsque le moteur aurait eu déjà assez de vitesse pour la produire. Après avoir lancé le moteur en mode pas à pas, la carte contrôleur surveille la tension induite par effet dynamo aux bornes de la phase en l'air et lorsque l'aimant cible se trouve entre les deux bobines qui le pousse et l'attire en même temps, elle lance un nouveau cycle de commutation. A chaque moment de l'évolution du rotor, les forces magnétiques sont toujours présentes ce qui lui donne un couple plus important. Les commutations suivent donc l'évolution du rotor ce qui permet une accélération au moteur.

Ces moteurs assurent tous les mouvements du drone mais cela reste insuffisant. Il faut que le drone puisse connaître ses environnements ainsi que son évolution dans l'espace par l'intermédiaire des différents capteurs embarqués.

## **I.4 – Capteurs de surveillance**

Pour s'assurer que le drone fonctionne correctement en vol, il doit se munir de divers détecteurs qui suivent instantanément son état de vol comme la température, la charge de la batterie ainsi que la vitesse de chaque moteur.

### I.4.1 – Température

Le drone est équipé de thermomètres afin de surveiller chaque moteur, ainsi que la batterie LiPo qui s'échauffe selon sa charge. Une valeur jugée trop élevée entraînera un atterrissage programmé de l'appareil pour protéger la détérioration d'un des équipements.

Le capteur DS1820 est un capteur numérique qui permet la mesure de la température sur une plage de  $-55^{\circ}$  à  $+125^{\circ}\text{C}$  avec un pas de  $0.5^{\circ}\text{C}$ . Sa configuration sur une carte électronique est simple comme le montre la Fig 1.7

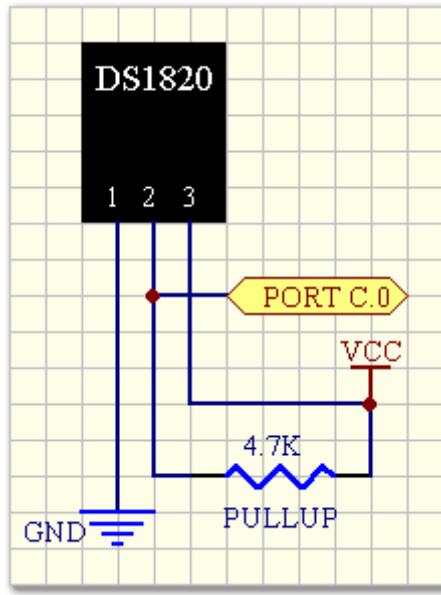


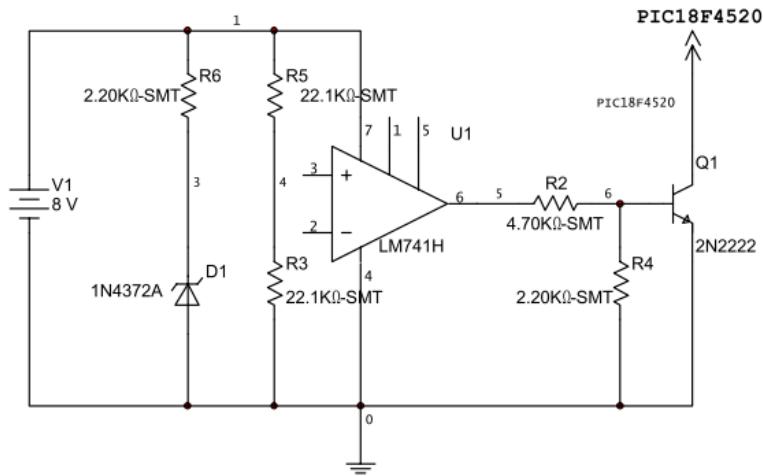
Figure 1. 7 : Montage du DS1820 [6]

Ce capteur de température s'appuie sur le protocole « one-wire » pour communiquer sa mesure. Comme son nom l'indique, un seul fil est nécessaire même si plusieurs périphériques sont utilisés.

### I.4.2 – Tension

Pour protéger l'ensemble du système d'une chute de tension, un détecteur de niveau de tension de la batterie est nécessaire afin d'arrêter le système. La détection du passage en-dessous du seuil de 5V est réalisée en comparant une tension de référence stabilisée (qui ne change pas même quand la tension de la batterie chute) avec la tension de la batterie. La stabilisation de la tension de référence est obtenue par la diode zener D1 de 3,6V, polarisée en inverse par la résistance R1. Le comparateur de tension, formé ici par un simple AOP de type LM741, reçoit cette tension de référence sur sa borne non inverseuse (broche 3). La tension non

stabilisée provenant de la batterie est quant à elle appliquée sur la borne inverseuse de l'AOP, au travers d'un pont diviseur composé de R2 et R3, qui divise la tension d'entrée par deux. Quand la tension de la batterie vaut 5V, la tension au point nodal de R2 et R3 est de 3,6V. Si la tension chute un tout petit peu en dessous de 5V, la borne inverseuse de l'AOP se retrouve avec une tension inférieure à celle présente sur la borne non-inverseuse, ce qui provoque le basculement de sa sortie de l'état bas à l'état haut. Ainsi, le transistor Q3 sera à l'état ON, allume la LED et déclenche une interruption logicielle dans le microcontrôleur central qui arrêtera l'ensemble du système. La Fig 1.8 nous montre le montage de ce circuit.



**Figure 1. 8 : DéTECTEUR de niveau Batterie faible [7]**

Dans notre cas, il faut surtout s'assurer que chaque microcontrôleur soit alimenté par une tension de 5V pour qu'ils fonctionnent correctement. En gardant les hypothèses suivantes :

$$V_z = 3,6V, R_2 = 10k\Omega, \text{Seuil} = 5V$$

On a la formule :

$$\text{Seuil} = \frac{V_z(R_2 + R_3)}{R_3}$$

On en tire que :  $R_3 = 25,7k\Omega$

#### I.4.3 – Vitesse moteur

Qui dit quadri-rotor, dit voilure tournante. Le contrôle du véhicule passe par des variations de vitesse sur les 4 groupes motopropulseurs

- A l'équilibre : même poussée sur chaque rotor.
- Pour les autres mouvements tels que : roulis, tangage, lacet

La poussée des rotors ne dépend que de la vitesse de rotation sur les 4 groupes moto-propulseurs. Ainsi les groupes moto-propulseurs sont primordiaux à dimensionner. On considère la masse du véhicule à 1,6 kg donc un dimensionnement en vol stationnaire de 400 g / rotor (portance du disque rotor). Pour réaliser le quadri-rotor il faut deux hélices avec pas à gauche et deux hélices avec pas à droite.

La formule suivante permet de calculer la vitesse de rotation de chaque rotor [8] :

$$T_h = \frac{R_{h0}ARNc\omega^2 C_z(\beta)}{6\pi}$$

$R_{h0}$  : la masse volumique de l'air

A : aire du disque rotor

R : rayon de la pale

c : Saumon

N = 2

$\omega$  : vitesse de rotation du rotor

$C_z$  : coefficient de portance des pales

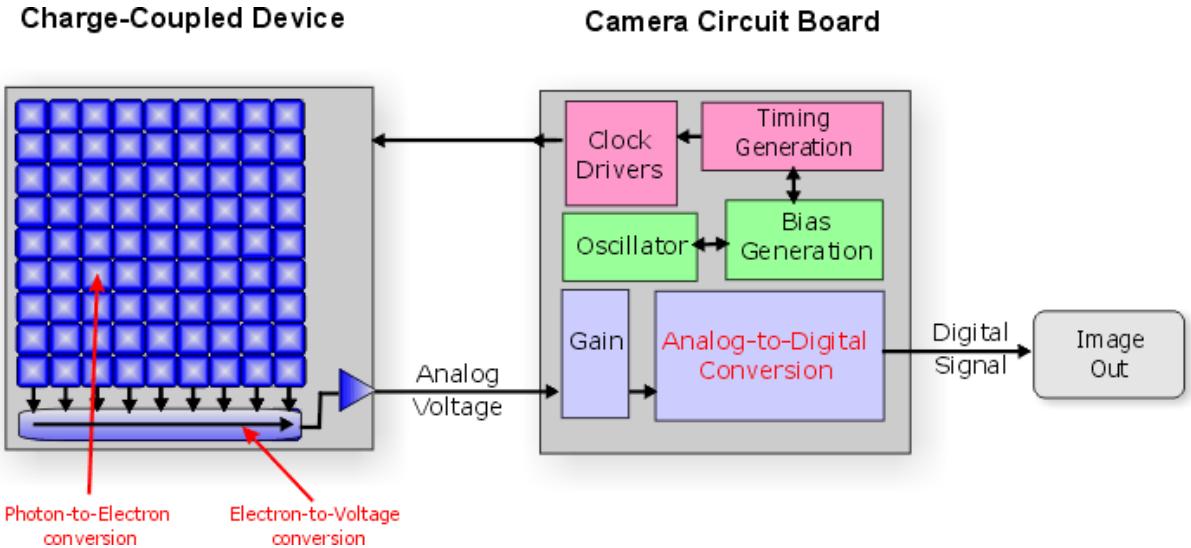
Pour des hélices de dimensions 10x4,5'', le vol stationnaire serait alors équivalent à une vitesse de rotation de 6000 tr/mn pour  $C_z = 0,1\beta$  et  $\beta = 7^\circ$

Pour faire voler donc le minidrone, il faut des moteurs qui tournent plus de 6000tr/mn pour lui permettre d'effectuer différent mouvements correspondants au pilotage.

### I.5 – Equipements de mission

Les drones sont surtout utilisés pour inspecter des zones à risque pour l'homme. Ainsi, ils doivent se munir de caméras pour observer son entourage et d'un système de communication qui lui permet de communiquer au sol.

### I.5.1 – La Caméra



**Figure 1. 9 : Fonctionnement caméra CCD [9]**

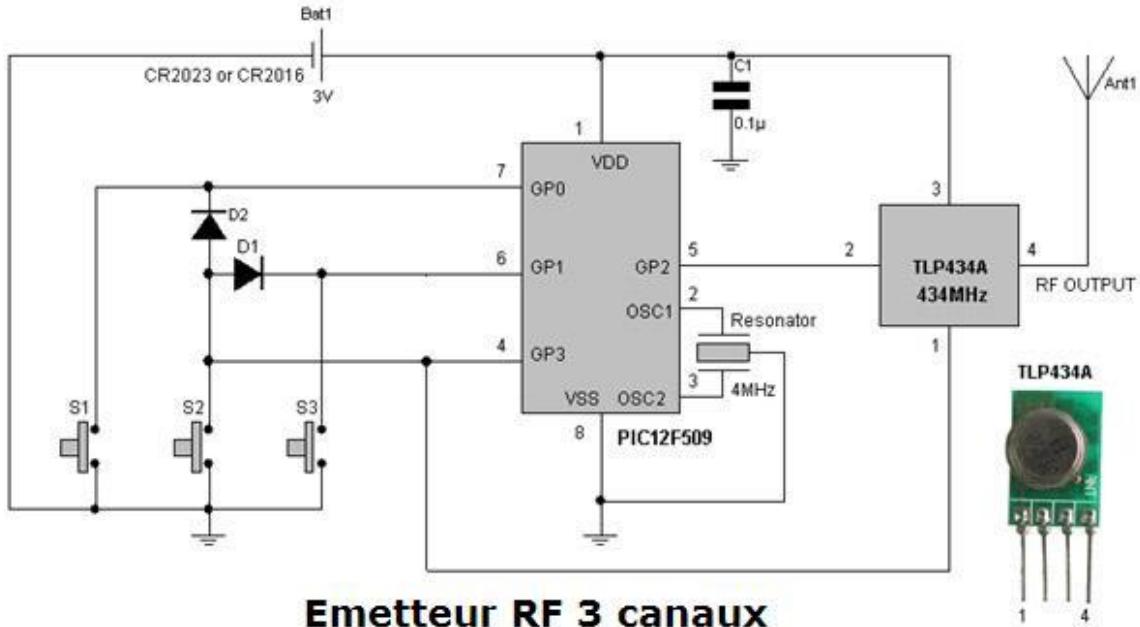
Une caméra CCD comme le montre la Fig 1.9 comprend des lignes et des colonnes, définissant les pixels. Le principe de lecture d'une CCD conduit à définir les bornes des colonnes par un dopage p gravé dans le silicium. En revanche, les bornes des lignes sont définies par une polarisation commandée. Le puits de potentiel qu'est un pixel est statique dans la phase d'acquisition du signal scientifique, puis variable pendant la lecture des pixels.

Les étapes de l'enregistrement et de la lecture d'une image CCD sont décrites ci-jointe :

- Durant la capture, les pixels se remplissent peu à peu en photo-électrons. .
- La variation du potentiel définissant les lignes du CCD conduit au transfert, ligne par ligne, vers le registre.
- Le registre transfère les photoélectrons vers la zone de lecture, pour charger un condensateur. La tension aux bornes du condensateur est ensuite amplifiée et numérisée.

### I.5.2 – Emetteur et récepteur embarqué

Ces modules sont destinés à transmettre ou recevoir des données venant du pilote. L'émetteur qu'on a utilisé pour le minidrone est un module à implémenter au sein du reste du système. Il s'agit d'un émetteur RF 3 canaux TLP434A présenté sur la Fig 1.10 avec son circuit de base.



**Figure 1. 10 : Schéma du module émetteur [10]**

Cette télécommande utilise un PIC12F509 de Microchip qui est un microcontrôleur 8 pins monopuce conçu pour les applications à faible nombre de broches. C'est un émetteur sans fil ultra petit est idéal pour les projets de télécommande ou les transferts de données à distance. Elle peut fonctionner avec une tension de 2V à 12V.

Le récepteur RF 3 canaux de type RLP434A qui est un récepteur radio compact travaille directement avec l'émetteur TLP434A sur une fréquence de fonctionnement de. Le module ainsi que le circuit correspondant est présenté sur la Fig 1.11 :

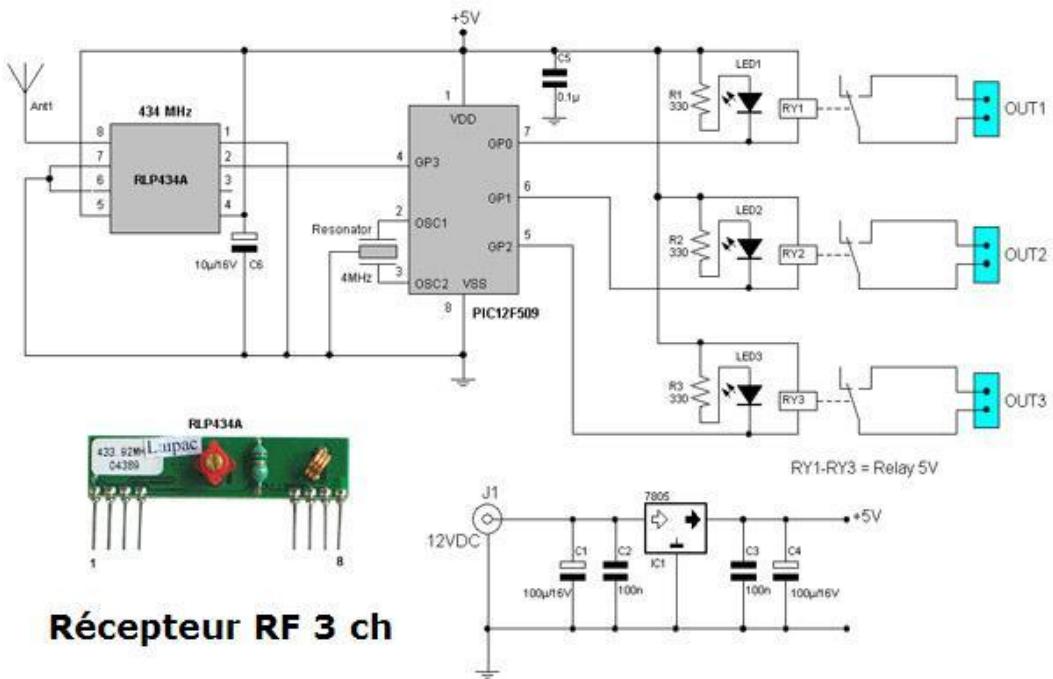


Figure 1. 11 : Schéma du module récepteur [10]

Il convient pour des débits de données allant jusqu'à 4.8KHz, et le courant typique d'opération est seulement 4.5mA.

Connaissant tous les équipements embarqués, l'architecture générale du mini-drone quadrirotor est représentée à la Fig 1.12.

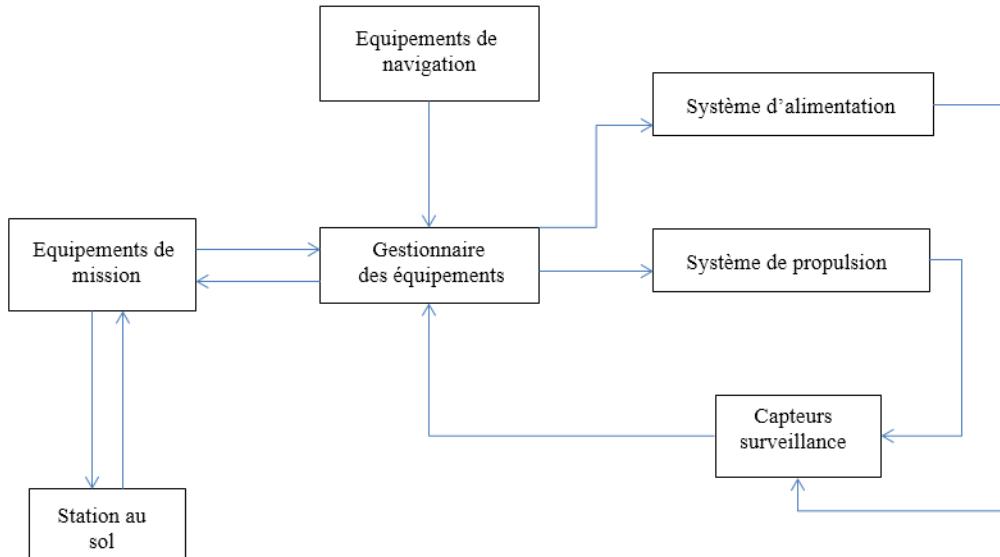


Figure 1. 12 : Architecture générale du mini drone

Le mini drone effectue son vol correctement par l'intermédiaire des équipements de navigation, de propulsion et des équipements de missions. Grâce aux capteurs, le drone rassemble donc à la fois les données dont la station sol a besoin, et celles utiles à lui-même. Son système de propulsion est spécifique par sa puissance face à sa légèreté.

Le microcontrôleur doit organiser les informations puis les envoyer au module émetteur. Le type de communication choisi pour l'émission est une transmission numérique Wi-Fi qui est en même temps utilisée pour la réception des commandes. Ainsi, pour assurer le bon fonctionnement du système, l'analyse et le traitement de ces informations seront étudiés dans le deuxième chapitre.

## CHAPTIRES II : ANALYSE ET TRAITEMENT DES DONNEES EXPLOITEES

Les équipements de navigation et de mission embarqués ne sont que des dispositifs nécessaires pour le bon fonctionnement du drone. Cependant, ils ne sont destinés qu'à soit fournir des données soit en récupérer depuis un système de contrôle embarqué qui sert à analyser et traiter ces données.

Dans ce chapitre, on va voir comment procède le système de contrôle à synchroniser toutes les tâches qui lui sont confiées suivant tous les dispositifs existants.

### II.1 - Système de drone

Un système de drone est composé de d'un ou plusieurs vecteurs aériens équipés de capteur de détection, d'une ou plusieurs stations au sol de commande et de recueil de détection et enfin de liaisons radioélectriques de données entre le vecteur aérien et la partie sol.

#### II.1.1 – Drone

Un drone ou UAV est un aéronef sans pilote à bord, télécommandé ou autonome, qui peut éventuellement emporter une charge utile destinée à des missions. Ces missions peuvent être d'ordre civil ou militaire ou autre encore en effectuant des surveillances, de recueil de renseignement, de combat, de transport.

##### a - Classification et caractéristique

Les drones quadrirotors sont classifiés parmi les cellules à voilures tournantes qui a la caractéristique de décoller et atterrir verticalement ainsi que la possibilité de faire un vol stationnaire. Un quadrirotor consiste en une armature en croix symétrique avec des moteurs et des rotors aux extrémités de chaque tige. Les rotors diamétralement opposés tournent dans le même sens. Les rotors sont généralement non articulés.

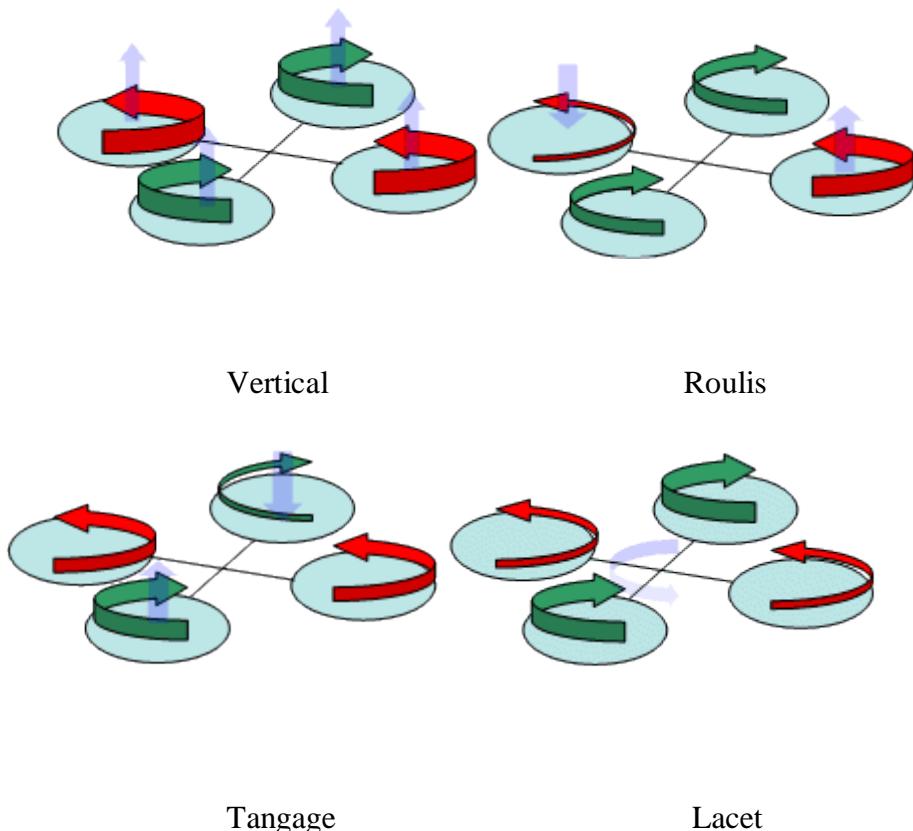


Figure 2. 1 : Type de drone quadrirotor

## b - Mouvement des quadrirotors

Le quadrirotor dispose de 4 mouvements pour se déplacer : le mouvement vertical, le mouvement de roulis, le mouvement de tangage et le mouvement de lacet.

- **Translation verticale** : le contrôle vertical se fait par variation de la vitesse de rotation des rotors. Pour un vol stationnaire, la vitesse de rotation est la même pour tous les rotors de façon à ce que la poussée globale des quatre rotors compense le poids de l'appareil. Pour changer l'altitude, on fait varier identiquement leur vitesse.
- **Translation horizontale** : on fait varier l'assiette de l'appareil, ce qui permet de modifier le vecteur translation en grandeur et en direction de chaque rotor.
- **Lacet** : le lacet s'obtient par la différence de couple de frottement créée au niveau des rotors. il suffit donc d'augmenter la vitesse de rotation de deux moteurs diamétralement opposés et diminuer la vitesse des deux autres moteurs. Ainsi la poussée globale est maintenue, mais il y a apparition d'un couple résultant.
- **Roulis et tangage** : le roulis et le tangage à altitude constante sont obtenus de la façon suivante : il faut faire varier la vitesse des moteurs des rotors diamétralement opposés. On augmente la vitesse de l'un et on diminue la vitesse de l'autre (On conserve la force globale opposée au poids). Le quadrirotor s'incline du coté où l'on a ralenti la vitesse du moteur et part dans cette direction.



**Figure 2. 2 : Les mouvements du drone**

### **II.1.2 – Opérateur**

L’opérateur est une personne ou groupe de personne qui assure la commande au sol pour orienter comme voulu le drone et effectuer ses missions. Il assure l’analyse et interprétation des données reçues et prend la décision convenable. Il doit donc munir de qualification pour mener à bien la mission.

#### **a - Surveillance du milieu d’évolution**

Lorsque le véhicule est en vol l’opérateur n’a plus de référence de la façon pour le guider. Le contrôle du véhicule par rapport à son environnement est alors primordial. Il est déjà muni de divers capteurs destiné à fournir toutes les informations nécessaires à la bonne commande. Les capteurs constituent l’une des parties cruciales pour la stabilité du véhicule car ils permettent de déterminer les paramètres d’état. La défaillance de l’un de ces capteurs peut mener à un dysfonctionnement dans la boucle de commande, à une instabilité et un accident. Par exemple, le système de localisation GPS permettant de localiser le véhicule par rapport au poste de contrôle ; le capteur de température permettant de contrôler le sur échauffement des éléments embarqués.

#### **b - Commande**

Puisqu’on parle de quadrirotor, le décollage et l’atterrissement ne peut être que verticalement. En vol, tous les mouvements définis en haut sont permis par l’opérateur pour effectuer une mission. Selon les paramètres d’états perçus par les capteurs, les commandes envoyées peuvent être soit directement transmises aux actionneurs soit remplacer par des commandes générées à la suite des résultats de ces derniers. Le principe de fonctionnement sera développé plus amplement en bas.

### **II.1.3 - Poste de commande**

Le poste de commande varie selon le type de drone utilisé. Les uns sont seulement commandés par une simple tablette ou télécommande, tandis que les autres sont commandés à partir d’un poste de control. Dans ce cas, la liaison se fait par satellite afin de permettre un vaste porté de pilotage. Si la portée de la liaison est assez faible, on peut l’accompagner par un poste de commande mobile comme celui de notre cas.

Le poste de commande est abrité dans un véhicule terrestre de type pick-up choisi surtout par la possibilité d’héberger l’installation du système de contrôle et les opérateurs au

commandes. Le drone n'est donc pas limité dans une zone de navigation puisque le poste de contrôle est mobile et peut suivre le drone en vol.

## II.2 - Format des données

Les données échangées entre le poste de contrôle et le quadrirotor est le même. On utilise le réseau wifi suivant la norme 802.11 comme moyen de transmission entre les deux entités. Les données sont envoyées sous forme de trame contenant toutes les informations nécessaires.

### II.2.1 - Données ascendantes

Les données à envoyer sont composées des commandes du drone codées sur 8 bits et des coordonnées GPS du poste mobile ayant une taille de 8 octets. Une information de 9 octets est encapsulée suivant les différents protocoles de la norme 802.11 avant d'être émise vers le drone. La latitude et la longitude sont codées en simple précision.

8 bits			32 bits			32 bits		
MSB	.....	LSB	MSB	.....	LSB	MSB	.....	LSB
Commandes			Latitude			Longitude		

**Figure 2. 3 : Format de données ascendantes**

Les caractéristiques sont détaillées dans le tableau suivant :

**Tableau I : Caractéristique des données ascendantes**

Données	Taille	Description
Commandes	8 bits	Un code de 8 bits par commande
Latitude	32 bits	<ul style="list-style-type: none"> <li>- Un bit de signe (MSB)</li> <li>- 8 bits d'exposant</li> <li>- 23 bits de mantisse</li> </ul>
Longitude	32 bits	<ul style="list-style-type: none"> <li>- Un bit de signe (MSB)</li> <li>- 8 bits d'exposant</li> <li>- 23 bits de mantisse</li> </ul>

## II.2.2 - Données descendantes

Les données de navigations et le flux vidéo forment les données descendantes. Les informations seront divisées en plusieurs trames et reconstituées à l'arrivée.

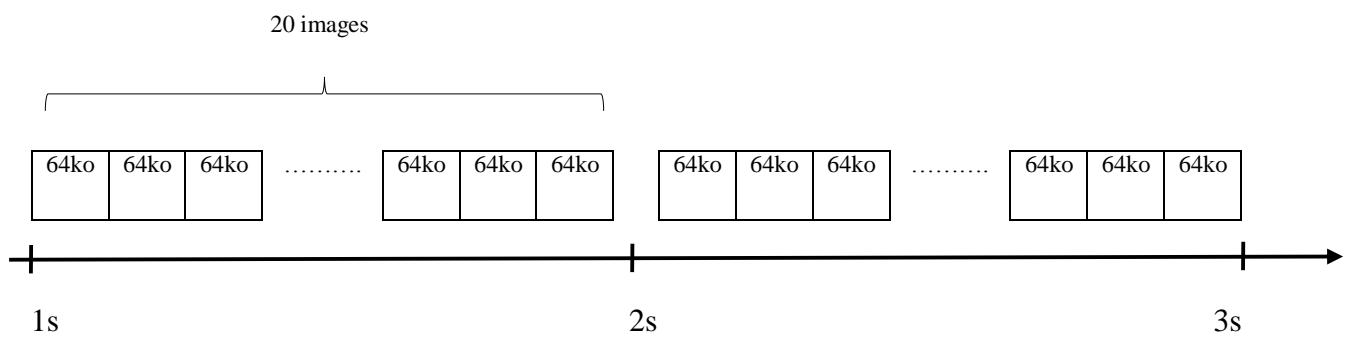
Les données de navigations sont toutes les données techniques concernant le drone en vol. Elles sont reçues à chaque seconde. Cela aide le pilote au sol et l'informe sur l'état des instructions qu'il a envoyé. La composition des données de navigations est représentée dans le tableau ci-dessous.

1 octet	8 octets	3 octets	3 octets	4 octet	1 octet	1 octet	5 octets									
Altitud e	GPS		Vitesse du drone		Valeurs angles		Vitesse moteur		Energi e	Vitess e vent	Température					
	Lat	Long	V x	V y	V z	Roulis	Tangage	Lacet	M1	M2	M3	M4	Batterie	M1	M2	M3

**Figure 2. 4 : Données de navigation**

Le flux vidéo est un ensemble de succession de 20 images par seconde que le drone renvoie au sol. Une image est donc reçue toutes les 50 ms et affichée sur IHM.

Pour information, l'image reçue est une image couleur dont la résolution est de 640x480 pixels. Le codec qui a été utilisé est le MJPEG (Motion Joint Photographic Experts Group), ce qui donne une taille voisinant les 64 ko par image.



**Figure 2. 5 : Données vidéo émises depuis le drone**

La taille et le type de données de navigation varient selon la précision voulue.

**Tableau II: Longueur des données**

Données		Longueur en octet	Type
Position du drone	Altitude	1	non signé
	Latitude	4	signé
	Longitude	4	signé
Vitesse du drone	Vx	1	signé
	Vy	1	signé
	Vz	1	signé
Valeur des angles de rotation	Roulis	1	signé
	Lacet	1	signé
	Tangage	1	signé
Vitesse des moteurs	Moteur 1	1	non signé
	Moteur 2	1	non signé
	Moteur 3	1	non signé
	Moteur 4	1	non signé
Niveau d'énergie		1	non signé
Vitesse du vent		1	non signé
Température	Batterie	1	non signé
	Moteur 1	1	non signé
	Moteur 2	1	non signé
	Moteur 3	1	non signé
	Moteur 4	1	non signé

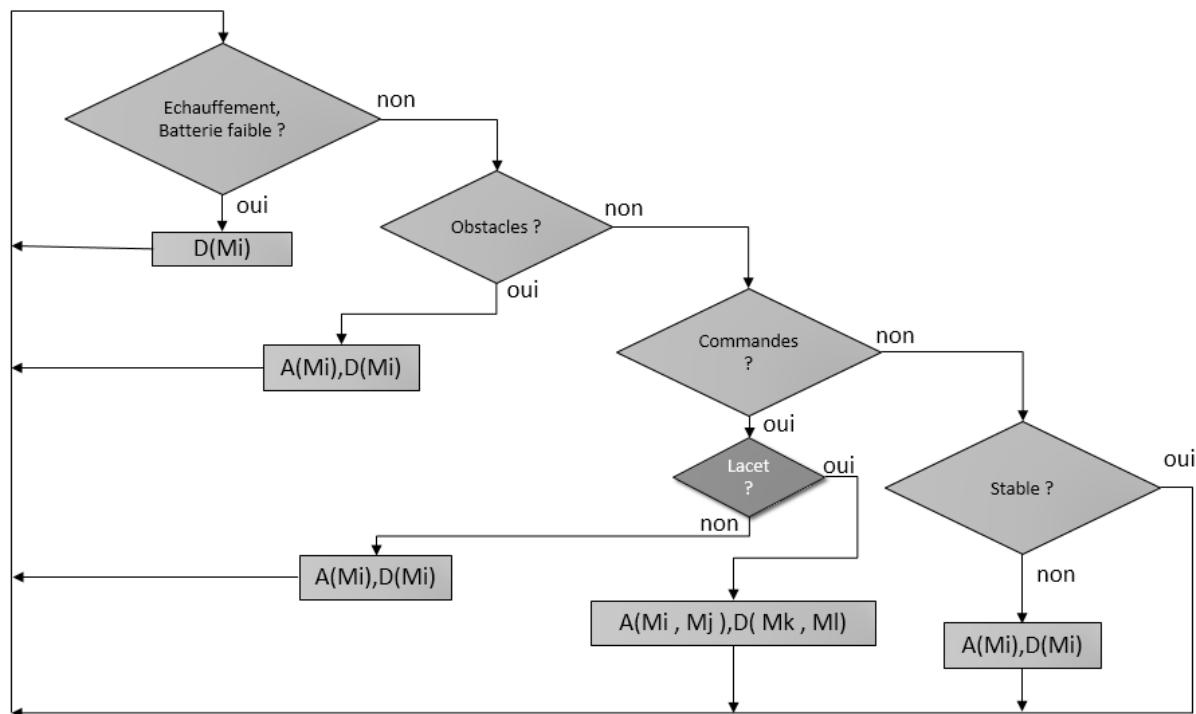
On a une information de 26 octets pour les données de navigation et une information de 64 ko pour une image. C'est à partir de ces données que l'interface façonne les informations vues sur le tableau de bord du drone.

Les cartes contrôleurs embarqués devraient être assez rapides pour analyser à la fois les données perçues du milieu de navigation ainsi que les données reçues de l'opérateur au sol. Cette rapidité est surtout nécessaire pour assurer la stabilité du véhicule et d'augmenter sa réactivité face à tout événements.

### II.3 - Traitement et Synchronisation des commandes

L'opérateur dispose de toutes les commandes possibles pendant le guidage du minidrone tant qu'il est dans la portée. On l'appelle « commande manuelle » reçue du poste au sol. Ella a une certaine limite selon les circonstances rencontrées en vol.

La procédure de « commande automatique » est alors enclenchée dans le cas où le minidrone est hors de portée, rencontre des obstacles sur son chemin, détecte des défaillances à son bord tel que des échauffements ou enfin lorsque la batterie est faible. La synchronisation des deux types de commande est figurée sur le diagramme suivant. Les exécutions des opérations se font en alternance selon un ordre de priorité étant donné que le microcontrôleur est mono tâche.



**Figure 2. 6 : Boucle d'exécution intégrée dans le système de contrôle principal**

Après que les moteurs démarrent correctement, les diverses phases de test imbriqué l'un dans l'autre s'exécute indéfiniment. Si le test est rejeté, la boucle passe au suivant, sinon elle actionne les moteurs concernés pour corriger le mouvement du drone ou pour répondre aux commandes reçues. C'est ainsi que le système fonctionne pour surveiller les opérations à exécuter par l'intermédiaire des lectures de données venant des capteurs. Cette boucle régit le drone pendant son vol

### II.3.1 - Commande manuelle

La commande manuelle est celle dont l'opérateur effectue après l'analyse et l'interprétation des données reçues depuis le drone. Etant donné que le drone n'est pas entièrement autonome celle-ci est donc toujours nécessaire pendant ses missions.

#### a - Commandes émises par l'opérateur

Le drone peut être piloté directement sur l'ordinateur à bord du poste de contrôle mobile par le biais du clavier soit à l'aide d'un joystick pour mieux manipuler les mouvement que peut effectuer le véhicule en vol comme présenté sur la Fig 2.7.

Les commandes que l'opérateur peut envoyer sont :

- Démarrage des moteurs
- Mouvement vertical (décollage)
- Vol stationnaire
- Changer angle de lacet
- Changer angle de tangage
- Changer angle de roulis
- Commande flip flop
- Arrêt des moteurs (atterrissement)



Figure 2. 7 : Joystick de pilotage de drone

## b - Décodage des commandes

Les commandes sont incluses dans le trame reçu par le drone dont la taille est de un (1) octet. Chaque commande citée ci-dessus est donc associée à une combinaison binaire de 8 bits (tableau). Le trame après avoir être reçu depuis l'antenne passe par une carte réseau afin de dissocier les trames ensuite le microprocesseur central décode chaque combinaison pour piloter l'interface de puissance directement raccorder sur chaque moteur qui va enfin interpréter le mouvement désiré. Cette chaîne est présentée sur la figure suivante :

**Tableau III : Combinaison des commandes**

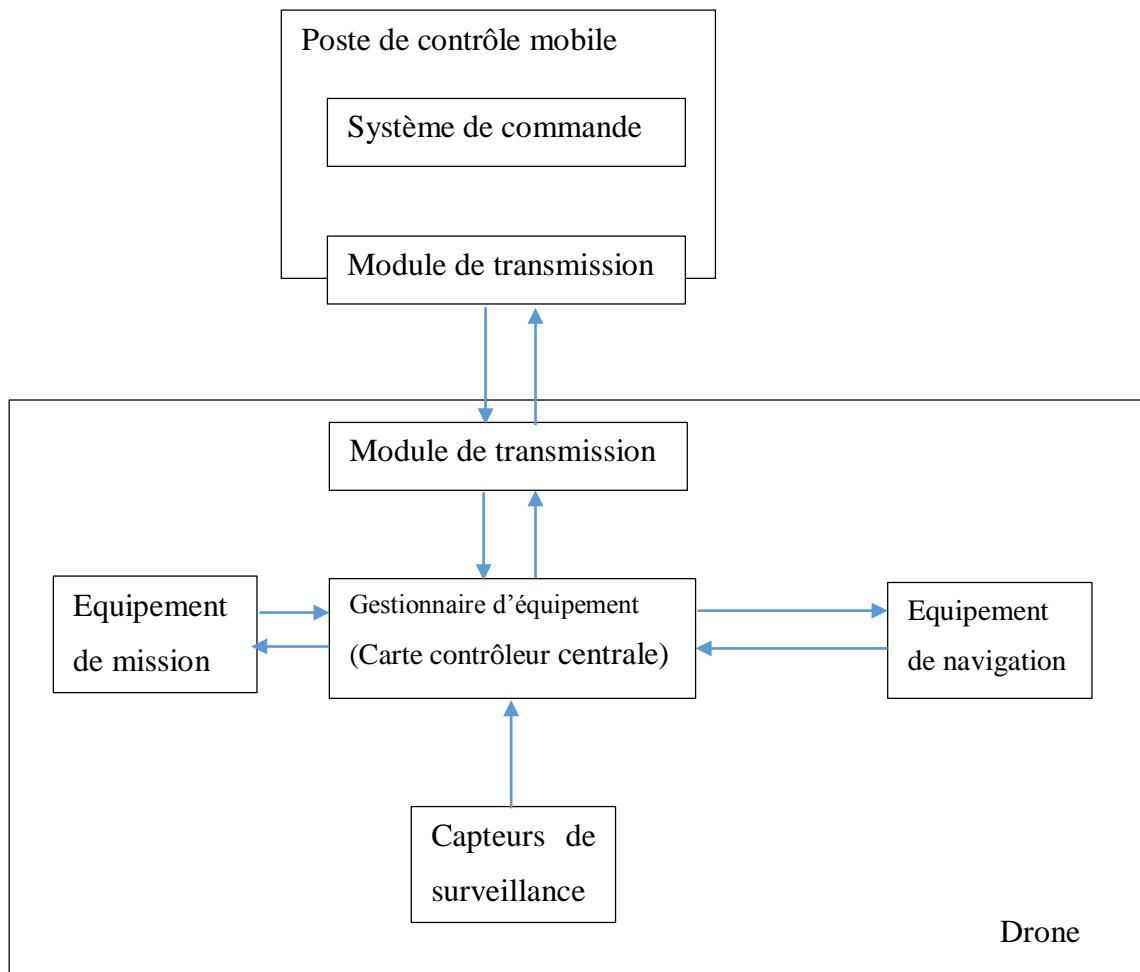
Commande	Combinaison binaires							
Nulle	0	0	0	0	0	0	0	0
Démarrage moteur	1	0	0	0	0	0	0	1
Montée	1	0	0	0	0	1	0	0
Vol Stationnaire	1	0	0	0	0	0	1	1
Lacet +	1	0	0	0	1	0	1	1
Lacet -	1	0	0	0	1	0	1	0
Tangage +	1	0	0	0	1	1	1	1
Tangage -	1	0	0	0	1	1	1	0
Roulis +	1	0	0	0	1	1	0	1
Roulis -	1	0	0	0	1	1	0	0
Flip flop	1	0	0	0	0	1	1	0
Descente	1	0	0	0	0	1	0	1
Arrêt moteur	1	0	0	0	0	0	1	0

La règle de gestion est seulement qu'on ne peut qu'envoyer une commande de 8 bits comme convenu dans le format de données échangées entre le drone et la station au sol.

### II.3.2 - Autonomie décisionnelle

Trop dépendant des commandes reçues depuis le poste de contrôle, un drone risque d'être dangereux s'il est à usage civil. Une moindre défaillance au niveau de la communication entre le drone et la station pourrait nuire au pilotage et même perdre le contrôle.

Ainsi, notre mini drone devrait être équipé d'une certaine autonomie décisionnelle face à tous événements. Il devrait être à la hauteur de connaître son état de vol instantanément ainsi que le milieu dans lequel il évolue à l'aide des équipements qui lui sont embarqués. La chaîne de traitement des données entre chaque dispositif est présentée sur la figure suivante :



**Figure 2. 8 : Enchaînement des dispositifs embarqués**

On a constaté que les commandes émises depuis la station ne seront pas forcément exécutées par le drone. La décision finale prise dépend des commandes ainsi que l'environnement où il évolue. Elle est obtenue par le traitement effectué par le microcontrôleur principal du gestionnaire d'équipement pour assurer une mission. Le drone doit donc traiter en même temps les ordres venant du sol et les informations des autres équipements embarqués, autrement dit il faut mettre en place une autonomie décisionnelle. Les éléments requis pour cette mise en place seront étudiés dans le prochain chapitre.

## **CHAPITRE III : INTERFACAGE DES EQUIPEMENTS EMBARQUES**

Les divers équipements mentionnés dans le chapitre I ne sont pas indépendants. Cela implique toujours qu'il fonctionne par module et leur vraie tâches ne se réalisent qu'ensemble. Il est alors nécessaire de concevoir une carte mère où transitent tous les flux de données entrant et sortant au système. Elle devrait savoir à quel moment effectuer une telle opération avec un ordre de priorité bien définie.

### **III.1- Généralités**

L'objectif est de concevoir un mini-drone quadrirotor pour des applications civiles. La communication entre le drone et le poste de contrôle au sol est assurée par une liaison par onde radio. Pour mener à bien une mission, le drone possède trois modes de pilotage qui se basculent entre eux selon les circonstances.

#### **III.1.1- Le pilotage manuel :**

Le drone traduit les commandes reçues depuis le poste de control et réagit comme attendu.

#### **III.1.2- Le pilotage semi-automatique :**

Ce mode est enclenché lorsqu'un ou plusieurs capteurs qui surveillent l'environnement de vol du drone détectent des obstacles. Le drone ne répond plus au pilotage manuel pendant une courte durée, le temps de surmonter l'obstacle. Dans ce cas, un calculateur préprogrammé donne les commandes pour les actions à entamer afin de dévier l'obstacle selon les données qu'il a reçu. Il est donc impératif que ces données transitent rapidement avec une bonne précision.

#### **III.1.3- Atterrissage forcé :**

Celui-ci est activé quand une anomalie de fonctionnement est détectée sur l'un des équipements embarqué tel qu'une surchauffe, la décharge des batteries. Ainsi, le drone par l'intermédiaire d'un plan de vol préétabli effectue un atterrissage.

Pour gérer les équipements embarqués, la carte mère du drone doit être composée de plusieurs modules tels que :

- Le module émetteur (Vidéo + données) sans fil pour transmettre les états de vol.
- Le module récepteur de commandes pour recevoir les commandes émises depuis le poste de control au sol.
- Le centre de traitements de commandes pour synchroniser les commandes reçues par rapports aux états de vol recueillis par le drone.
- Les circuits d'interface des moteurs qui sépare la partie à grande puissance de la partie intelligente à faible puissance.
- Et les interfaces des périphériques qui se trouvent entre chaque périphérique et la carte mère principale.

Dans la suite, nous allons voir comment ces périphériques se communiquent entre eux pour qu'ils puissent réaliser une tâche.

### **III.2- Interfaces utilisées**

Les interfaces sont surtout utilisées pour assurer la communication entre un périphérique et un équipement. Pour notre cas, l'interfaçage permet au microcontrôleur central de recevoir les données depuis les capteurs ou d'en envoyer vers les actionneurs.

#### **III.2.1- Interfaces de puissance**

Pour pouvoir stabiliser la quadri rotor, nous devons commander indépendamment la vitesse de chaque moteur. Cependant, le microcontrôleur ne peut pas commander directement ces moteurs car pour n'importe quel type de moteur, le microcontrôleur n'arriverait pas à fournir suffisamment de courant. Pour remédier à ce problème, on fait appel à une interface de puissance.

Le MOSFET (Metal Oxyde Semiconductor Field Effect Transistor) est le transistor qui nous convient, car contrairement au bipolaire, il ne nécessite qu'une différence de potentiel au niveau du gate pour commuter et laisser passer plusieurs ampères. [11]

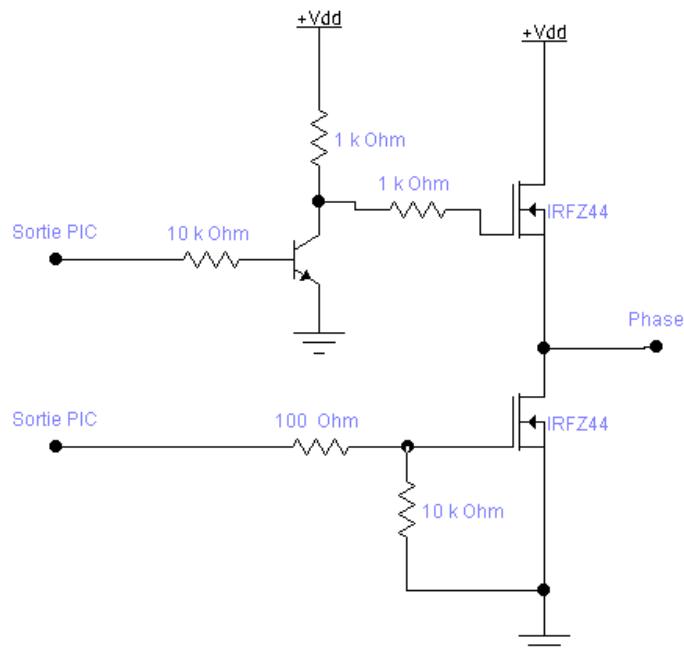
Pour ce faire, les MOSFET sont montés en 3 ponts en H qui vont s'occuper chacun d'une phase d'un moteur.

Les différentes combinaisons qui se trouvent sur les entrées de commandes se traduisent par la commutation des polarités des trois phases comme décrites dans le tableau IV.

**Tableau IV: Les 6 phases du moteur vues à la sortie de l'interface de puissance**

Séquences	Phase A1	Phase B1	Phase C1
1	A+	B flottant	C-
2	A flottant	B+	C-
3	A-	B+	C flottant
4	A-	B flottant	C+
5	A flottant	B-	C+
6	A+	B-	C flottant

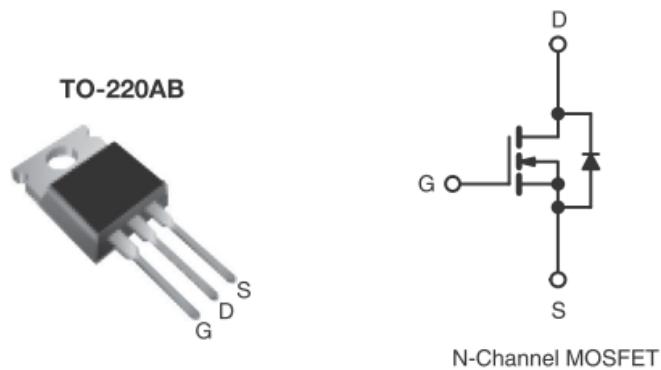
Le schéma simplifié de la Fig 3.1 nous montre les différents interrupteurs auxquels le contrôleur pilote, où A1, B1, C1 étant les trois phases du moteur à piloter.



**Figure 3. 1 : Pilotage du gate des transistors montés en pont**

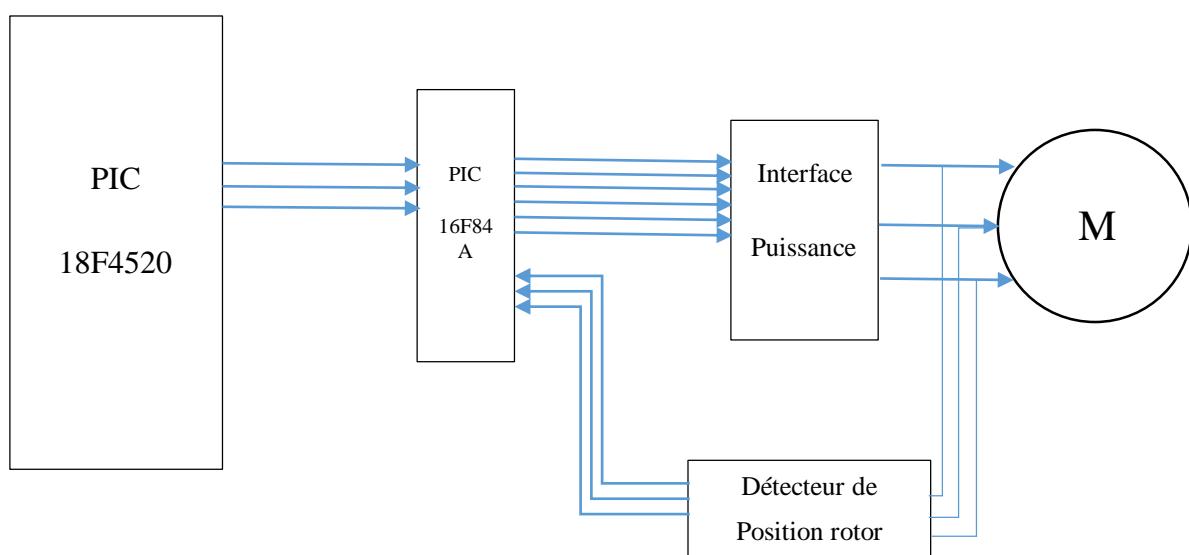
Le pilotage du gate des transistors dans chaque partie est différentes due à leur mise en série et à cause de la charge inductive créée par le moteur brushless. La figure III.3 suivante nous montre le circuit qui sert à saturer le transistor pour permettre de changer la polarité au niveau de la phase.

Le Mosfet qu'on a choisi étant l'IRFZ44 puisque la configuration de notre circuit nécessite un Mosfet de canal N et qu'on peut le trouver sur le marché local. Le brochage ainsi que son boîtier sont présentés sur la Fig 3.2



**Figure 3. 2 : Configuration du mosfet IRFZ44 à canal N**

L'assemblage de l'interface de puissance, moteur et circuit de détection de position du rotor est montré sur la Fig 3.3. Les deux cartes sont toujours en communications avec le contrôleur principal afin d'assurer la synchronisation de toutes les tâches.



**Figure 3. 3 : Schéma de principe de l'interface de puissance**

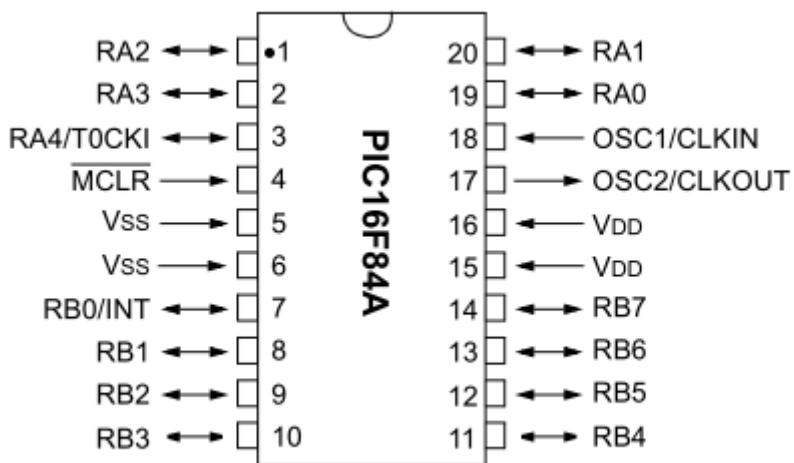
Le dispositif de détecteur de position de rotor du moteur est encore nécessaire pour pouvoir synchroniser la fréquence de commutation des phases avec la vitesse de rotation.

### III.2.2- Contrôleur de vitesse

Pour contrôler la vitesse, la carte contrôleur utilise un microcontrôleur afin d'assurer le séquencement des opérations avec un timing bien précis. Ce microcontrôleur doit comporter :

- 6 ports pour piloter une interface de puissance.
- 3 ports munis d'interruption pour un autopilotage synchrone du moteur.
- 3 ports pour communiquer avec le système central du drone afin d'agir sur la vitesse des moteurs.

Le PIC16F84 est compatible à notre application car il répond à tous nos critères cités ci-dessus. Son brochage est présenté sur la Fig 3.4.



**Figure 3. 4 : Configuration des ports du PIC16F84A**

Le tableau V indique les différents ports du microcontrôleur avec leur utilisation et les descriptions correspondantes.

**Tableau V : Les ports du microcontrôleur**

Ports	Fonctions	Description
Rb1	Pilotage de la phase 1 de la partie basse	Lorsque l'un de ces ports est mis au niveau haut, la phase correspondante sera reliée à la masse
Rb2	Pilotage de la phase 2 de la partie basse	
Rb3	Pilotage de la phase 3 de la partie basse	
Ra3	Pilotage de la phase 1 de la partie haute	Lorsque l'un de ces ports est mis au niveau haut, la phase correspondante sera reliée à la batterie mais le courant subit encore un découpage
Ra4	Pilotage de la phase 2 de la partie haute	
Rb0	Pilotage de la phase 3 de la partie haute	
Rb4	-	-
Rb5	Interruption pour synchroniser la fréquence de commutation avec la vitesse du rotor	Lorsque la valeur de l'une de ces ports change de niveau, l'interruption est levée et la phase passe à la suivante
Rb6		
Rb7		
V <sub>dd</sub>	Alimentation du microcontrôleur	La tension d'alimentation du microcontrôleur est de 5V
V <sub>ss</sub>	Masse du microcontrôleur	-
OSC1/CLKIN	Porte d'entrée pour l'oscillateur externe	-
OSC2/CLKOUT		
MCLR	Remis à zéro du PIC	-
Ra0	3 bits pour imposer une vitesse à un moteur	Les trois bits permettent une variation de la vitesse avec 8 possibilités
Ra1		
Ra2		

Le microcontrôleur est alimenté par une tension de 5V monté avec oscillateur à un quartz de 4Mhz pour générer le signal d'horloge. Le contrôleur communique avec l'interface de puissance par l'intermédiaire de 6 sorties décrites par C0 à C5. Enfin, le détecteur de position renvoie la position du rotor par les trois entrées S1 à S3. Les entrées B1 à B3 offrent les huit (08) combinaisons qui serviront de commande de vitesse au contrôleur.

### **III.2.4- Interfaces pour les capteurs**

On dispose de deux types de capteurs dans le système embarqué. Il s'agit de capteur numérique dont les détecteurs d'obstacles et de capteur analogique dont les gyroscopes sur les trois axes de la centrale inertie. Ceux numériques délivrent tout de suite un niveau logique soit un niveau bas. Ils ne nécessitent donc aucune interface entre eux et le système de commande central, car ces données sont directement exploitables. Ceux analogiques sont caractérisés par des potentiomètres linéaires qui délivrent à leurs bornes des tensions analogiques variables. Il est alors nécessaire de traiter d'abord cette donnée par un CAN afin de la discréteriser. Ce module est déjà intégré dans le microcontrôleur qu'on utilise, c'est à l'aide d'une configuration informatique qu'on procède à la conversion.

Pour que chaque module fonctionne correctement, il faut assurer l'architecture logicielle : c'est-à-dire la programmation des microcontrôleurs.

## **III.3- Architectures logicielles**

Le système embarqué est constitué par deux (02) microcontrôleurs de type 16F84A et 18F4520 [Annexe 1]. Le langage de programmation qu'on a choisi est le langage C et on a utilisé le compilateur « mikroC ».

### **III.3.1- Gestion de la carte contrôleur**

Le programme à implémenter dans le PIC16F84A doit assurer trois fonctions principales nécessaires au pilotage de l'interface de puissance à savoir:

- La commutation des tensions des phases
- La commande des gaz
- La détection de la position du rotor

### a- La commutation des tensions des phases

Le moteur brushless est une machine synchrone auto piloté, et c'est la commutation des semi-conducteurs qui gère l'alimentation du moteur. Cette commutation se fait selon le circuit de détection de position du rotor. Ce système permet une correspondance entre le flux d'excitation et l'induit évitant ainsi les décrochages éventuels du moteur. La détection d'un des six points fixes de  $60^\circ$  électrique va déclencher la commutation de l'un des transistors de l'onduleur.

Les niveaux de sortie des ports qui commandent les interrupteurs forment alors 6 combinaisons binaires décrites dans le tableau VI. Ainsi, ces 6 combinaisons se bouclent sur eux même afin de créer le mouvement de rotation du moteur.

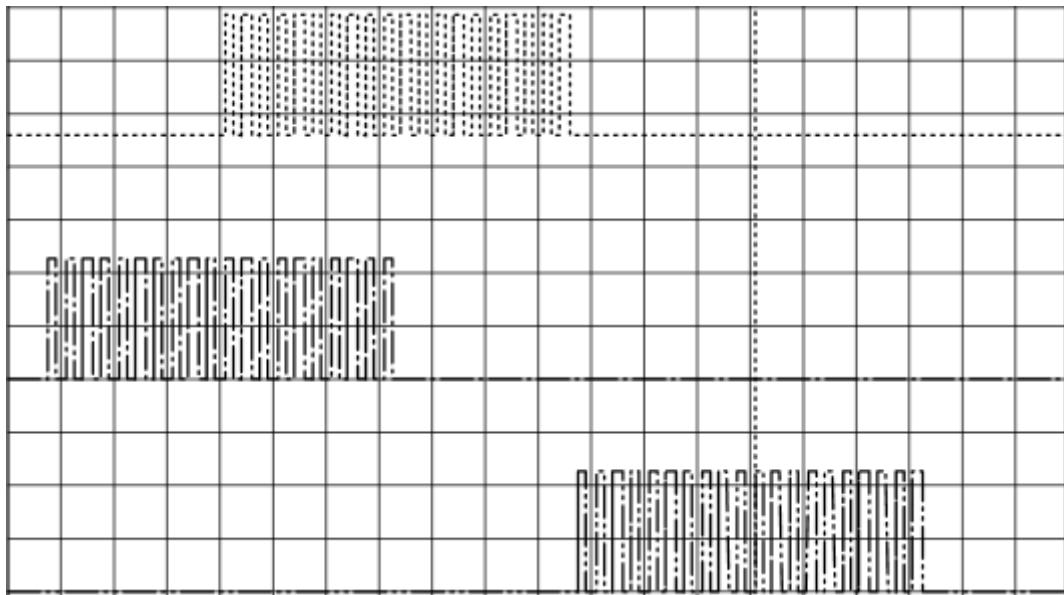
**Tableau VI : les combinaisons des 6 phases de commutation du moteur**

Niveaux	Haut			Bas		
MOSFETS	1	2	3	1	2	3
Ports	RA3	RA4	RB0	RB1	RB2	RB3
Combinaisons	0	0	1	1	0	0
	0	0	1	0	1	0
	1	0	0	0	1	0
	1	0	0	0	0	1
	0	1	0	0	0	1
	0	1	0	1	0	0

Le principe consiste à exécuter une séquence de commande de ces interrupteurs. Le niveau '1' d'un port impose la fermeture d'un transistor qui fait l'interface soit depuis une source de courant de l'alimentation, soit depuis la masse.

### b- La commande des gaz

Il reçoit la voie gaz venant du microcontrôleur principal afin de produire un découpage de la tension de la batterie pour modifier la tension moyenne appliquée au moteur. On utilise des signaux PWM ou Pulse Width Modulation. Il s'agit d'un signal carré dont la période est fixe mais dont le rapport cyclique (rapport entre l'état haut et l'état bas) change au cours du temps. Le but est d'obtenir une allure de tension semblable à celle sur la Fig 3.5.

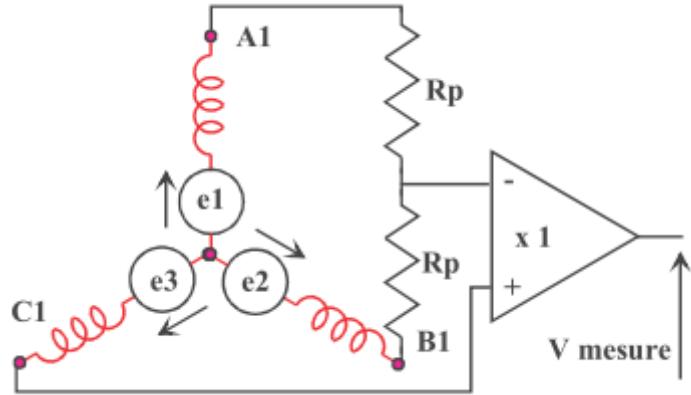


**Figure 3. 5 : Découpage du courant d'alimentation par modulation de fréquence**

Comme le microcontrôleur qu'on utilise ne dispose pas de module PWM, on devrait alors générer manuellement cette fonction. Respectant déjà les ordres des phases mentionné sur le II.2.1.a, ce que nous allons faire consiste à discréteriser les niveaux de conduction. La fonction suivante est destinée à effectuer cette tâche. L'argument « bit » reçoit la phase à exécuter et celui « haut » fixe le temps de conduction du niveau haut du signal, ce qui fera varier le rapport de cycle pour faire varier la vitesse de rotation du moteur.

### c- La détection de la position du rotor

Elle consiste à mesurer la tension induite présente sur la phase en l'air lorsque le moteur aurait eu déjà assez de vitesse pour la produire. La Fig 3.6 nous montre un simple circuit qui permet de faire cette opération.

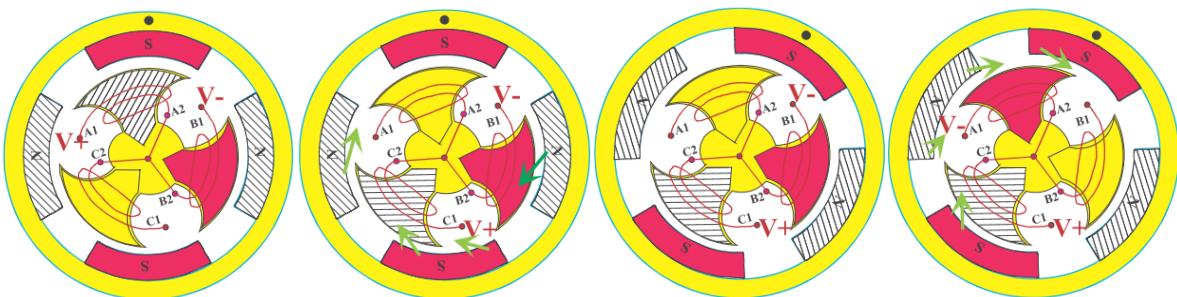


**Figure 3. 6 : Circuit de détection de position du rotor**

d-

Les tensions  $e_1$ ,  $e_2$ ,  $e_3$  sont induites dans la bobines par passage des aimants. Elles sont identiques mais seulement déphasées de plus ou moins  $120^\circ$  l'une des autres.

La commutation des différentes phases du moteur sont illustrées sur la Fig 3.7 . Les flèches montrent les forces de répulsion et de d'attraction au moment de la commutation. En gris est le pôle nord tandis qu'en rouge celui sud. Ce sont des séquences d'équilibre – commutation – équilibre - commutation.



**Figure 3. 7 : Commutations des phases du moteur [12]**

On applique une tension entre deux pôles des bobines, le troisième en l'air. On attend que le rotor prenne une position d'équilibre. On commute la tension d'une borne vers l'autre 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 ainsi tourné d'un pas. Et ainsi de suite pour produire le mouvement de rotation.

### III.3.2- Gestion du microcontrôleur central

Le système qui gère toutes les décisions est le plus complexe dans le système embarqué du drone. Chaque décision consiste à imposer un tour de chaque moteur afin de mettre le drone dans un mouvement voulu. Il s'agit d'accélérer ou décélérer un moteur. Nous allons d'abord voir comment le drone va agir sur chacun de ces quatre moteurs pour se déplacer dans l'espace.

#### a- Ordre de priorité d'exécution

La mission principale du microcontrôleur centrale est de synchroniser les commandes du pilote ainsi que les données des capteurs. Comme notre PIC est mono tâche, il faut donc exécuter toutes les opérations en alternance suivant un ordre de priorité comme illustré sur le tableau VII. Les priorités se qualifient en quatre niveaux.

**Tableau VII : Priorité des commandes à exécuter**

Périphériques	Description	Priorité
Capteur à infrarouge	Détecteur d'obstacles	2
Centrale inertielle	Contenant 3 gyroscopes sur les trois axes pour la stabilisation	4
Commande manuelle	Pilotage depuis la station au sol	3
Température embarquée	Mesure pour le risque d'échauffement	1
Niveau batterie	Détection de niveau de batterie faible	1

Cette priorité est importante pour organiser le traitement de données du drone. C'est notre programme en langage C qui ordonne ces niveaux de priorité pour pouvoir réaliser les contraintes attendues.

L'extrait de code correspondant à cette boucle principal est mis dans l'Annexe 3.

## b- Gestion des mouvements

Soit à implémenter deux principales fonctions qui servent à accélérer ou décélérer un moteur et qui prendra en argument l'identité de ce moteur.

A(Mi) : accélère le moteur Mi.

D(Mi) : décélère le moteur Mi.

H(Mi) : maintient la vitesse du moteur à sa dernière valeur

i étant le numéro d'un moteur cible.

**Tableau VIII : correspondance entre mouvement et actionneurs**

Mouvements	Interprétation	Moteur 1	Moteur 2	Moteur 3	Moteur 4
Gauche	Roulis(-)	D	A	H	H
Droite	Roulis(+)	A	D	H	H
Avant	Tangage(-)	H	H	A	D
Arrière	Tangage(+)	H	H	D	A
Rotation gauche	Lacet(-)	A	A	D	D
Rotation droite	Lacet(+)	D	D	A	A

Selon le calcul effectué au sein du microcontrôleur central, l'accélération d'un moteur se fait par l'envoi de trois bit sur le contrôleur. Le codage est indiqué sur le tableau IX. Une combinaison sert à démarrer les moteurs, une autre pour démarrer la phase d'autopilotage. Les six autres servent à varier la vitesse.

## c- Détection des obstacles

Il suffit que chaque détecteur d'obstacles délivre une tension de sortie de 0V ou 5V afin d'être considérée comme un niveau logique pour le port TOR du PIC. Le niveau de tension 5V indique qu'il existe un obstacle détecté aux environs du capteur. Les résultats obtenus servent également à stabiliser le drone.

Les détecteurs d'obstacles assurent la capacité d'autonomie décisionnelle au drone. Quand l'un d'entre eux est activé, une commande intégrée au système qui sert à esquiver l'obstacle est enclenché. Elle est prioritaire par rapport aux autres instructions de commandes.

La durée de cette phase dépend de la distance et de l'envergure de l'obstacle.

**Tableau IX : résultat de traitement du module IR**

Environnement	Réaction du drone
Obstacle à gauche	Lacet +
Obstacle à droite	Lacet -
Obstacle avant	Roulis -
Obstacle arrière	Roulis +

#### **d- Traitement des informations du capteur de température**

Le capteur de température DS1820 s'appuie sur le protocole one-wire pour communiquer sa mesure. En raison de la simplicité d'utilisation et du large déploiement de ce capteur, les commandes pour le faire fonctionner et le contrôler ont été implémentées dans la librairie « One\_Wire ». Il y a au total 3 fonctions :

- Ow\_Reset permet de reseter le capteur – OW\_Reset
- Ow\_Read permet la reception des données - OW\_Read
- Ow\_Write permet d'envoyer des commandes au capteur – OW\_Write

Le protocole de communication One\_Wire permet d'utiliser plusieurs capteurs sur un même fil. Chaque capteur est alors identifié par un code sur 64-bits. Etant donné que nous n'utilisons qu'un seul capteur de température ici, la procédure est ici simplifiée. Pour lire une température et l'afficher, il faut suivre les étapes suivantes :

- Envoi de la commande CONVERT\_T au capteur (mesure de la température)
- Envoi de la commande READ\_SCRATCHPAD au capteur (placement de la température dans le buffer du capteur)
- Lecture du buffer

- Affichage

Avant de transmettre les commandes CONVERT\_T et READ\_ SCRATCHPAD, il faudra au préalable envoyer au capteur un pulse de reset (OW\_Reset) puis la commande SKIP ROM pour adresser tous les capteurs sans avoir recours au code ROM d'indentification.

#### Décodage de la température

A la lecture, le code renvoyé par le capteur est un code binaire sur 16 bits (les 8 premiers marquent le signe) :

LS BYTE	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MS BYTE	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>	2 <sup>-1</sup>
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
	S	S	S	S	S	S	S	S

S = SIGN

**Figure 3. 8 : Format des registres de température**

Avant d'afficher la température, il est donc indispensable de décoder la suite binaire. Pour cela, nous utilisons la relation donnée dans la datasheet du capteur :

**Tableau X : Relation entre température et données**

Température (°C)	Sortie digitale (Binaire)	Sortie digitale (Hexadécimale)
+85.0	0000 0000 1010 1010	00AAh
+25.0	0000 0000 0011 0010	0032h
.0.5	0000 0000 0000 0001	0001h
0.0	0000 0000 0000 0000	0000h
-0.5	1111 1111 1111 1111	FFFFh
-25.0	1111 1111 1100 1110	FFCEh
-55.0	1111 1111 1001 0010	FF92h

Après lecture du code binaire, la variable de lecture contient donc une suite de 16 bits. Les 8 premiers (MSB) nous permettent de savoir si la température est positive ou négative. Si elle est négative, il faudra déterminer le complément à 2 des 8 derniers bits (LSB). Ceci étant fait, vous remarquerez que le dernier bit permet de déterminer la partie décimale. La

résolution étant de  $0.5^{\circ}\text{C}$ , s'il est égal à 1, alors le 10<sup>ème</sup> de  $^{\circ}\text{C}$  est égal à 5 et s'il est égal à 0, alors le 10<sup>ème</sup> de  $^{\circ}\text{C}$  est égal à 0. Les 7 bits restants correspondent à la partie entière.

L'architecture finale de la carte mère embarquée incluant tous les composants essentiels est présentée sur la figure ci-dessous. Les deux microcontrôleurs assurent la synchronisation des données et le pilotage des moteurs brushless.

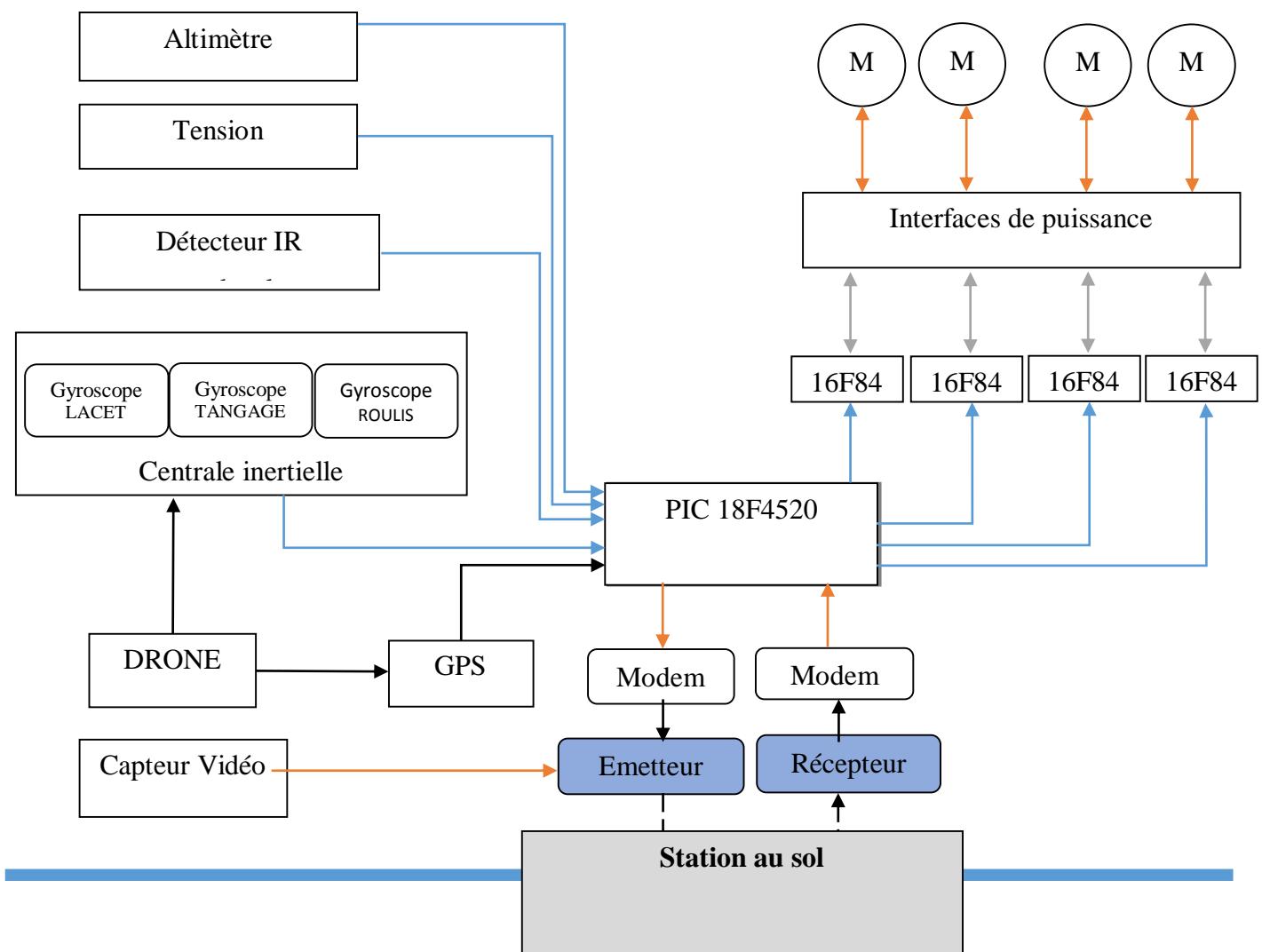


Figure 3. 9 : Architecture générale des équipements

Dans ce chapitre nous avons vu la conception de la carte mère centrale qui gère toutes les actions à effectuer pour une commande émise par un opérateur ou pour une circonstance de vol perçue par les équipements embarqués du mini drone. La partie intelligente est assurée par un programme écrit en langage C et pilote ainsi la partie moteur par l'intermédiaire d'une interface qui permet ainsi le control général du mini drone

## CONCLUSION

Depuis quelques années, le drone est devenu un outil aérien accessible à tout le monde. Son utilisation est de plus en plus fréquente du fait qu'il est très facile à piloter, et qu'il peut offrir diverses informations selon les besoins. Le travail que nous avons réalisé dans ce mémoire révèle le fonctionnement interne d'un drone quadrirotor.

Dans le premier chapitre, nous avons parlé des équipements dans le drone. Nous avons étudié le système de propulsion qui crée le mouvement du drone, ainsi que les autres dispositifs utiles à la mission.

Dans le deuxième chapitre, nous avons développé le mode de traitement de données que le drone perçoit dans son milieu soit celles reçues de l'opérateur par synchronisation effectuée par une architecture logicielle intégrée.

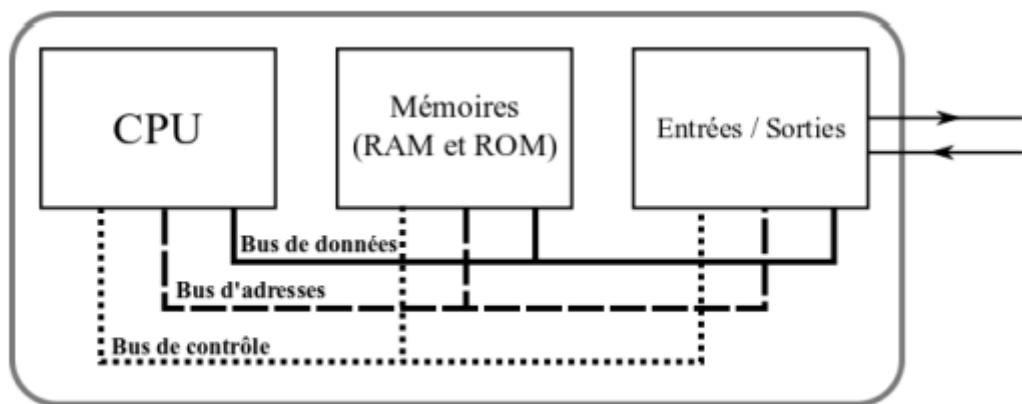
Dans le troisième chapitre, nous avons conçu une carte mère qui effectue tous les traitements. Toutes les interfaces utilisées entre les microcontrôleurs et les autres dispositifs ont été étudiées. Les programmes d'exploitation des données sont développés avec le langage C.

Pour améliorer la performance du mini-drone, on peut remplacer les microcontrôleurs de la carte mère par des micro-processeurs. Ainsi, la vitesse de traitement des données serait plus rapide et offrant plus de précision dans la prise de décision du système.

## ANNEXE 1 : Spécificité des Microcontrôleurs

### ❖ Architecture interne d'un système à Microprocesseur

Tout système à base de microprocesseur peut être représenté par le schéma ci-dessous :



CPU : Central Processor Unit

**Figure A. 1 : Architecture d'un système à microprocesseur**

Un système est caractérisé par :

- Le type de CPU : Pentium, 68010, Power PC, Microcontrôleur
- Les fréquences d'horloge.
- Les capacités mémoires RAM et ROM.
- Le nombre et la nature des dispositifs d'entrées sorties.
- La taille des bus de données (8, 16, 32, 64)
- La taille des bus d'adresse (16, 24, 32, 64).
- Le type des bus d'extension : PCI, PCM, G64, G96, VME, SCSI, AGP
- Les systèmes d'exploitation utilisés : MSDOS, Windows, Linux, UNIX, OS9

Il existe sur le marché deux type de processeur dont :

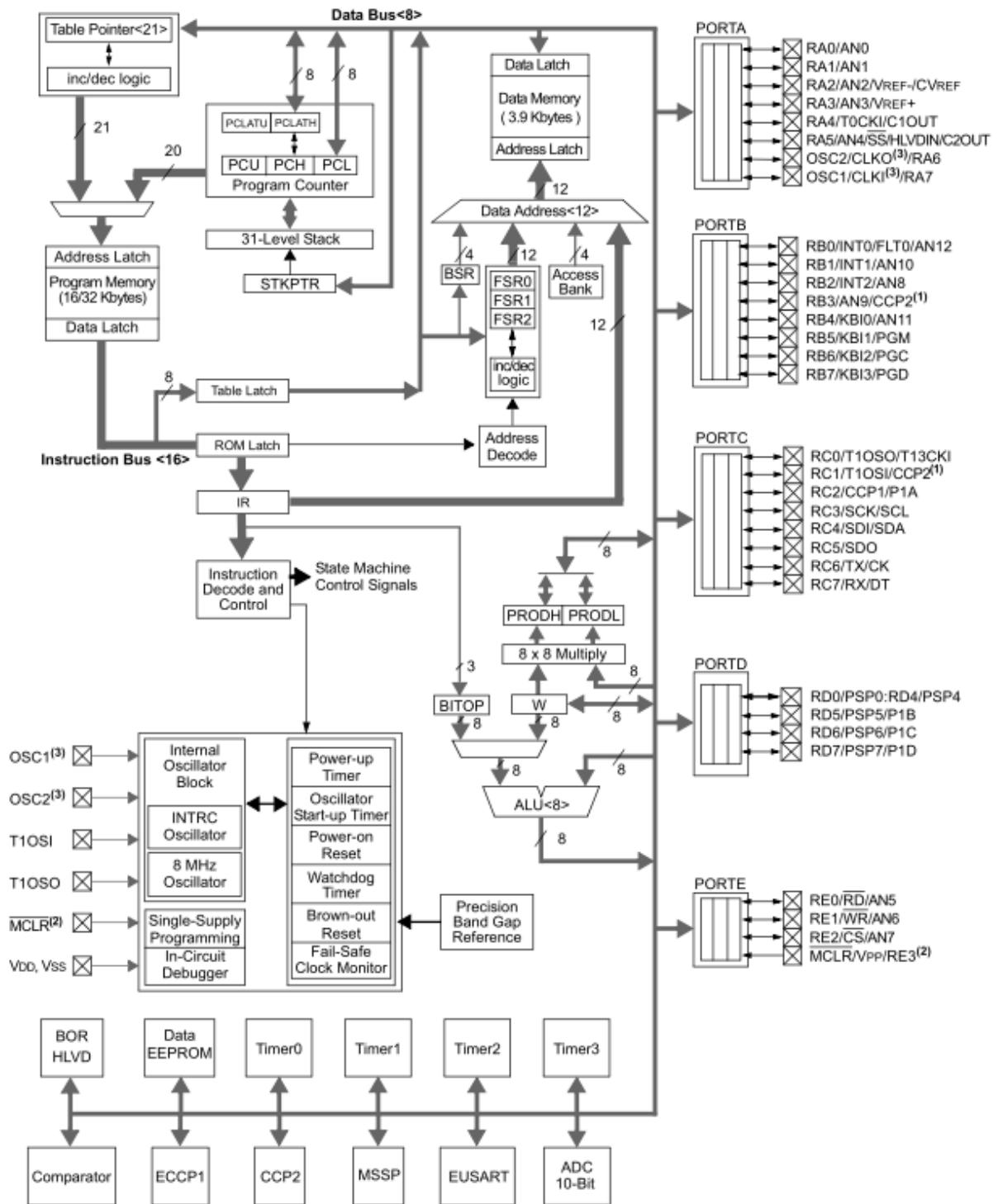
- **Les CISC : Complex Instruction Set Computer**

Peu de registres internes, Jeu d'instructions très complet.

- **Les RISC: Reduced Instruction Set Computer**

Beaucoup de registres internes, Jeu d'instructions très réduit.

## ❖ Architecture générale du PIC18F4520



**Figure A. 2 : Architecture interne du PIC18F4520**

Pour mettre au point le module CAN du microcontrôleur PIC18F4520, il faut poursuivre des étapes selon les spécificités du besoin d'utilisation.

Le microcontrôleur 18F4520 est la partie centrale de notre carte mère. Toutes les opérations entrantes ou sortantes de notre drone y passent toutes. Il interface les différents éléments embarqués. Le critère de choix de notre microcontrôleur repose sur les différents types de périphérique. Tout d'abord, les commandes de pilotage transitent dans le système du drone par le biais du récepteur HF qui, après quelques opérations délivre des signaux PWM au microcontrôleur.



**Figure A. 3 : Boîtier du PIC18F4520**

❖ **Les étapes à mettre en œuvre**

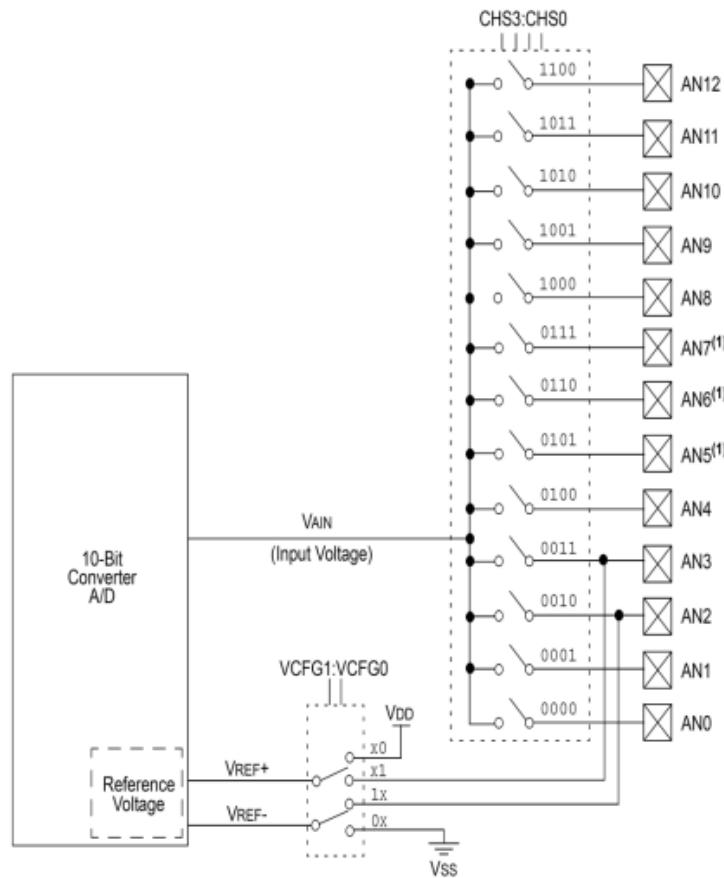
- Initialisation des paramètres du convertisseur.
- Sélection de la voie de conversion.
- Attente de la stabilisation de l'échantillonneur bloqueur.
- Lancement d'une conversion.
- Attente de fin de conversion.
- Lecture des données.

❖ **Extrait de code pour le traitement analogique du PIC18F4520**

```
/*----- Initialisation du CAN */
void init_can(void)
{
    ADCON1 = 0b11000100; /* RA3=AN3 analogique */
    ADCON0 = 0b01011001; /* CAN=activé */
    TRISA = 0b00001011; /* RA0, RA1 et RA3 en entrée RA2, RA4 à RA7 en sortie */
}

/*----- Utilisation du CAN */
unsigned int convertir(void)
{
    Delay10TCYx(12); /* attente 12x10x0,4=48us */
    ADCON0bits.GO = 1; /* lancement conversion */
    while(ADCON0bits.GO); /* attente fin de conversion */
    return (ADRES); /* retour du résultat */
}
```

## ❖ Brochage du module CAN du PIC18F4520



**Figure A. 4 : Exemple de branchement d'un module CAN d'un PIC**

Les principaux registres à configurer pour spécifier la fonction précise d'une porte CAN sont : **ADCON0, ADCON1, ADCON2** caractérisées par un octet chacun :

ADCON0 REGISTER							
U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON
bit 7				bit 0			

ADCON1 REGISTER							
U-0	U-0	R/W-0	R/W-0	R/W-0 <sup>(1)</sup>	R/W <sup>(1)</sup>	R/W <sup>(1)</sup>	R/W <sup>(1)</sup>
—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
bit 7				bit 0			

ADCON2 REGISTER							
R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0
bit 7				bit 0			

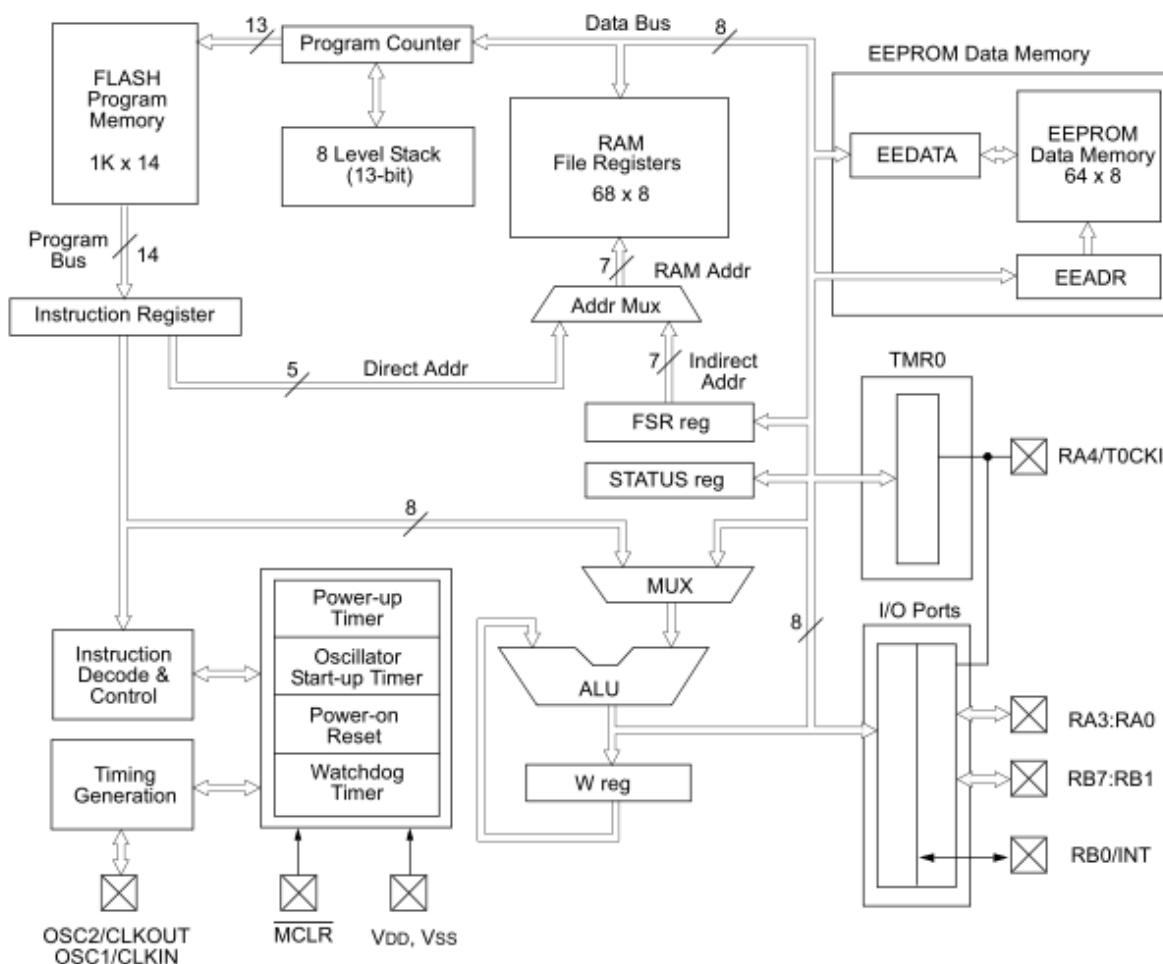
**Figure A. 5 : Caractéristique de chaque registre**

## ❖ Architecture du PIC16F84A

Pour bien connaître ce microcontrôleur, on doit décortiquer cette référence 16F84A :

- 16 indique qu'il s'agit d'un PIC mid-range
- F indique une mémoire de type flash
- 84 est la clé unique pour identifier le microcontrôleur

Un 16F84-04 est donc un PIC Mid-Range (16) dont la mémoire programme est de type FLASH (F), réinscriptible de type 84 et capable d'accepter une fréquence d'horloge de 4MHz en théorie. Son architecture interne est présentée sur la figure ci-dessous :



**Figure A. 6 : Architecture interne du microcontrôleur 16F84A**

## ANNEXE 2 : Moteurs brushless de type outrunner

On appelle « outrunner » les moteurs brushless 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 brushless outrunners) 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 brushless outrunners 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 brushless outrunners sont souvent utilisés dans des applications qui nécessitent un fort couple, car ils peuvent être reliés à la charge sans nécessiter de dispositif de réduction. Leur coefficient Kv est relativement faible par rapport aux autres types de moteurs brushless. Les principales applications des moteurs brushless outrunners sont les suivantes : ventilateurs, moteurs de disques durs, Cd-rom, moteurs de vélos électriques (intégrés dans le moyeu), bateaux ou avions radio commandés.



**Figure A. 7 : Moteur brushless outrunner**

## ANNEXE 3 : EXTRAIT CODE SOURCE

Le code source a été écrit avec le langage C du compilateur « mikroC ». Ce bout de code sert essentiellement à piloter le moteur brushless précédemment mentionné.

### ❖ Initialisation des variables, fonction de temporisation et phase de démarrage du moteur

```
#define RB0 PORTB.F0
#define RB1 PORTB.F1
#define RB2 PORTB.F2
#define RB3 PORTB.F3
#define RB4 PORTB.F4
#define RB5 PORTB.F5
#define RB6 PORTB.F6
#define RB7 PORTB.F7
#define RA0 PORTA.F0
#define RA1 PORTA.F1
#define RA2 PORTA.F4
#define RA3 PORTA.F3
#define RA4 PORTA.F2

void tempo (int temp)
{
    int i=0;
    for(i=0;i<temp;i++)
        Delay_1ms();
}

void init_mot (unsigned char )
{
    int i;
    unsigned char temp,bit_val;
    unsigned char BIT[] = {0b001100,0b001010,0b100010,0b100001,0b010001,0b010100};
    for(i=0;i<6;i++)
    {
        if(INTCON.RBIF==0)
        {
            bit_val= BIT[i];
            regul_moteur(bit_val,temp);
            INTCON.RBIF=1;
        }
    }
}
```

### ❖ Boucle principale du programme de pilotage moteur

```
void main()
{
    unsigned char temp,bit_val;
    int i,compteur;
    //      unsigned char BIT[] = {0b001100,0b001010,0b100010,0b100001,0b010001,0b010100};
    TRISA= 0b00111;
    TRISB= 0b11110000;
    INTCON.GIE=1;
    INTCON.RBIE=1;
    OPTION_REG=0b00001000;

    //*****INITIALISATION DU MOTEUR*****
    temp=3;
    while(compteur<4000)
    {
        init_mot(temp);
        compteur++;
    }
    temp=7;
    while(compteur<2000)
    {
        init_mot(temp);
        compteur++;
    }
    //***** *****
    *****

    for(; ;)
    {
        temp=10; // doit être inférieur à 20 et modifié par Pic18 (ra0,ra1,ra2)
        // Pour regler le rapport cyclique
        init_mot(temp);
    }
}
```

## REFERENCES

- [1]: <http://www.google.com/patents/EP1579176B1?cl=fr>
- [2]: <http://sweetrandomscience.blogspot.com/2013/07/comment-les-telephones-portables-et.html>
- [3]: [http://lvps87-230-19-121.dedicated.hosteurope.de/mediawiki/index.php?title=Activité\\_6:\\_Détecenter\\_les\\_obstacles\\_avec\\_des\\_déetecteurs\\_infrarouges](http://lvps87-230-19-121.dedicated.hosteurope.de/mediawiki/index.php?title=Activité_6:_Détecenter_les_obstacles_avec_des_déetecteurs_infrarouges)
- [4]: <http://www.directindustry.fr/prod/keller/capteurs-pression-piezoresistifs-membranes-ceramique-affleurants-13762-513691.html>
- [5]: [http://electronics-diy.com/electronic\\_schematic.php?id=303](http://electronics-diy.com/electronic_schematic.php?id=303)
- [7]: <http://sonelec-musique.com>
- [8] ; « La propulsion électrique des modèles réduits d'avions et de planeurs » G. Brocard
- [9]: <http://benoit.dubois2.free.fr/Xbird/>
- [10]: [http://fr.wikipedia.org/wiki/Capteur\\_photographique](http://fr.wikipedia.org/wiki/Capteur_photographique)
- [11] : F. de Dieuleveult, « Electronique appliquée aux hautes fréquences », 2<sup>ème</sup> édition, Dunod, Paris, 1999
- [12] : <http://odt55.pagesperso-orange.fr/THEORIE%20DOC/MONFORD%20THEORIE>
- [13] : <http://www.becoz.org/these/memoirehtml/ch05s03.html>, Modulations multiporteuses, 2013
- [14]: <https://shop.chipcad.hu/oldweb/hirkepek/>

**Auteur :** ANDRIAMBELO Frangio Lazareh

**Titre :** Traitement de données dans un drone

**Nombre de pages :** 51

**Nombre de figures :** 29

**Nombre de tableaux :** 10

**Résumé :**

L'objectif du présent mémoire est de concevoir tous les équipements nécessaires à un minidrone de type quadrirotor pour pouvoir être piloté facilement du sol par un module émetteur récepteur embarqué. La gestion de ces équipements a été centralisée au niveau d'un microcontrôleur programmé avec le langage C pour synchroniser les tâches d'ensemble afin d'assurer ses missions.

**Mots-clés:** Quadri rotor, OFDM, Wi-Fi, Microcontrôleur, Moteur brushless, capteur

**Abstract:**

This memory aim at designing all useful amenities for a quadrotor to be flown easily ground by an onboard transceiver module. The management was centralized at a microcontroller programmed with C to synchronize all tasks to ensure its missions.

**Keywords:** Quadrirotor, OFDM, Wi-Fi, Microcontroller, BLDC, sensor

**Encadreur:** Monsieur HERINANTAINA Edmond Fils

**Adresse de l'auteur:**

- Lot VB 72 EDA Ambatoroka secteur 3 ANTANANARIVO 101

**Contacts:**

- Tél: 034 05 236 81
- E-mail: f\_lazareh@yahoo.com