



جامعة تونس المنار

Université de Tunis El Manar



المدرسة الوطنية للمهندسين بتونس

école nationale d'ingénieurs de Tunis

Département Génie Électrique

Projet de Fin d'Études

Présenté par

**Amal JEMAI**

Pour l'obtention du

**Diplôme National d'Ingénieur en Génie Électrique**

**Conception d'un module d'acquisition de données  
d'un cerf-volant pour les éoliennes aéroportées**

Réalisé au

**LAGEPP**

Soutenu le 10 octobre 2024

Devant le Jury :

Président	: M <sup>me</sup> Nahla KHRAIEF
Rapporteur	: M. Khaled JELASSI
Encadreur Organisme d'accueil	: M. Tanguy SIMON M. Fabien MIEYEVILLE
Encadreur ENIT	: M. Joseph HAGGÈGE

Année universitaire 2023/2024

## Remerciements

Je tiens à exprimer ma gratitude à toutes les personnes qui ont contribué de loin ou de près à la réussite de ce projet.

Je souhaite au premier lieu d'exprimer ma reconnaissance envers mon encadrant Monsieur Tanguy SIMON pour sa disponibilité et son encadrement de qualité pendant toute la durée de stage. Je tiens aussi à remercier Monsieur Fabien MIEYEVILLE d'avoir veillé à ce que le projet soit réalisé dans les meilleures conditions possibles.

Mes remerciements s'adressent également à mon encadrant universitaire, Monsieur Joseph HAGGÈGE qui n'a jamais cessé de m'encourager et de me conseiller tout au long de ce stage et de ma formation à l'ENIT.

Je remercie énormément toute ma famille, en particulier mon père et ma soeur pour leurs encouragements tout au long mon parcours académique, ainsi que mes amis pour leur présence et leur soutien.

## Résumé

L'énergie des éoliennes aéroportées est un domaine émergent qui vise à exploiter l'énergie éolienne à haute altitude à l'aide d'aéronefs attachées. Dans ce rapport, nous allons présenter la conception d'un boîtier de capteurs pour faire l'acquisition des données pour les dispositifs volants de ces éoliennes, tels que les accélérations, les vitesses angulaires, la position et les données environnementales. Étant donné que le pilotage des aéronefs dans l'air est l'un des grands défis auxquels ces systèmes sont confrontés, ces données sont cruciales pour étudier le comportement de ces systèmes et valider leurs modèles dynamiques, ce qui nous permettra de confirmer par la suite des lois de commande robustes et fiables. Ce travail a été fait dans le cadre d'un projet plus général qui porte sur la conception d'un prototype complet d'éolienne aéroportée qui sera publié en open-source.

**Mots clés :** Éolienne aéroportée, acquisition de données, modèles dynamiques, lois de commande, énergie.

## Abstract

In this work, we present the development of a data acquisition system designed for a soft kite within an Airborne Wind Energy System (AWES). The system is designed to record essential flight data, such as positional data, angular velocities, and angular accelerations, which are necessary for validating and improving dynamic models of the kite. To optimize control strategies and comprehend the kite's behaviour in different wind conditions, these data points are essential. Real-time data transmission to the ground station is possible with our data acquisition system, guaranteeing prompt analysis. The availability of accurate and timely data is essential for advancing the performance and reliability of AWES, making our system an important tool in the ongoing development of this innovative technology.

**Key Words :** Airborne Wind Energy, kite, Data acquisition, dynamic models, control strategies.

# Table des matières

<b>Table des figures</b>	<b>vii</b>
<b>Liste des tableaux</b>	<b>ix</b>
<b>Introduction générale</b>	<b>x</b>
<b>1 Contexte général</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Présentation de l'organisme d'accueil . . . . .	1
1.2.1 Présentation générale . . . . .	1
1.2.2 Organigramme . . . . .	2
1.2.3 Équipes de recherche . . . . .	2
1.3 Présentation du sujet . . . . .	3
1.3.1 Principe de fonctionnement général . . . . .	7
1.3.2 Problématique . . . . .	8
1.3.3 Etude de l'existant . . . . .	8
1.3.3.1 Systèmes ground-gen . . . . .	8
1.3.3.2 Systèmes fly-gen . . . . .	13
1.3.4 Présentation du notre projet/Synthèse . . . . .	14
1.4 Conclusion . . . . .	15
<b>2 Etude préliminaire</b>	<b>16</b>
2.1 Introduction . . . . .	16
2.2 Etude du modèle de système . . . . .	16
2.2.1 Le système des coordonnées . . . . .	17
2.2.2 Modèle dynamique du cerf-volant . . . . .	17
2.2.2.1 Steering model . . . . .	18
2.2.2.2 Kinematic model . . . . .	18
2.3 Cahier des charges du système . . . . .	19

2.3.1	Les données à acquérir . . . . .	19
2.3.2	Exigences opérationnelles . . . . .	19
2.4	Calcul de la force de traction générée par le cerf-volant . . . . .	20
2.5	Relation entre les données acquises par le cerf-volant et les paramètres du modèle de Wood . . . . .	23
2.6	Conclusion . . . . .	26
<b>3</b>	<b>Réalisation et tests</b>	<b>28</b>
3.1	Introduction . . . . .	28
3.2	Description de la solution . . . . .	28
3.2.1	Architecture générale du système . . . . .	28
3.2.2	Flux de données . . . . .	30
3.3	Choix du matériel . . . . .	30
3.3.1	Microcontrôleur . . . . .	31
3.3.2	Capteurs . . . . .	31
3.3.2.1	IMU . . . . .	31
3.3.2.2	GPS . . . . .	32
3.3.2.3	Capteur de vitesse de vent de Holybro . . . . .	32
3.3.2.4	Jauges de contrainte . . . . .	33
3.3.2.5	Convertisseur ADC en I2C . . . . .	34
3.3.2.6	Télémetrie . . . . .	34
3.3.3	Alimentation . . . . .	35
3.4	Implémentation logicielle . . . . .	35
3.4.1	Environnement de travail . . . . .	35
3.4.1.1	L'autopilote PX4 . . . . .	36
3.4.1.2	QGroundControl . . . . .	36
3.4.2	Programmation et configuration . . . . .	36
3.4.2.1	IMU . . . . .	36
3.4.2.2	GPS . . . . .	38
3.4.2.3	Convertisseur ADC en I2C . . . . .	38
3.4.2.4	Enregistrement des données (Data logging) . . . . .	42
3.4.2.5	Télémetrie . . . . .	43
3.5	Tests . . . . .	43
3.6	Conclusion . . . . .	45
	<b>Conclusion générale</b>	<b>48</b>

<b>Références Bibliographiques</b>	<b>49</b>
<b>Annexe 1</b>	<b>52</b>
.1 Détails technologiques des systèmes présentés dans les tableaux de la partie "étude de l'existant" du premier chapitre : . . . . .	52
.1.1 Systèmes ground-gen . . . . .	52
.1.2 Systèmes fly-gen . . . . .	53
<b>Annexe 2</b>	<b>54</b>
.2 Les principales caractéristiques des capteurs à partir des fiches techniques .	54
.2.1 Accel/Gyro à 3 axes : BMI055 . . . . .	54
.2.2 Accel/Gyro à 3 axes : ICM-42688-P . . . . .	54
.2.3 GPS M8N . . . . .	55
.2.4 Jauge de contrainte WIKA F2812 . . . . .	55
<b>Annexe 3</b>	<b>57</b>
.3 Installation de Git . . . . .	57
.4 Mise à jour firmware PX4 à l'aide de QGroundControl . . . . .	57
.5 Les paramètres de "SD Logging" qui existe dans QGroundControl . . . . .	58

# Table des figures

1.1	Organigramme du laboratoire LAGEPP, [19] . . . . .	2
1.2	(a) Système Ground-gen, (b) Système Fly-gen, [5] . . . . .	3
1.3	(a) LEI SLE (Leading Edge Inflatable, Supported Leading Edge) Kite; (b) LEI C-kite; (c) Foil Kite, design from Skysails, [5] . . . . .	4
1.4	(d) Glider, design from Ampyx Power; (e) Swept rigid wing, [5] . . . . .	4
1.5	Semi-rigid wing, design from Kitegen, [5] . . . . .	5
1.6	Les différents catégories des systèmes AWE selon le nombre de câble, [5] . . . . .	5
1.7	(a) Avion avec quatre turbines, conçu par Makani Power. (b) Avion composé d'un cadre d'ailes et de turbines, conçu par Joby Energy. (c) Aérostat de levage toroïdal avec une éolienne au centre, conçu par Altaeros Energies. (d) Quadrirotor à suspension statique en autorotation, conçu par Sky WindPower, [5] . . . . .	6
1.8	Au dessus; La phase de génération, au dessous; La phase de reprise, [23] . . . . .	7
1.9	Classification des prototypes à génération d'énergie au sol des entreprises, source : [5] . . . . .	9
1.10	Les prototypes des systèmes Fly-gen, source : [5] . . . . .	13
2.1	L'angle de cap, $\gamma$ , et la position $\mathbf{p}$ dans le repère $(e_x, e_y, e_z)$ , en relation avec l'angle $\theta$ , l'angle de l'azimuh $\phi$ , et la longueur de câble, $r$ ., [25] . . . . .	18
2.2	Les forces aérodynamiques ressenties par le cerf-volant en vol parallèle au sol [8] . . . . .	20
2.3	le repère $(X, Y, Z)$ qui est le même $(e_x, e_y, e_z)$ dans la figure 2.3, l'angle, $\theta$ , l'angle de l'azimuh, $\phi$ , et le repère $(L_N, L_E, L_D)$ , , [11] . . . . .	24
2.4	Vue du plan tangent au sphère et perpendiculaire au rayon $r$ , en représentant les repères C et L . . . . .	25
3.1	L'architecture du boîtier de capteurs . . . . .	29
3.2	Pixhawk 6C . . . . .	31
3.3	GPS M8N de Holybro . . . . .	32

3.4	capteur de vitesse de vent de Holybro . . . . .	33
3.5	Jauge de contrainte WIKA F2812 . . . . .	34
3.6	ADS1115, convertisseur ADC en I2C . . . . .	34
3.7	Sik Telemetry Radio V3 de Holybro . . . . .	35
3.8	Régulateur de tension 5V . . . . .	35
3.9	calibration de l'accéléromètre à travers QGroundControl, [21] . . . . .	37
3.10	processus de calibration de l'accéléromètre, [21] . . . . .	37
3.11	calibration de GPS, [22] . . . . .	38
3.12	État de GPS dans MAVLink console . . . . .	39
3.13	Le fichier default.px4board . . . . .	40
3.14	Arrêter l'ADC par défaut . . . . .	41
3.15	L'état de l'ADS1115 dans le console de MAVLink . . . . .	41
3.16	Dossier des topics personnalisés . . . . .	42
3.17	Exemple des topics personnalisés . . . . .	43
3.18	Module de télémétrie connecté à l'autre module pour échanger les données	43
3.19	Montage expérimental pour les tests . . . . .	44
3.20	Le système d'acquisition de données au cours des tests . . . . .	45
3.21	Les vitesses angulaires acquises par l'accéléromètre . . . . .	46
3.22	Les accélérations linaires acquises par le gyroscope . . . . .	47
23	Accéléromètre : BMI055 . . . . .	54
24	Gyroscope : BMI055 . . . . .	54
25	Accéléromètre : ICM-42688-P . . . . .	55
26	Gyroscope : ICM-42688-P . . . . .	55
27	GPS M8N . . . . .	55
28	Jauge de contrainte WIKA F2812 . . . . .	56
29	Première étape pour la mise à jour de PX4 . . . . .	57
30	Deuxième étape pour la mise à jour de PX4 . . . . .	58

# Liste des tableaux

1.1	Tableau pour les prototypes Fly-gen des institutions académiques . . . . .	10
1.2	Tableau pour les prototypes ground-gen des institutions académiques . . .	11
1.3	Tableau pour les prototypes Fly-gen des institutions académiques . . . . .	12
1.4	Tableau pour les prototypes fly-gen des institutions académiques . . . . .	14

# Introduction générale

Dans le paysage dynamique de la transition énergétique, l'exploration de nouvelles voies vers une production d'énergie plus efficace et plus durable est devenue une priorité.

L'énergie éolienne est depuis longtemps considérée comme une source d'énergie renouvelable prometteuse pour répondre à la demande croissante d'électricité tout en réduisant les émissions de gaz à effet de serre. Traditionnellement, les éoliennes terrestres et marines constituent la principale technologie d'exploitation de cette ressource naturelle.

Ces systèmes éoliennes sont confrontés à plusieurs défis majeurs, notamment celui à atteindre des altitudes élevées. De plus les dépenses élevées liées à la construction et la maintenance ainsi que l'utilisation importante des matériaux représentent également des obstacles significatifs.

C'est là qu'interviennent les aérogénérateurs volants, en anglais ” **Airborne Wind Energy systems** ”. Contrairement aux éoliennes conventionnelles qui se basent sur des structures fixes au sol ou en mer, ces systèmes exploitent le potentiel énergétique du vent en utilisant des dispositifs aéroportés tels que des cerfs-volants ou des drones, lesquels sont capables de se déplacer à différentes altitudes pour capter des vents plus forts et plus constants.

Ces systèmes ont également l'avantage remarquable de consommer jusqu'à 90 % moins de matériaux que les éoliennes classiques. En effet, la diminution de l'empreinte matière ouvre de nouvelles opportunités pour une transition énergétique plus respectueuse de l'environnement.

À partir des bases établies par des travaux théoriques publiés à la fin des années 1970 et au début des années 1980, plusieurs prototypes ont été développés depuis les années 2000. Les différentes évolutions ont été distinguées en fonction du type d'aéronef (flexible, semi-rigide ou rigide), du nombre de câbles (de 1 à 3), ainsi que de la localisation des machines électriques et des convertisseurs de puissance (à bord ou au sol). Alors que les années de 2000 à 2010 ont été marquées par une augmentation du nombre de recherches et développements portant sur différents concepts.

Il est difficile de piloter de manière autonome les éoliennes aéroportées en raison de leur environnement dynamique et incertain. Les capteurs fiables jouent un rôle crucial dans la précision des modèles dynamiques utilisés pour stabiliser et optimiser ces systèmes. En l'absence de ces capteurs, il serait compliqué de garantir un contrôle autonome efficace, ce qui pourrait compromettre la performance et la sécurité des éoliennes.

Notre projet consiste alors à concevoir le boîtier de capteurs d'un cerf-volant pour les éoliennes aéroportées dans le but d'acquérir des données réelles sur ces systèmes, ce qui va mener à comprendre sa dynamique et pouvoir valider ses modèles. Ce qui apporte encore plus de valeur à ce projet est notre dévouement envers l'open-source. En rendant le module accessible au public, notre objectif est de favoriser et d'accélérer davantage de recherches dans ce domaine des systèmes d'énergie éolienne.

Dans un premier temps, nous présentons le contexte général de notre projet en détaillant l'entreprise d'accueil et une étude bibliographique sur les différents systèmes d'éoliennes aéroportées.

Ensuite pour le deuxième chapitre, nous abordons une étude théorique pour comprendre le modèle de notre système et pour pouvoir réaliser le boîtier des capteurs.

Et finalement, nous terminons avec le troisième chapitre par une présentation de la partie de la réalisation des deux côtés matériel et logiciel et une partie de tests afin d'effectuer une validation expérimentale préliminaire.



# Chapitre 1

## Contexte général

### 1.1 Introduction

Dans ce chapitre nous allons introduire le contexte général du projet en question tout en procédant à un état de l'art pour les systèmes AWE. Nous examinerons par la suite les différentes solutions existantes dans la littérature ou en milieu industriel pour situer notre projet par rapport aux travaux existants.

### 1.2 Présentation de l'organisme d'accueil

#### 1.2.1 Présentation générale

Le Laboratoire d'Automatique, Génie des Procédés et Génie Pharmaceutique (LAGEPP) est affilié à l'Université Claude Bernard Lyon 1 et au Centre National de la Recherche Scientifique. C'est un laboratoire pluridisciplinaire couvrant les domaines du génie des procédés, de l'automatique, de l'ingénierie des produits, de l'ingénierie pharmaceutique et de la physicochimie.

Le laboratoire vise à conduire des projets de recherche multidisciplinaires en réunissant des chercheurs issus de diverses disciplines, tout en s'assurant de maintenir un niveau élevé de qualité scientifique dans chacune de ces disciplines.

Les activités de recherche se concentrent principalement sur deux thématiques scientifiques majeures :

- Procédés physicochimiques et procédés en milieux complexes, dispersés et changeants (procédés de production de nanoparticules, cristallisation, lyophilisation).
- Modélisation dynamique, observation et contrôle des processus.

Le LAGEPP est organisé en quatre équipes scientifiques qui rassemblent des chercheurs

et des enseignants-chercheurs spécialisés dans les trois domaines disciplinaires principaux : l'automatique, le génie des procédés et le génie pharmaceutique. Les équipes sont soutenues par les collaborateurs du service commun, qui est structuré en fonction des compétences : service Administratif et Financier, service Ingénierie de Projets, service Analyse et Instrumentation et service Informatique.

## 1.2.2 Organigramme

Dans cette partie, nous présentons l'organigramme de l'entreprise d'accueil qui décrit la structure hiérarchique et fonctionnelle des équipes, permettant une répartition claire des responsabilités et des rôles.



FIGURE 1.1 – Organigramme du laboratoire LAGEPP, [19]

## 1.2.3 Équipes de recherche

Il existe principalement quatre équipes de recherche qui sont les suivantes :

- Dynamique, Commande et Observation des Procédés
- Génie Pharmaceutique
- Procédés des milieux dispersés
- Systèmes non linéaires et procédés

## 1.3 Présentation du sujet

Contrairement aux éoliennes classiques, ces systèmes ne font pas appel à des structures statiques pour exploiter l'énergie du vent, mais utilisent les forces aérodynamiques pour suivre une trajectoire périodique dans l'air, au moyen d'un contrôle automatique. Ce concept a pour effet de diminuer l'emploi de matériaux tels que le béton et l'acier, tout en rendant la construction, le transport et l'installation plus simples, ce qui se traduit par des dépenses de capital plus faibles et des réductions des impacts environnementaux.

Chaque système AWE est composé généralement d'un dispositif volant, un boîtier à capteurs, un ou plusieurs câbles, un système de génération d'énergie et un système de contrôle pour l'aéronef. Il existe principalement deux catégories de systèmes en fonction de la façon dont l'énergie électrique est produite au sol ou sur l'aéronef, voir figure 1.2. Nous allons détailler ces deux catégories par la suite.

- **Systèmes de génération de l'énergie au sol (Ground-gen systems en anglais) :**  
Dans ce type de système l'énergie électrique est produite au sol par des travaux mécaniques effectués par la force de traction, transmis de l'aéronef au système au sol par un ou plusieurs câbles, qui entraîne une génératrice électrique. La plupart des prototypes conçus par les chercheurs sont "Ground-gen".
- **Systèmes de génération de l'énergie sur l'aéronef (Fly-gen systems en Anglais) :**  
pour cette catégorie de système, l'énergie électrique est produite à bord de l'aéronef par des hélices et petites génératrices puis elle est transmise au sol par un câble spécial qui transporte des câbles électriques.

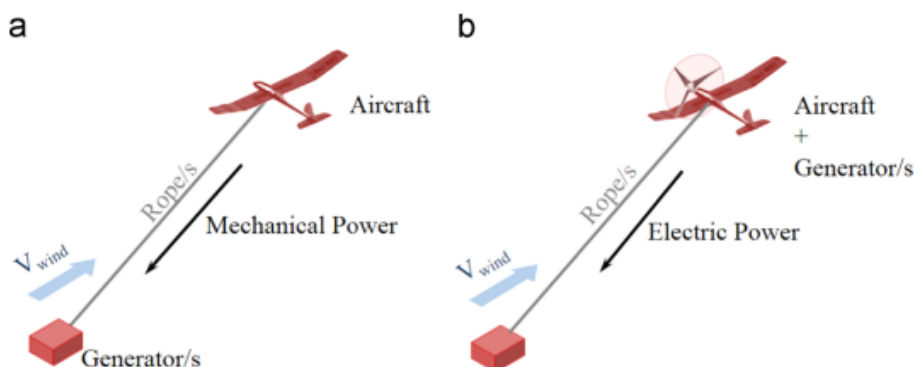


FIGURE 1.2 – (a) Système Ground-gen, (b) Système Fly-gen, [5]

Pour optimiser la production de l'énergie, il faut maximiser la force de traction de dispositif volant. C'est la raison pour laquelle la majorité des prototypes pilotent l'aéronef

pour qu'il vole en **"mouvement transversal"** ou en anglais **"crosswind motion"**, Ce terme fait référence à une trajectoire de vol qui s'étend principalement sur une surface perpendiculaire au flux du vent.

Pour éviter de tordre le câble au cours du vol, le dispositif volant suit une trajectoire en huit qui est une forme particulière de "mouvement transversal".

Chacun des systèmes ground-gen ou fly-gen se divise en différentes catégories selon d'autres critères. Nous allons détailler ça dans ce qui suit.

Les systèmes de génération de l'énergie au sol se diffèrent essentiellement dans les types d'aéronefs. Il existe deux catégories, Les aéronefs à ailes rigides et les aéronefs à ailes flexibles. La figure 1.10 montre les principaux prototypes des aéronefs à ailes flexible.

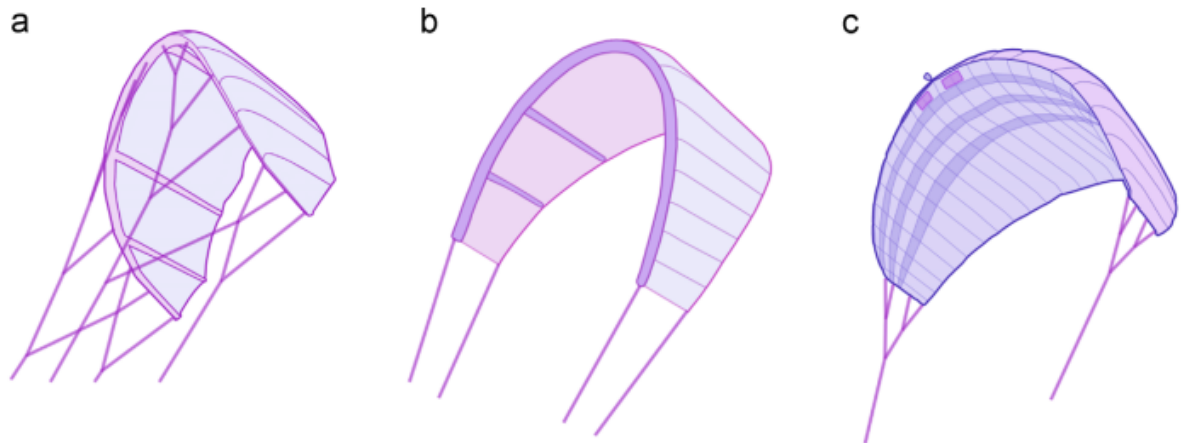


FIGURE 1.3 – (a) LEI SLE (Leading Edge Inflatable, Supported Leading Edge) Kite ; (b) LEI C-kite ; (c) Foil Kite, design from Skysails, [5]

La figure 1.4 illustre deux catégories des aéronefs à ailes rigides.

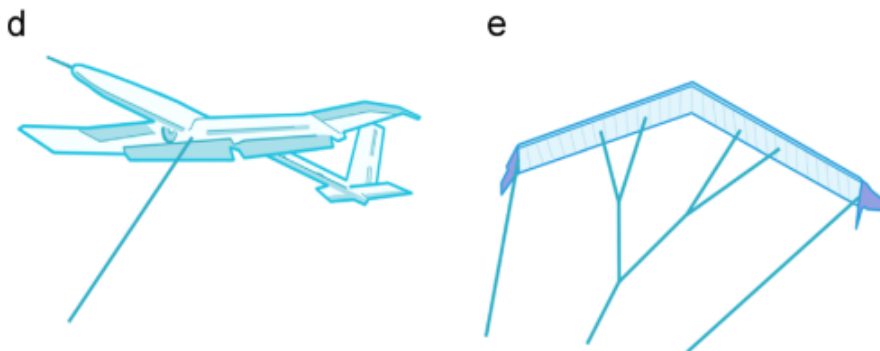


FIGURE 1.4 – (d) Glider, design from Ampyx Power ; (e) Swept rigid wing, [5]

Il existe un autre type développé par une entreprise qui s'appelle **Kitegen** qui est semi-rigide comme le montre la figure 1.5.

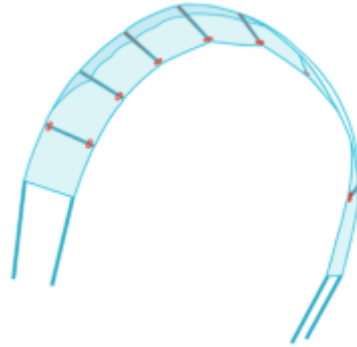


FIGURE 1.5 – Semi-rigid wing, design from Kitegen, [5]

Ces systèmes se distinguent en outre par le nombre de câbles qui attachent les dispositifs volants au sol. Il existe essentiellement trois catégories.

- **Un seul câble** : l'aéronef est attaché à un seul câble avec une poulie et un moteur qui tire le câble dans un sens ou un autre pour changer le mouvement de l'aéronef dans l'air. Puisque le pilotage de l'aéronef ne peut pas être assuré par un seul câble, ces systèmes nécessitent des actionneurs à bord du dispositif volant ou du boîtier de commande sous l'aéronef, tel est le cas pour le cerf-volant.
- **Deux câbles** : On va toujours agir sur la longueur du câble pour déplacer le dispositif volant. Ces systèmes permettent de piloter le dispositif volant depuis le sol mais sans pouvoir contrôler l'angle d'attaque, voir figure 1.6,(a) et (b).
- **Trois câbles** : Le fait d'avoir trois câbles nous permet de contrôler l'aéronef et son angle d'attaque au même temps, c'est plus avantageux mais ça va augmenter l'utilisation de matériels par ce que chaque câble a besoin d'un actionneur.

La figure suivante, 1.6, montre les différentes catégories mentionnées ci-dessus.

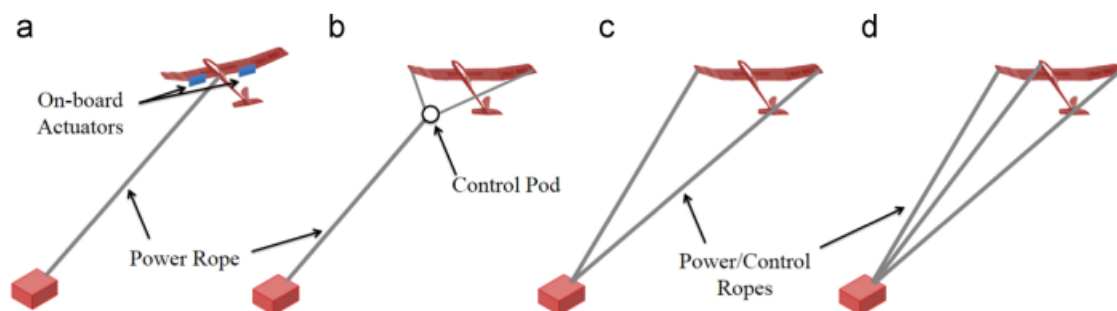


FIGURE 1.6 – Les différents catégories des systèmes AWE selon le nombre de câble, [5]

Ces systèmes génèrent la force de traction au moyen de la force de portance, en anglais **Lift force**.

Revenant maintenant aux systèmes de génération de l'énergie sur l'aéronef, il existe aussi des entreprises et institutions académiques qui ont développé quelques prototypes.

Pour ces systèmes, l'énergie électrique est produite à bord de l'avion pendant son vol et elle est transmise au sol par une corde spéciale qui intègre des câbles électriques. La conversion d'énergie électrique dans les systèmes "fly-gen" est réalisée à l'aide d'une ou plusieurs génératrices ou hélices spécialement conçues.

La figure 1.7 montre les différents types d'aéronefs pour les systèmes "fly-gen"

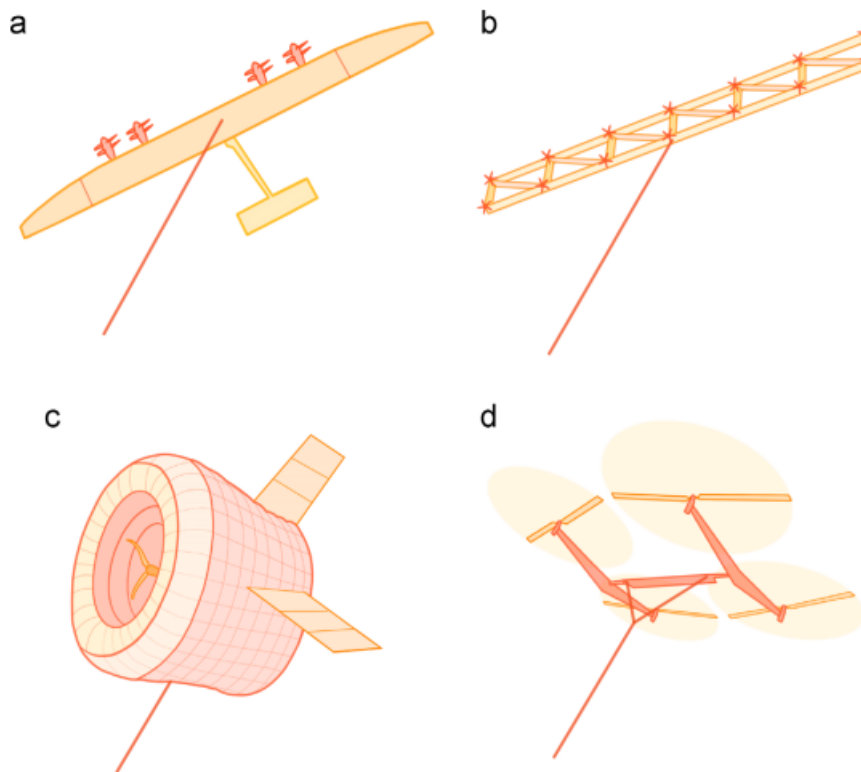


FIGURE 1.7 – (a) Avion avec quatre turbines, conçu par Makani Power. (b) Avion composé d'un cadre d'ailes et de turbines, conçu par Joby Energy. (c) Aérostat de levage toroïdal avec une éolienne au centre, conçu par Altaeros Energies. (d) Quadrirotor à suspension statique en autorotation, conçu par Sky WindPower, [5]

Il existe principalement 3 modes de vol pour ces systèmes selon lesquels on peut les classer :

- **Wings lift** : La portance générée par les ailes, aussi appelée portance aérodynamique, est la force qui s'oppose au poids d'un aéronef et lui permet de voler.

- **Buoyancy and static lift** : des aérostats de forme aérodynamique remplis de gaz plus léger que l'air. Ces aérostats reposent sur le principe de la poussée d'Archimède.
- **Rotor thrust** : Réalisée avec les mêmes turbines utilisées pour la génération de puissance électrique et qui soulèvent au même temps l'aéronef.

### 1.3.1 Principe de fonctionnement général

Dans cette partie nous allons nous concentrer de plus aux systèmes à génération de l'énergie au sol. Ce sont les systèmes les plus courants et la majorité des études scientifiques ont été menées sur ces systèmes. L'énergie électrique pour ces systèmes se produit grâce à des forces mécaniques engendrées par la traction de l'aéronef. Il existe principalement deux phases de génération de l'énergie comme illustré dans la figure ci-dessous.

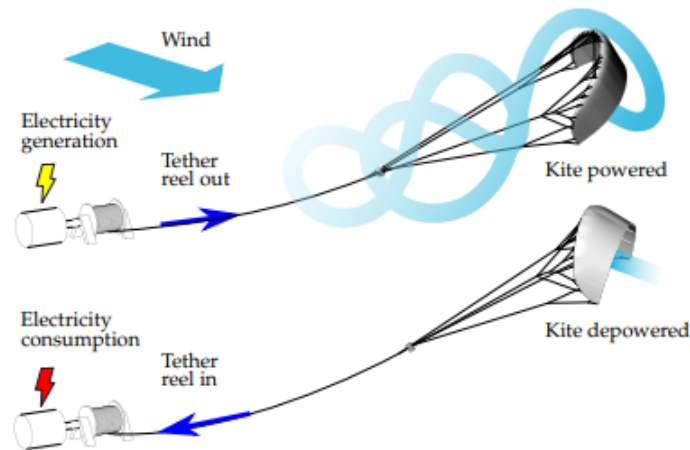


FIGURE 1.8 – Au dessus ; La phase de génération, au dessous ; La phase de reprise, [23]

- **Phase de génération** : Dans un premier temps, l'aéronef décolle puis monte jusqu'à atteindre l'altitude désiré, la phase de génération peut alors commencer, le dispositif volant commence à suivre une trajectoire dans l'air pour maximiser la force de traction grâce à l'augmentation de vitesse, ce qui optimise la production de l'énergie.
- **Phase de reprise** : Dans cette étape, nous allons utiliser une partie de l'énergie générée et la génératrice servira de moteur pour ramener l'aéronef à sa position initiale, puis nous recommencerons la procédure de génération. Pour maximiser l'efficacité, il est essentiel de veiller à ce que l'énergie produite soit bien supérieure à celle utilisée dans cette étape. Cette efficacité est généralement garantie à l'aide de la stratégie de contrôle ; en suivant une trajectoire de reprise qui minimise la force de traction et/ou en modifiant l'angle d'attaque.

Les systèmes à génération de l'énergie sur l'aéronef ne nécessitent pas de phase de reprise et fonctionnent à longueur de câble fixe. Cela évite la phase de consommation d'énergie, mais nécessite généralement de vents plus élevés puisqu'ils possèdent une charge plus importante.

### **1.3.2 Problématique**

Le principal inconvénient des éoliennes aéroportées est qu'ils sont beaucoup plus compliqués à piloter. Certains systèmes sont déjà en cours de commercialisation, cependant les lois de commande mises en place manquent de résistance face à la variété des conditions de fonctionnement. Bien qu'il existe beaucoup de recherches dans le domaine de l'automatisation de ces systèmes, il manque toujours des informations et des données en open source sur les différentes technologies utilisées et l'accessibilité à des données réelles sur ces systèmes. En plus pour pouvoir concevoir une commande robuste et piloter les aéronefs il faut bien valider les équations des modèles dynamiques avec des données qui décrivent le comportement de ces aérogénérateurs. D'où la nécessité de réaliser un dispositif qui soit capable d'acquérir plusieurs informations.

### **1.3.3 Etude de l'existant**

#### **1.3.3.1 Systèmes ground-gen**

Dans cette partie, nous allons effectuer une classification des différents prototypes et produits développés par les entreprises ou les institutions académiques.

Le tableau suivant illustre les différents prototypes des entreprises sans trop de détails sur la partie des capteurs ou des technologies utilisées.

Système AWE	Entreprise	Classe de puissance	Type de force	Actionneurs		Nombre de fils
				Au sol	En l'air	
LEI kite ; cerf-volant avec bord d'attaque gonflable (a)	KiteGen Stem	kW	Force de Portance	x		2
	Windlift			x		3
	Kitenergy			x		2
	Swiss Kite Power 2			x		3
	KitePower			x	x	1
	Swiss Kite Power 1			x	x	1
Foil kite (c)	Skysails Power	Kw-Mw	Force de Portance		x	1
	Enerkite	kW		x		3
Delta kite	Enerkite	kW	Force de portance	x		3
Swept Rigid Wing (e)	Enerkite	kW	Force de portance	x		3
Glider (planeur) (d)	Ampyx Power	kW-Mw	Force de portance		x	1
	e-Kite	kW		x		2
	Kitemill	kW			x	1
Semi-rigid wing	KiteGen stem	Mw	Force de portance	x		2
Parachute	Guangdong Tech	Mw	Trainée		x	2
Aérostat	Omnidea	kW	Effet Magnus		x	2
Glider avec des rotors	TwingTec	kW	Force de portance		x	2

FIGURE 1.9 – Classification des prototypes à génération d'énergie au sol des entreprises, source : [5]

Le tableau suivant montre les différents prototypes développés par des institutions académiques. Quelques informations technologiques ne sont pas incluses dans le tableau, elles sont indiquées dans l'annexe 1.

Prototype	Type de l'aéro-nef	Classe de puissance	Type de force	Actio-neurs	Nomb-re de câble	Capteurs	microco-ntrôleur
Ampyx Power [17]	Planeur	kW /MW	Portance	Embarqués	1	non indiqué	non indi-qué
ABB Cor-poate Research [10]	Planeur	kW	Portance	Embarqués	1	IMU; SBG systems, GPS Antenna; TW1421 (Tallysman), tube pi-tot et un transducteur numérique (MS4525DO)	ARDUINO Mega 2560
Kyushu University (Tarek N. Dief, Mostafa A. Rushdi Amr M. Halawa, Shigeo Yoshida [7], [23])	Voile souple gon-flable	7 kW	Portance	Embarqués : servomo-teur	3	IMU, GPS	ARDUINO
SkySails GmbH, Hamburg for Royal Institute of Technology, Stockholm [24]	Voile souple : ram-air	Non indi-qué	Portance	Embarqués : electric motor and gear box	1	IMU, Ané-momètres, Télémétrie, Tranceiver, Jauges de contrainte	Non indi-qué

TABLE 1.1 – Tableau pour les prototypes Fly-gen des institutions académiques

Prototype	Type de l'aéro-nef	Classe de puissance	Type de force	Actio-neurs	Nomb-re de câble	Capteurs	microco-ntrôleur
Université Carlos III de Madrid [4]	Voile Delta	non in-diqué	Portance	Au sol	2	<b>Embarqués :</b> IMU (Pixhawk 4 flight controller), GPS (2 GNSS RTK MO-DULE) <b>Au sol :</b> Ul-trasonic wind station, télé-metrie, motor hall sensors (mesure de l'angle de rotation), ré-cépteur GNSS, jauges de contrainte, encodeurs, 3 caméras	ARDU-INO (pour la station de sol) et Pixhawk 4 (embar-qué pour l'acqui-sition de données)
Université de Grrenoble – GIPSA Lab [20] (outdoor model)	Voile souple gon-flable (kite-surf)	200 W	Lift	Au sol	1	Torque-meter, axis dual coder (position et trajectoire)	Non indi-qué

TABLE 1.2 – Tableau pour les prototypes ground-gen des institutions académiques

Prototype	Type de l'aéro-nef	Classe de puissance	Type de force	Actio-neurs	Nomb-re de câble	Capteurs	microco-ntrôleur
Small-scale prototype for research in AWE : Fagiano et Trevor Mark [9]	Voile souple gon-flable	W	Portance	Au sol : Steering motor, center line mo-tors	3	<b>Embarqués :</b> IMU <b>Au sol :</b> Line angle sensor, GPS-18x by Garmin, Airspeed sensors, trois jauges de contrainte (s ELPF-T3 E-500L/10F/AMP)	Real time machine (Speed-Goat (pro-grammé à l'aide de xPC Targetr toolbox de Mat-lab)

TABLE 1.3 – Tableau pour les prototypes Fly-gen des institutions académiques

Dans la partie suivante, nous allons examiner les principaux éléments des systèmes de production d'énergie à bord des aéronefs.

### 1.3.3.2 Systèmes fly-gen

Il n'existe pas trop de recherche scientifique pour ce type des systèmes. L'exploration académique est plus limitée que les systèmes ground-gen mais il existe quelques entreprises et institutions académiques qui ont développé des prototypes.

Le tableau suivant montre les différents prototypes qui utilisent des systèmes Fly-gen développé par des entreprises :

Système AWE	Entreprise	Principe de vol	Type	Système de génération de l'énergie
Turbines sur un aéronef attaché à un câble	Makani Power	Force de portance des ailes	Crosswind	6/8 turbines
	Joby Energy			Plusieurs turbines
Quadcopter	Sky Windpower	Poussée des rotors (rotors thrust)	Non-crosswind	4 turbines
Turbine sur un ballon plus léger que l'air (lighter-than-air)	Altaeros Energies	Flottabilité (Buoyancy)	Non-crosswind	1 turbine
Turbine à effet Magnus	Omnidea	Flottabilité (Buoyancy)	Non-crosswind	éolienne flottante (buoyant wind turbine)

FIGURE 1.10 – Les prototypes des systèmes Fly-gen, source : [5]

Le tableau suivant illustre une classification pour un prototype fly-gen développé par une institution de recherche. Il y a des autres détails qui sont indiqués dans l'annexe 1.

Prototype	Type d'aéro-nef	Classe de puissance	Type de force	Actio-neurs	Nomb-re de câble	Capteurs	microco-ntrôleur
University of North Carolina at Charlotte Mitchell Cobb, Nihar Deodhar, Christopher Vermillion [6]	Aérostat	non indiqué	Effet Magnus	Au sol : Trois moteurs DC	3	<b>Embarqués :</b> IMU, GPS	High performance host/target computer

TABLE 1.4 – Tableau pour les prototypes fly-gen des institutions académiques

Dans cette partie on a détaillé les différents prototypes et systèmes existantes pour les deux catégories des éoliennes aéroportées. Dans la prochaine partie, on va présenter notre projet.

### 1.3.4 Présentation du notre projet/Synthèse

Il est vrai qu'il existe des prototypes fabriqués qui sont publiés en open-source comme le prototype robokite [18], et celui développé dans le cadre d'un projet Hackaday intitulé "Flying wing turbine" [15]. De plus, il y a également une plateforme de test appelé AWE-some [2]. Cependant, il peut être indispensable de concevoir de nouveaux systèmes afin de surmonter les contraintes de ces prototypes déjà existants. Le travail qui va être présenté dans les prochains chapitres est un premier pas dans le projet de prototype complet d'éolienne aéroportée qui sera publié en open-source pour effectuer des expérimentations et étudier le comportement de ces systèmes. Ce prototype permettra de valider de nouvelles lois de commande visant à améliorer la fiabilité de ces éoliennes.

L'objectif principal alors de ce stage est de concevoir **un boîtier de capteurs dans le but d'acquérir des données sur le cerf-volant pour pouvoir valider les modèles dynamiques.**

Une fois le boîtier est bien finalisé et testé dans des conditions favorables du vent, nous allons publier le plan de conception et les données acquises en **open-source**, ce qui peut être bénéfique dans le futur pour ceux qui veulent étudier ces systèmes.

## 1.4 Conclusion

Au cours de ce chapitre nous avons présenté le contexte général du projet durant lequel nous avons exploré le domaine des éoliennes aéroportées, les solutions existantes et préciser l'objectif de ce travail, ce qui nous permet de bien situer le projet par rapport à la littérature et au marché actuel.

# Chapitre 2

## Etude préliminaire

### 2.1 Introduction

Dans cette partie nous allons explorer la théorie pour mieux comprendre le système à travers l'étude de son modèle dynamique. Une fois nous avons compris le système et son comportement, nous définirons le cahier des charges du projet dans le but de choisir soigneusement les composants et les technologies les plus adaptés au système.

### 2.2 Etude du modèle de système

Avant d'entamer la partie de conception du boîtier, il est essentiel de commencer par une compréhension initiale de système et son comportement en examinant un modèle approprié.

Parmi les nombreux modèles qui existent dans la littérature, le modèle présenté en [25] a l'avantage d'être assez simple pour donner une bonne compréhension du fonctionnement du système tout en ayant été validé expérimentalement. Le modèle de Wood modélise un prototype avec une voile souple attachée par deux fils à la station de sol.

### 2.2.1 Le système des coordonnées

Selon Tony Alan Wood [25], pour décrire la position et la vitesse d'un cerf-volant, et en se référant à [11], on va utiliser les coordonnées sphériques. Donc on peut exprimer les coordonnées cartésiennes de la position  $\mathbf{p}$  ( $p_x, p_y, p_z$ ) en fonction de l'angle  $\theta$  (latitude), l'angle de l'azimuh  $\phi$  (longitude), et la longueur de câble  $r$ , comme le montre la relation (2.1).

$$\begin{aligned} p_x &= r \sin(\theta) \cos(\phi) \\ p_y &= r \sin(\theta) \sin(\phi) \\ p_z &= r \cos(\theta) \end{aligned} \quad (2.1)$$

La figure 2.1 illustre le système des coordonnées en supposant que la direction du vent est horizontale avec une petite déviation d'angle  $\phi_w$  avec l'axe  $e_x$ . La vitesse du cerf-volant dans les coordonnées sphériques s'écrit comme suit :

$$\mathbf{v} = \begin{bmatrix} v_\theta \\ v_\phi \\ v_r \end{bmatrix} = \begin{bmatrix} r\dot{\theta} \\ r \cos(\theta)\dot{\phi} \\ \dot{r} \end{bmatrix} \quad (2.2)$$

Selon [25], l'angle de cap,  $\gamma$ , est défini comme suit (2.3) [25, eq. 2.2 p 13].

$$\gamma = \arctan 2(v_\phi, v_\theta) = \arctan 2\left(\cos(\theta)\dot{\phi}, \dot{\theta}\right) \quad (2.3)$$

### 2.2.2 Modèle dynamique du cerf-volant

Dans le cadre de notre projet, nous avons choisi d'étudier le modèle de contrôle de Tony Alan Wood. En se référant à sa thèse [25], L'effet de l'entrée de contrôle qui est la différence de longueur des lignes est complexe et influencé de plusieurs conditions et paramètres, alors il propose un modèle simplifié qui peut être ajusté à partir des données réelles. Le modèle se divise en deux parties, le modèle de pilotage, en anglais "steering model", [25, eq. 2.6 p 21], et le modèle de mouvement, en anglais "kinematic model", [25, eq. 2.7a et 2.7b p 22].

Dans le modèle de contrôle de Wood, nous allons considérer l'hypothèse suivante :

**Hypothèse 1** *La vitesse de déroulement de ligne est considérablement plus lente que la composante de vitesse du cerf-volant perpendiculaire au câble.*

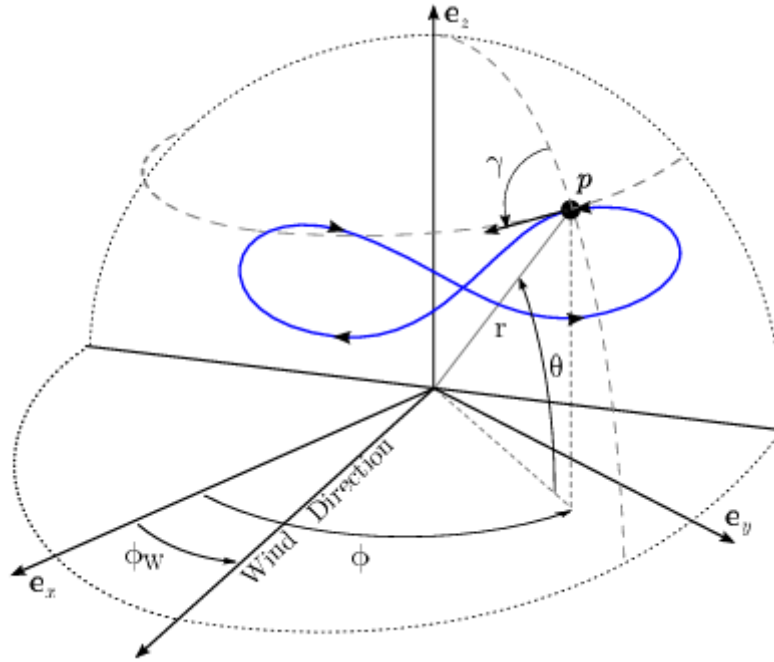


FIGURE 2.1 – L'angle de cap,  $\gamma$ , et la position  $\mathbf{p}$  dans le repère  $(e_x, e_y, e_z)$ , en relation avec l'angle  $\theta$ , l'angle de l'azimuh  $\phi$ , et la longueur de câble,  $r$ ., [25]

### 2.2.2.1 Steering model

Dans ce modèle nous allons s'intéresser à la relation entre l'entrée de contrôle (steering input) qui est principalement la différence de longueur entre les câbles et le taux de changement de l'angle de cap (heading angle).

$$\dot{\gamma}_k = K\delta(t - t_d) \quad (2.4)$$

Avec :

$K$  : gain de direction, en anglais steering gain

$t_d$  : retard de commande, en anglais actuation delay

Dans ce modèle  $K$  et  $d$  sont constant dans l'horizon de contrôle.

### 2.2.2.2 Kinematic model

Ce modèle consiste à donner les vitesses angulaires du cerf-volant en fonction de sa vitesse et de son angle de cap  $\gamma$ .

$$\dot{\theta}(t) = \frac{v_{\theta\phi}(t)}{r} \cos(\gamma(t)) \quad (2.5)$$

$$\dot{\phi}(t) = \frac{v_{\theta\phi}(t)}{r \cos(\theta(t))} \sin(\gamma(t)) \quad (2.6)$$

avec :  $v_{\theta\phi}(t)$  : La norme du vecteur vitesse projeté sur le plan tangent à la sphère au point p, et nous avons :

$$v_{\theta\phi} = \sqrt{v_{\theta}^2 + v_{\phi}^2} \quad (2.7)$$

Après avoir examiné ces modèles, quelles sont les données nécessaires pour étudier le comportement de ces systèmes ? Dans la prochaine section, nous allons aborder le cahier des charges du boîtier.

## 2.3 Cahier des charges du système

### 2.3.1 Les données à acquérir

Avant de concevoir le dispositif d'acquisition de données, il faut définir les grandeurs à mesurer et son cahier des charges qui répond à nos besoins. Il est alors essentiel de définir les types de données comme suit :

- **Vitesses angulaires** : Calculer la vitesse angulaire autour des axes du cerf-volant pour comprendre sa dynamique de rotation, en particulier lors de manœuvres rapide et la réaction aux changements des conditions du vent.
- **Accélérations linéaires** : Mesurer de l'accélération du cerf-volant le long de sa trajectoire de vol, ce qui peut nous aider à estimer la position.
- **Données de position** : Préciser la position du cerf-volant par rapport au station du sol.
- **Vitesse de vent apparent** : Mesurer la vitesse et la direction du vent par rapport au cerf-volant. Comprendre le vent apparent est crucial pour ajuster l'angle d'attaque du cerf-volant et optimiser ses capacités de récupération d'énergie.
- **Tension de ligne** : Acquérir des valeurs de force de ligne qui nous aident à estimer la puissance générée.
- **Données environnementales** : Acquérir des données comme la température qui pourraient aider à comprendre les conditions dans lesquelles le cerf-volant opère.

### 2.3.2 Exigences opérationnelles

- Le boîtier doit communiquer en temps réel avec la station du sol.
- Il est primordial que le système fonctionne sous les conditions naturelles de température et supporte des différents vitesses du vent.

- Le boîtier doit être léger et étanche.
- Il est essentiel que le boîtier ait une alimentation électrique appropriée et embarquée.
- Le boîtier doit supporter les chocs en cas d'un accident avec le sol.

Après avoir spécifié le cahier des charges de notre système, nous avons besoin de certaines informations pour pouvoir choisir nos capteurs par la suite. L'une de ces informations clés est la force exercée sur le cerf-volant pendant son vol. Dans ce qui suit, nous établirons un bilan des forces pour estimer la force totale à laquelle le cerf-volant est soumis.

## 2.4 Calcul de la force de traction générée par le cerf-volant

Il est essentiel d'estimer la force de tension des lignes, ce qui nous permettra de choisir les capteurs de forces adéquats.

La figure 2.2 montre les forces aérodynamiques exercées sur un cerf-volant attaché au sol par un câble.

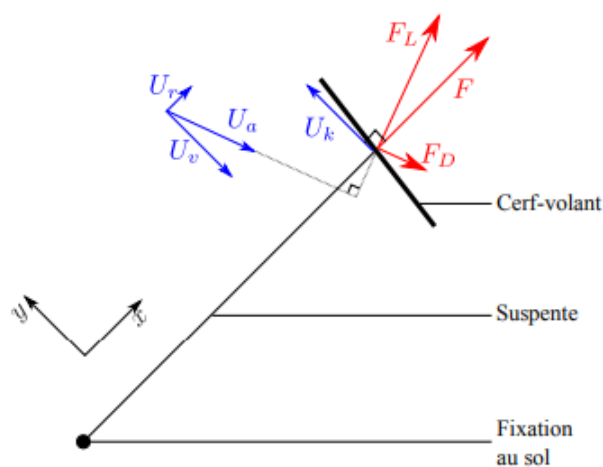


FIGURE 2.2 – Les forces aérodynamiques ressenties par le cerf-volant en vol parallèle au sol [8]

## 2.4. CALCUL DE LA FORCE DE TRACTION GÉNÉRÉE PAR LE CERF-VOLANT<sup>21</sup>

avec :

- $F_L$  : Force de portance
- $F_D$  : Force de traînée
- $F$  : La résultante des deux forces
- $U_k$  : Direction de la vitesse de cerf-volant
- $U_r$  : Direction du vent réel
- $U_a$  : Direction du vent apparent
- $U_v$  : Direction de la vitesse du vent

Le cerf-volant est principalement soumis à la somme des forces  $F_L$ , force de portance, et  $F_D$ , la force de traînée. La somme de ces deux forces s'exprime de la manière suivante :

$$F = \sqrt{F_D^2 + F_L^2} \quad (2.8)$$

Avec

$$F_D = \frac{1}{2}\rho C_D A_{\text{kite}} V_a^2 \quad (2.9)$$

et

$$F_L = \frac{1}{2}\rho C_L A_{\text{kite}} V_a^2 \quad (2.10)$$

On a alors :

$$F = \frac{1}{2}\rho C_N A_{\text{kite}} V_a^2 \quad (2.11)$$

avec :

$$C_N = \sqrt{C_D^2 + C_L^2} \quad (2.12)$$

- $C_L$  : Coefficient de portance de cerf-volant
- $C_D$  : Coefficient de traînée de cerf-volant
- $C_N$  : Coefficient de force totale de cerf-volant
- $A_{\text{kite}}$  : Aire de cerf-volant en  $m^2$
- $\rho$  : Densité de l'air en  $kg/m^3$
- $V_a$  : Vitesse du vent apparent

Pour les cerf-volants et vu qu'ils sont conçus pour maximiser la force de portance au cours d'un vol, le coefficient de traînée est négligeable par rapport au coefficient de portance,  $C_D \ll C_L$  [8], d'où  $C_N \approx C_L$ .

Nous allons concevoir un prototype avec un cerf-volant qui pèse 1,2 kg et possède une taille de  $2.5 \text{ m}^2$ . Notre cerf-volant est une voile de kitesurf de type UNO, son coefficient  $C_L$  vaut généralement 1,2. En outre la densité volumique de l'air  $\rho = 1,2 \text{ kg/m}^3$  pour une altitude de 300m et une température de  $15 \text{ }^\circ\text{C}$ .

Donc pour calculer la force totale appliquée sur le cerf-volant, il reste qu'à calculer approximativement la vitesse apparente du vent ressentie par le cerf-volant  $V_a$ . Pour calculer cette vitesse, il est essentiel de connaître la vitesse de déplacement du cerf-volant dans l'air et l'angle  $\alpha$  entre la direction de vent et la direction de vol de notre cerf-volant. Selon [8], la norme du vecteur vent apparent  $V_a$  est donnée par la relation suivante :

$$V_a = \sqrt{V_k^2 + V_w^2 + V_k \times V_w \times \cos \alpha} \quad (2.13)$$

Avec  $V_k$  est la norme du vent vitesse et  $V_w$  est la norme du vecteur vent.

Si nous supposons que le cerf-volant vole en mouvement transversal par rapport au vent, en anglais en "crosswind motion", cela signifie que sa direction de vol est perpendiculaire à la direction de vent, dans ce cas  $\alpha = \frac{\pi}{2}$ , d'où  $V_a = \sqrt{V_k^2 + V_w^2}$ .

Dans notre cas la vitesse du vent ne dépasse pas généralement 10 m/s. Il reste à estimer la vitesse à laquelle le cerf-volant vole,  $V_k$ . Une approximation de la vitesse  $V_k$  peut être trouvée en fonction des coefficients aérodynamiques  $C_D$  et  $C_L$ . Lors d'une manoeuvre transversale au vent, le cerf-volant vole avec une vitesse plus grande que la vitesse de vent, et on a la relation suivante :  $\frac{C_L}{C_D} = \frac{V_k}{V_w}$ .

Or le quotient entre les coefficients aérodynamiques est approximativement entre 1.5 et 3, si on considère les conditions environnementales et réelles sous lesquelles le cerf-volant est soumise, on peut alors estimer que la vitesse du cerf-volant est supérieure à celle du vent d'environ 1,5 à 3 fois.

Après avoir effectué les calculs numériques, nous avons trouvé que la force totale à laquelle le cerf-volant est soumis, sous les conditions mentionnées ci-dessus, est comprise entre deux valeurs , 600 N et 1800 N pour 10 m/s de vent et entre 390 N et 1160 N pour 8 m/s de vent. Pour obtenir une bonne précision de mesure durant la majorité des tests et sachant que les jauges de contrainte tolèrent une force de  $2 \times F_{nom}$ , on choisira des jauges de contrainte de 500 N par câble (notre prototype possède deux câbles donc au totale les deux peuvent supporter 1000 N).

## 2.5 Relation entre les données acquises par le cerf-volant et les paramètres du modèle de Wood

Pour valider les équations du modèle de Wood (2.6) et (2.5), il faut avoir une relation entre les paramètres du modèle et les données que nous pouvons acquérir à l'aide du boîtier de capteurs.

### Les paramètres du modèle :

- L'angle de cap,  $\gamma$  qui est l'angle entre la vitesse à laquelle le cerf-volant se déplace et la direction du nord de la sphère.
- La position du cerf-volant, dans les coordonnées sphériques, représentée par l'angle,  $\theta$  et l'angle de l'azimuh,  $\phi$ .

### Les données acquises par le boîtier de capteur

- Les accélérations angulaires
- Les accélérations linéaires
- L'orientation

Certainement, les accélérations acquises par le boîtier ne sont pas les mêmes représentées dans le repère  $(e_x, e_y, e_z)$ , voir figure 2.1. Donc il faut trouver une relation pour lier ces accélérations avec ce repère. Selon [11], nous pouvons définir un système de coordonnées non-inertiel, local et centré à la position du cerf-volant dans l'air  $L = (L_N, L_E, L_D)$ , comme le montre la figure 2.3.

Les accélérations obtenues à l'aide du boîtier seront représentées dans ce référentiel local, qui est défini comme suit :

- L'axe  $L_N$ , ou nord local, est tangent à la sphère et pointe vers son zénith.
- L'axe  $L_D$ , local down en anglais, pointe vers le centre de la sphère (c-à-d. la station du sol), il est donc perpendiculaire au plan tangent à la sphère à l'emplacement de cerf-volant.
- L'axe  $L_E$ , ou l'est local, complète la base et forme le plan tangent avec  $L_N$ .

On note  $G = (X, Y, Z) = (e_x, e_y, e_z)$ , le repère de la station au sol. Un vecteur dans le repère L peut être exprimé dans le repère G par la matrice suivante, [11].

Soit le vecteur  $p_L$  appartient à L :

$$\mathbf{p}_L = \begin{bmatrix} p_N \\ p_E \\ p_D \end{bmatrix} \quad (2.14)$$

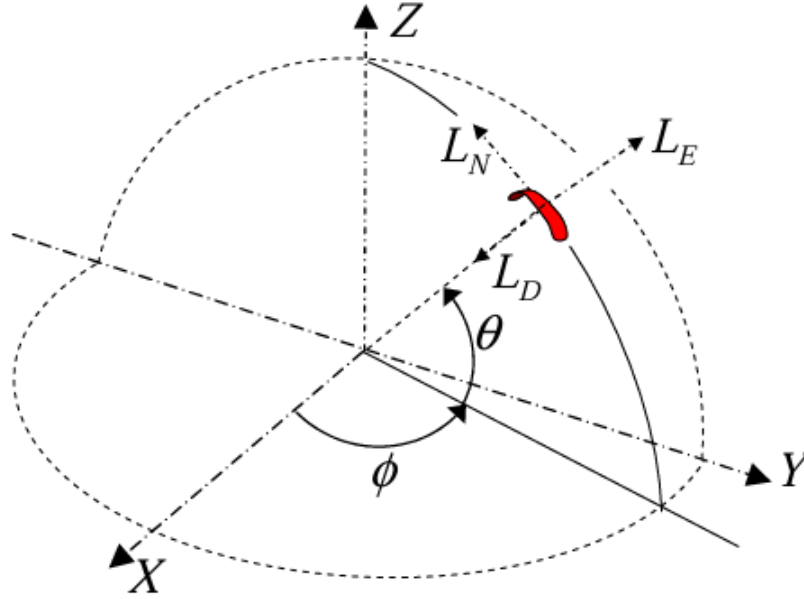


FIGURE 2.3 – le repère  $(X, Y, Z)$  qui est le même  $(e_x, e_y, e_z)$  dans la figure 2.3, l'angle,  $\theta$ , l'angle de l'azimuh,  $\phi$ , et le repère  $(L_N, L_E, L_D)$ , , [11]

Le vecteur  $p_G$  représenté dans le repère G s'écrit comme suit :

$$\mathbf{p}_G = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (2.15)$$

avec :

$$p_G = R p_L \quad (2.16)$$

Où R est défini par :

$$R = \begin{pmatrix} -\cos(\phi(t)) \sin(\theta(t)) & -\sin(\phi(t)) & -\cos(\phi(t)) \cos(\theta(t)) \\ -\sin(\phi(t)) \sin(\theta(t)) & \cos(\theta(t)) & -\sin(\phi(t)) \cos(\theta(t)) \\ \cos(\theta(t)) & 0 & -\sin(\theta(t)) \end{pmatrix} \quad (2.17)$$

Les accélérations mesurées par l'IMU sont données dans un autre repère appelé  $C = (V, S, L_D)$ , voir figure 2.4, qui est orienté avec le cerf-volant et est donc obtenu par une rotation du repère L de l'angle  $\gamma$  autour de l'axe  $L_D$ . Ainsi :

- V est l'axe dans la même direction que la vitesse de cerf volant. Il est obtenu par une rotation de l'angle  $\gamma$ .
- L'axe S est l'axe  $L_D$  obtenu par une rotation de l'angle  $\gamma$ .
- $L_D$  est le même axe que celui du repère L.

## 2.5. RELATION ENTRE LES DONNÉES ACQUISES PAR LE CERF-VOLANT ET LES PARAMÈTRES

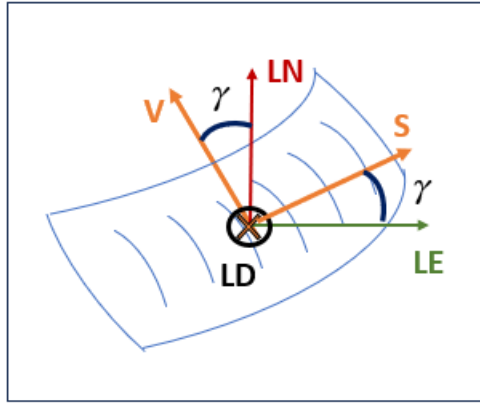


FIGURE 2.4 – Vue du plan tangent au sphère et perpendiculaire au rayon  $r$ , en représentant les repères C et L

D'une part, nous remarquons que **la vitesse angulaire autour de l'axe  $L_D$ , mesurée par notre IMU, est la même que la vitesse angulaire  $\dot{\gamma}$** .

D'autre part, nous pouvons avoir les angles des coordonnées sphériques  $\theta$  et  $\phi$  à partir des accélérations mesurées par l'IMU, mais il faut établir une relation entre le repère C du cerf-volant et le repère local L.

Soit le vecteur  $p_C$  représenté dans le repère C s'écrit comme suit :

$$p_C = \begin{bmatrix} p_V \\ p_S \\ p_D \end{bmatrix} \quad (2.18)$$

D'après la figure 2.4, Le vecteur  $p_L$ , représenté dans le repère L, s'écrit comme suit ;

$$p_L = R_1 p_C \quad (2.19)$$

Avec  $R_1$ , la matrice de rotation de l'angle  $\gamma$  autour de l'axe  $L_D$ , définie par :

$$R_1 = \begin{pmatrix} \cos(\gamma) & \sin(\gamma) & 0 \\ -\sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (2.20)$$

Donc pour avoir les angles  $\theta$  et  $\phi$ , nous allons suivre les étapes suivantes :

- Intégrer deux fois les accélérations pour avoir la position du cerf volant dans le repère C.
- Transformer les coordonnées de la position du repère C au repère L à l'aide de la matrice  $R_1$  (2.20).

- Puis transformer ces coordonnées du repère local L au repère G à l'aide de la matrice R (2.17).

- Nous aurons donc la position du cerf-volant dans le repère G, en coordonnées cartésiennes, et nous pouvons par la suite avoir les coordonnées sphériques à travers les relations suivantes :

$$\theta = \arcsin \frac{z}{r} \quad (2.21)$$

$$\phi = \arctan 2(y, x) \quad (2.22)$$

Revenant à les équations du modèle de Wood (2.5) et (2.6), il nous manque encore la vitesse  $v_{\theta\phi}$ ,

En regardant la relation (2.7), nous pouvons avoir la vitesse  $v_{\theta\phi}$  en calculant les vitesses  $v_\theta$  et  $v_\phi$ . D'après les deux figures 2.3 et 2.4, nous constatons que la vitesse  $v_\theta$  est la vitesse linéaire selon l'axe  $L_N$  du repère C, et la vitesse  $v_\phi$  est la vitesse linéaire selon l'axe  $L_E$  du même repère. On obtient alors :

$$v_\theta = \dot{p}_N \quad (2.23)$$

$$v_\phi = \dot{p}_E \quad (2.24)$$

D'après (2.20) et (2.16), nous avons :

$$p_N = p_V \cos(\gamma) + p_S \sin(\gamma) \quad (2.25)$$

et :

$$p_E = -p_V \sin(\gamma) + p_S \cos(\gamma) \quad (2.26)$$

En développant les calculs, on aura :

$$v_\theta = \dot{p}_N = \dot{p}_V \cos(\gamma) - p_V \dot{\gamma} \sin(\gamma) + \dot{p}_S \sin(\gamma) + p_S \dot{\gamma} \cos(\gamma) \quad (2.27)$$

et :

$$v_\phi = \dot{p}_E = -\dot{p}_V \sin(\gamma) - p_V \dot{\gamma} \cos(\gamma) + \dot{p}_S \cos(\gamma) - p_S \dot{\gamma} \sin(\gamma) \quad (2.28)$$

Au cours de cette section, nous avons essayé d'établir une relation entre les données mesurées par le boîtier et les paramètres du modèle de Wood étudié au début du chapitre.

## 2.6 Conclusion

Dans ce chapitre, nous avons effectué une étude préliminaire, débutant par une étude d'un modèle dynamique du système dans le but de comprendre son comportement et son

fonctionnement durant le vol. Ensuite, nous avons détailler le cahier des charges dans le but de se préparer pour choisir technologiquement les capteurs appropriés à intégrer dans le boîtier. Ce choix sera abordé dans le prochain chapitre. Et finalement, nous avons essayé de définir une relation entre les aspects expérimentaux et théoriques.

# Chapitre 3

## Réalisation et tests

### 3.1 Introduction

Après avoir étudié notre système au niveau de son modèle et spécifié nos besoins en établissant le cahier des charges du boîtier de capteurs, nous allons aborder dans ce chapitre la partie réalisation du projet en détaillant l'implémentation matérielle et logicielle.

### 3.2 Description de la solution

#### 3.2.1 Architecture générale du système

La figure 3.1 montre la structure générale de notre système. Nous allons détailler les types de capteurs à choisir pour répondre au cahier des charges.

**IMU :** Il est composé d'un gyroscope pour mesurer les vitesses angulaires, d'un accéléromètre pour les accélérations linéaires, d'un magnétomètre pour l'orientation par rapport au champ magnétique et parfois d'un baromètre pour la mesure de la pression atmosphérique.

**GPS :** Il nous donne une idée de la position du véhicule, plus précisément de sa longitude et de sa latitude.

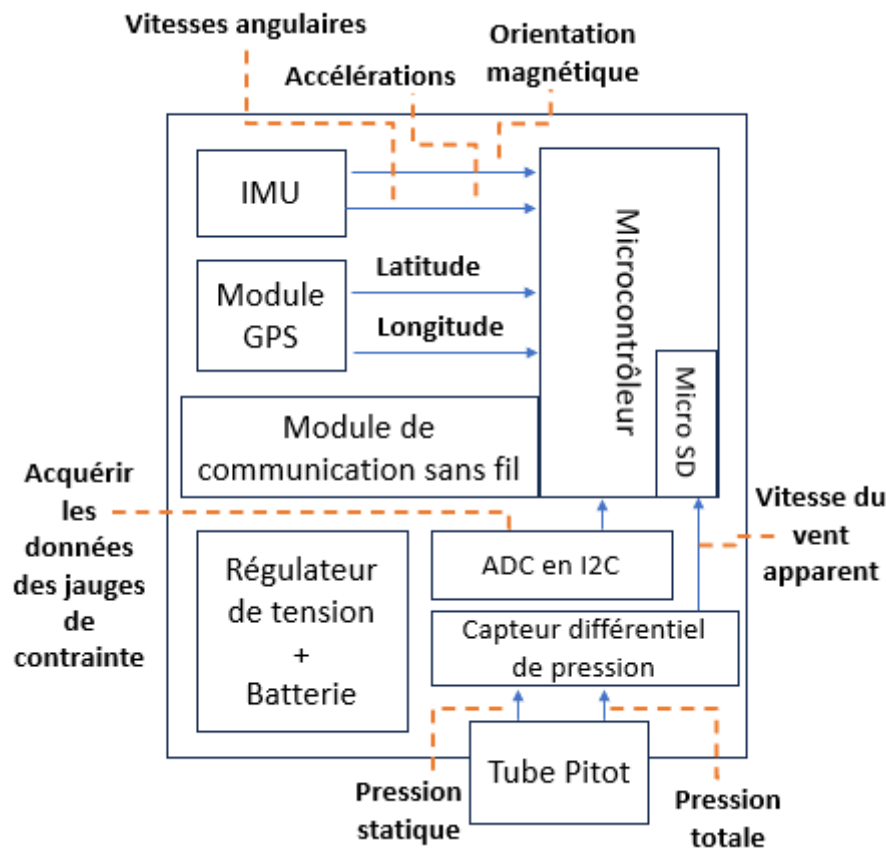


FIGURE 3.1 – L'architecture du boîtier de capteurs

**Tube pitot et capteur différentiel de pression (Airspeed sensor) :** Le tube de Pitot capte à la fois la pression totale et statique, puis le capteur de pression différentielle mesure la différence de pression entre les deux orifices. La pression dynamique, dérivée de la différence de pression, permet le calcul de la vitesse apparente du vent selon le principe de Bernoulli.

**Convertisseur ADC en I2C :** pour mesurer les forces de ligne, nous avons besoin de cellules de charge, ces cellules de charge donnent une valeur de sortie analogique, donc nous avons besoin de ce convertisseur pour être en mesure de lire les valeurs des forces de ligne avec le microcontrôleur.

**Module de communication sans fil :** Pour suivre le comportement de notre cerf-volant en vol, nous avons besoin d'un système de communication sans fil. Ce système envoie des données en temps réel depuis les capteurs à notre station au sol au moyen d'ondes électromagnétiques.

**Alimentation ; Régulateur de tension et Batterie :** Une alimentation embarquée est nécessaire pour faire fonctionner les capteurs. Cette alimentation est composée essentiellement d'une Batterie qui doit être la plus légère possible et d'un régulateur de tension pour ajuster la tension avec nos besoins.

**Microcontrôleur :** Un microcontrôleur est nécessaire pour lire les données, les traiter et établir la communication sans fil avec la station du sol.

### 3.2.2 Flux de données

Les accélérations linéaires, les vitesses angulaires, la position du véhicule, la vitesse apparente du vent et les forces sur la ligne seront tous inclus dans le flux final des données. Ces données peuvent être enregistrées sur une carte microSD et transférées à l'aide d'un module de communication sans fil.

Au cours de la partie "étude de l'existant", abordée dans le chapitre 1, nous avons exploré les différentes architectures et les choix technologiques possibles pour réaliser le boîtier de capteurs. La plupart des prototypes dans la littérature possède une structure principale similaire à celle mentionnée précédemment, qui contient essentiellement l'IMU, le GPS et le capteur de vitesse du vent, sauf que pour certains, ils ajoutent d'autres types de capteurs comme le capteur de mesure de l'angle d'élévation, ou des capteurs (caméras) pour suivre le dispositif volant dans l'air, dans le but de fournir plus de précision au niveau de la mesure de la position. Pour le choix technologique, il existe des différentes approches et nous allons détailler notre choix technologique dans la section suivante. La figure montre les liaisons entre les différents capteurs choisis.

## 3.3 Choix du matériel

Dans un premier temps, nous allons détailler le choix technologique de matériel nécessaire pour répondre au cahier des charges convenablement. Les composants matériels d'un système d'acquisition de données pour les éoliennes aéroportées (AWES) sont conçus pour être légers et capables de fonctionner dans des conditions environnementales exigeantes. Afin de choisir la meilleure technologie, nous avons recueilli des informations à partir des prototypes publiés et réalisés. Nous avons également envisagé d'utiliser des technologies liées à la robotique, en particulier pour les drones, car elles nécessitent un poids léger, une résilience aux conditions de vol réelles et une efficacité énergétique.

### 3.3.1 Microcontrôleur

Pour le microcontrôleur, nous avons décidé de travailler avec **”Pixhawk 6C flight controller”**, qui est désigné principalement pour les applications des drones et les véhicules sans pilote. Il est connu principalement pour sa compatibilité open-source.

Il est basé sur un microcontrôleur STM32H7 et équipé d’un IMU intégré et un emplacement pour une carte microSD, ce qui nous dispense d’acheter de matériel supplémentaire.

Pixhawk 6C fournit plusieurs ports d’entrée/sortie pour connecter des périphériques tels que le GPS, les modules de télémétrie, les ESC (Electronic Speed Controller pour les moteurs) et les caméras. Ce contrôleur de vol prend en charge divers protocoles de communication tels que UART, I2C et CAN, ce qui le rend très adaptable à différentes configurations matérielles.



FIGURE 3.2 – Pixhawk 6C

### 3.3.2 Capteurs

#### 3.3.2.1 IMU

L’IMU intégré a principalement deux accéléromètres/gyroscopes différents, un magnétomètre et un baromètre.

Deux accéléromètre/gyroscope à 3 axes :

La sensibilité dépend de l’intervalle de mesure pour les accéléromètres et les gyroscopes, voir annexe 2.

**Accel/Gyro : ICM-42688-P de TDK InvenSense**

**Accel/Gyro : BMI055 de Bosch Sensortec**

Magnétomètre : IST8310 de AKM

**Marge de valeurs :** de  $\pm 1600$  uT à  $\pm 2500$  uT

**Sensibilité :** 3.3 LSB/uT

Baromètre : MS5611 de TE Connectivity

**Marge de valeurs :** 10 mbar to 1200 mbar

**Précision :**  $\pm 1.5$  mbar

### 3.3.2.2 GPS

Nous avons choisi d'utiliser le GPS M8N de Holybro qui est un module GNSS à haute performance conçu pour le positionnement et la navigation précis. Il est basé sur le chipset u-blox M8, qui prend en charge plusieurs systèmes satellites tels que GPS, GLONASS, Galileo et BeiDou.

**altitude maximale mesurée :** 50 km

**Précision :** 2 m



FIGURE 3.3 – GPS M8N de Holybro

### 3.3.2.3 Capteur de vitesse de vent de Holybro

Ce capteur possède le **MS4525DO**, un capteur de pression MEMS (Micro-Electro-Mechanical Systems) qui fournit des lectures précises de la pression différentielle à l'aide du tube pitot, ensuite en utilisant la loi de Bernoulli nous pouvons calculer la vitesse du vent apparent. Il communique en plus via I2C, fournissant une sortie numérique pour

une intégration facile avec les contrôleurs de vol comme Pixhawk 6C. Il est composé principalement par, voir figure 3.4 :

- Tube pitot PT40.
- Tube en caoutchouc 40cm.
- Le capteur différentiel de pression numérique.

**Marge de valeurs mesurées :** de 1 m/s à 100 m/s



FIGURE 3.4 – capteur de vitesse de vent de Holybro

#### 3.3.2.4 Jauges de contrainte

D'après l'étude de dimensionnement que nous avons fait, nous allons choisir deux jauges de contrainte, chacun ayant une capacité de force de 500 N. Nous devons, en plus, choisir en priorité des capteurs qui non seulement répondent aux exigences de force mais qui ont aussi le poids le plus faible possible.

C'est pourquoi nous allons travailler avec des jauges de contrainte **WIKA f2812**, voir figure 3.5, qui pèsent 100 g, voir le datasheet en annexe 2.



FIGURE 3.5 – Jauge de contrainte WIKA F2812

### 3.3.2.5 Convertisseur ADC en I2C

Pour lire les valeurs analogiques de sortie des jauges de contraintes avec notre contrôleur Pixhawk 6C, nous avons besoin certainement d'un convertisseur ADC en I2C.

Nous avons constaté que le convertisseur **ADS1115** possède déjà une bibliothèque compatible qui peut être intégrée avec l'environnement logiciel directement. La figure 3.6 illustre le convertisseur ADS1115.

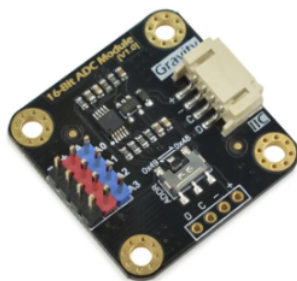


FIGURE 3.6 – ADS1115, convertisseur ADC en I2C

### 3.3.2.6 Télémétrie

Le radio télémétrie Sik de Holybro (Sik Telemetry Radio V3), comme le montre la figure 3.7, est un module de communication sans fil conçu pour la transmission de données entre un drone (ou autre véhicule télécommandé) et une station au sol. Il utilise le firmware SiK, un firmware open-source créé spécialement pour les radios de télémétrie, et fonctionne sur les bandes de fréquences 433 MHz ou 915 MHz. Ce dispositif fournit une communication longue distance entre la véhicule et la station au sol jusqu'à 400 m de distance.



FIGURE 3.7 – Sik Telemetry Radio V3 de Holybro

### 3.3.3 Alimentation

La majorité des capteurs sont alimentés par le Pixhawk 6C, qui fonctionne sur une entrée 5V pour l'alimentation. Nous avons déjà un module d'alimentation qui est le module PM02 de Holybro mais nous n'avons pas besoin d'une alimentation de puissance, donc nous devons concevoir un circuit simple d'alimentation pour éviter les surdimensionnements. Pour résoudre ce problème, nous avons décidé de concevoir un circuit qui peut fournir une tension régulée de 5V, provenant d'une batterie Li-Ion de 7,4 V. La solution est alors d'utiliser un régulateur L7805CV, comme le montre la figure 3.8, pour avoir une alimentation 5 V propre.

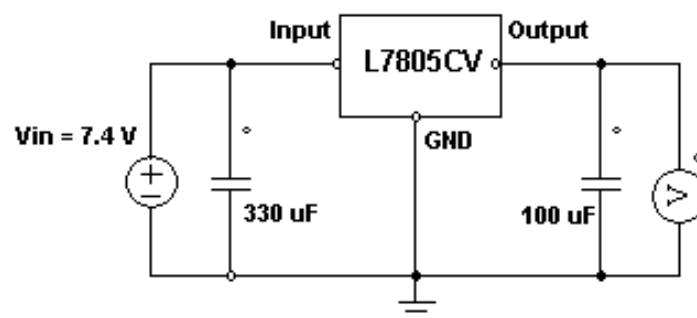


FIGURE 3.8 – Régulateur de tension 5V

## 3.4 Implémentation logicielle

### 3.4.1 Environnement de travail

Dans cette section nous allons décrire les différents logiciels utilisés pour finaliser la réalisation du boîtier.

### 3.4.1.1 L'autopilote PX4

L'autopilote PX4 est une suite logicielle de contrôle de vol en open-source conçue pour les systèmes autonomes comme les véhicules aériens sans pilote (UAV).

Il fonctionne sur le système d'exploitation en temps réel NuttX qui fournit les services essentiels dont PX4 a besoin pour gérer ses tâches, gérer la communication et interagir avec le matériel.

Il est aussi compatible avec plusieurs logiciels de contrôle au sol, tel que QGroundControl, ce qui rend le suivi en temps réel et l'analyse des données possibles.

### 3.4.1.2 QGroundControl

QGroundControl est un logiciel de station de contrôle au sol (GCS) qui est open-source et utilisé pour l'exploitation et la surveillance des véhicules aériens sans pilote (UAV) et d'autres systèmes télécommandés. Il offre des données de télémétrie en direct, y compris l'état du véhicule, les niveaux de batterie, les données GPS et les lectures des capteurs, et prend en charge le protocole MAVLink pour la communication entre la station au sol et le contrôleur de vol. Le protocole MAVLink est un protocole de messagerie très léger qui a été conçu pour l'écosystème des drones.

## 3.4.2 Programmation et configuration

### 3.4.2.1 IMU

Pour l'IMU, nous n'avons pas besoin d'une phase de programmation séparée car l'autopilote PX4 traite directement les données IMU à l'aide de QGroundControl. Ces données sont facilement accessibles et peuvent être consultées en temps réel grâce à l'inspecteur MAVLink, qui fournit des informations détaillées sur les sorties du capteur.

Cependant, une calibration des capteurs de l'IMU est essentiel avant de commencer l'acquisition de données. Pour que le système fonctionne correctement, ce processus garantit des lectures de capteur précises et fiables. La figure 3.9 montre un exemple de calibration à travers QGroundControl pour l'accéléromètre. Le processus consiste alors à positionner le contrôleur de vol en suivant les images présentées sur l'écran, voir figure 3.10.

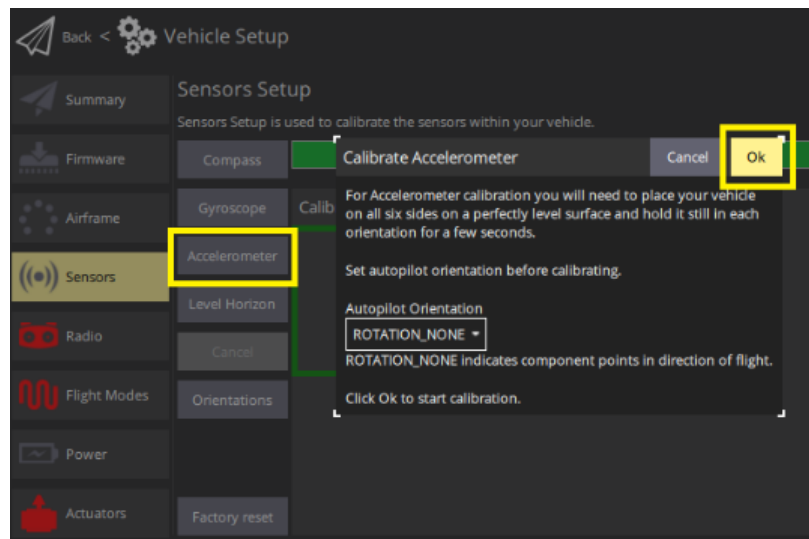


FIGURE 3.9 – calibration de l'accéléromètre à travers QGroundControl, [21]

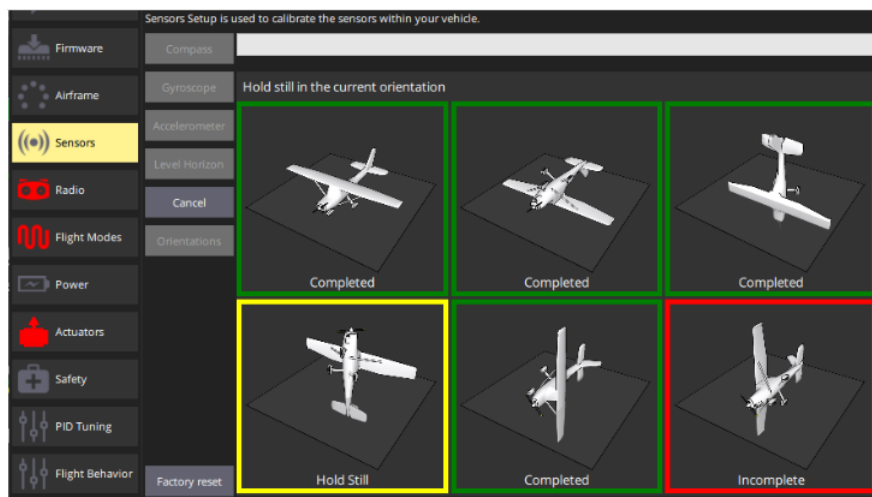


FIGURE 3.10 – processus de calibration de l'accéléromètre, [21]

Une procédure similaire est requise pour calibrer les autres capteurs comme le magnétomètre et le gyroscope.

### 3.4.2.2 GPS

La configuration du module GPS M8N est simple ; il suffit de le brancher sur le port GPS1 du Pixhawk. Certainement nous devons réaliser une calibration pour que le capteur fonctionne correctement. D'une manière similaire à la calibration des accéléromètres, le GPS sera calibré à l'aide de QGroundControl en calibrant le "compass", voir la figure 3.11.

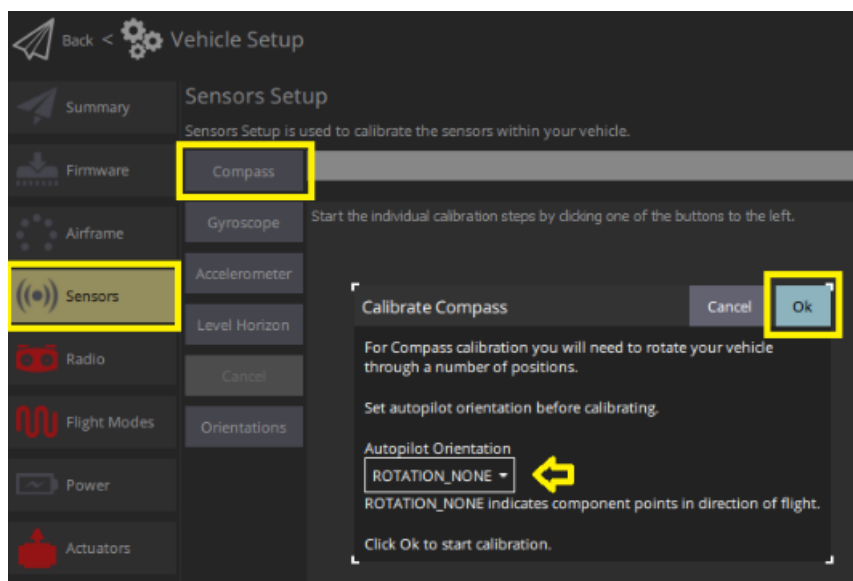


FIGURE 3.11 – calibration de GPS, [22]

Après avoir connecter et calibrer le GPS, il doit fonctionner comme prévu. Mais, nous avons eu un problème ; le GPS n'a pas détecté les signaux provenant des satellites. Pour vérifier l'état de GPS, nous utilisons la commande **"gps status"** dans le **"MAVLink console"** de QGroundControl, et il a montré qu'aucun satellite n'était trouvé, voir figure 3.12. Une fois à l'extérieur, le GPS a réussi à capter des signaux et il a fonctionnait comme prévu.

### 3.4.2.3 Convertisseur ADC en I2C

Nous avons décidé d'utiliser le convertisseur ADS1115 car, comme indiqué dans la section du choix de matériel, il dispose déjà d'une bibliothèque qui s'interface directement avec PX4. Pour faire fonctionner l'ADS1115, nous allons suivre les étapes suivantes.

- **Clonage du référentiel de PX4** : Le dépôt ou le référentiel d'un "firmware" est l'endroit où les modifications apportées aux fichiers sont stockées. Il contient le code source, la configuration des fichiers, la documentation, les scripts de compilation, les bibliothèques

```

nsh> gps status
INFO [gps] Main GPS
INFO [gps] protocol: UBX
INFO [gps] status: OK, port: /dev/ttyS0, baudrate: 115200
INFO [gps] sat info: disabled
INFO [gps] rate reading: 0 B/s
INFO [gps] rate position: 5.00 Hz
INFO [gps] rate velocity: 5.00 Hz
INFO [gps] rate publication: 5.00 Hz
INFO [gps] rate RTCM injection: 0.00 Hz
sensor_gps
timestamp: 170507492 (0.122855 seconds ago)
timestamp_sample: 0
latitude_deg: 0.000000
longitude_deg: 0.000000
altitude_msl_m: -17.000000
altitude_ellipsoid_m: 0.000000
time_utc_usec: 0
device_id: 11010053 (Type: 0xA8, SERIAL:0 (0x00))
s_variance_m_s: 1000.00006
c_variance_rad: 3.14159
epf: 4294967.50000
epv: 3750328.25000
hdop: 99.99000
vdop: 99.99000
noise_per_ms: 95
jamming_indicator: 12
vel_m_s: 0.00000
vel_n_m_s: 0.00000
vel_e_m_s: 0.00000
vel_d_m_s: 0.00000
cog_rad: 0.00000
timestamp_time_relative: 0
heading: nan
heading_offset: 0.00000
heading_accuracy: 0.00000
rtcm_injection_rate: 0.00000
automatic_gain_control: 1404
fix_type: 0
jamming_state: 0
spoofing_state: 1
vel_ned_valid: False
satellites_used: 0 ←
selected_rtc_instance: 0
rtcm_crc_failed: False
rtcm_msg_used: 0

```

FIGURE 3.12 – État de GPS dans MAVLink console

de modules et les fichiers de simulation et de test. La première étape consiste à cloner le dépôt de PX4, qui est essentiel pour effectuer et suivre les modifications apportées au code source.

Pour cloner le dépôt, nous devons installer Git, (voir l'annexe 3 pour les détails). Après cette étape, le référentiel PX4 sera placé dans le fichier "Downloads".

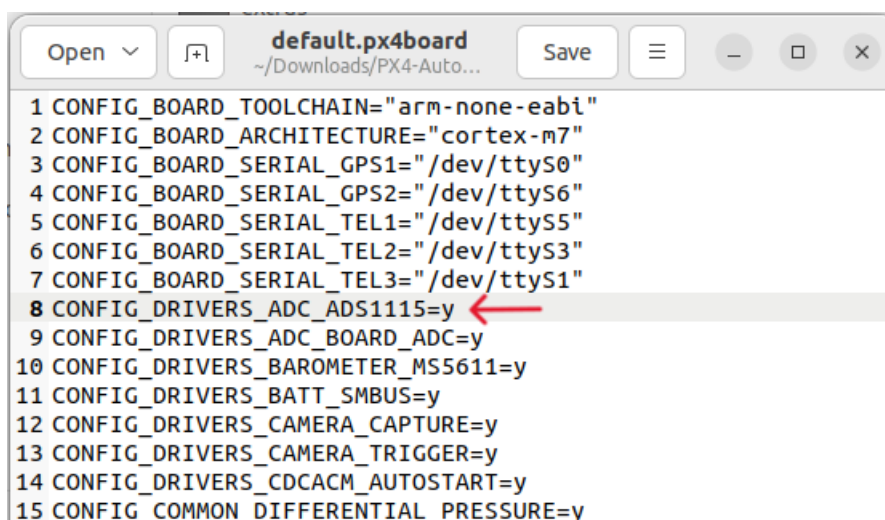
- **Modification du code source** : Ensuite, nous devons vérifier si le pilote de l'ADS1115 existe dans le fichier "drivers" qui a le chemin de fichier suivant, "PX4-autopilot/src/drivers/adc/ads1115".

Si nous n'avons pas trouvé le pilote, nous pouvons le télécharger à partir de Github [1]. Ensuite nous l'intégrons dans le référentiel de PX4.

Une fois le pilote disponible dans notre référentiel PX4, nous modifierons la ligne suivante pour activer le pilote de ADS1115, comme indiqué sur la figure 3.13 :

**CONFLDRIVERS\_ADC\_ADS1115=y**

Nous pouvons accéder à cette ligne en suivant ce chemin :



```

1 CONFIG_BOARD_TOOLCHAIN="arm-none-eabi"
2 CONFIG_BOARD_ARCHITECTURE="cortex-m7"
3 CONFIG_BOARD_SERIAL_GPS1="/dev/ttyS0"
4 CONFIG_BOARD_SERIAL_GPS2="/dev/ttyS6"
5 CONFIG_BOARD_SERIAL_TEL1="/dev/ttyS5"
6 CONFIG_BOARD_SERIAL_TEL2="/dev/ttyS3"
7 CONFIG_BOARD_SERIAL_TEL3="/dev/ttyS1"
8 CONFIG_DRIVERS_ADC_ADS1115=y
9 CONFIG_DRIVERS_ADC_BOARD_ADC=y
10 CONFIG_DRIVERS_BAROMETER_MS5611=y
11 CONFIG_DRIVERS_BATT_SMBUS=y
12 CONFIG_DRIVERS_CAMERA_CAPTURE=y
13 CONFIG_DRIVERS_CAMERA_TRIGGER=y
14 CONFIG_DRIVERS_CDCACM_AUTOSTART=y
15 CONFIG_COMMON_DIFFERENTIAL_PRESSURE=y

```

FIGURE 3.13 – Le fichier default.px4board

**"Downloads/PX4-Autopilot/boards/px4/fmu-v6c/default.px4board"**

Une fois modifié, nous enregistrons les changements et nous passons à l'étape suivante.

- **Compilation** : Préalablement à la compilation du firmware PX4, nous devons nous assurer que toutes les dépendances requises sont installées.

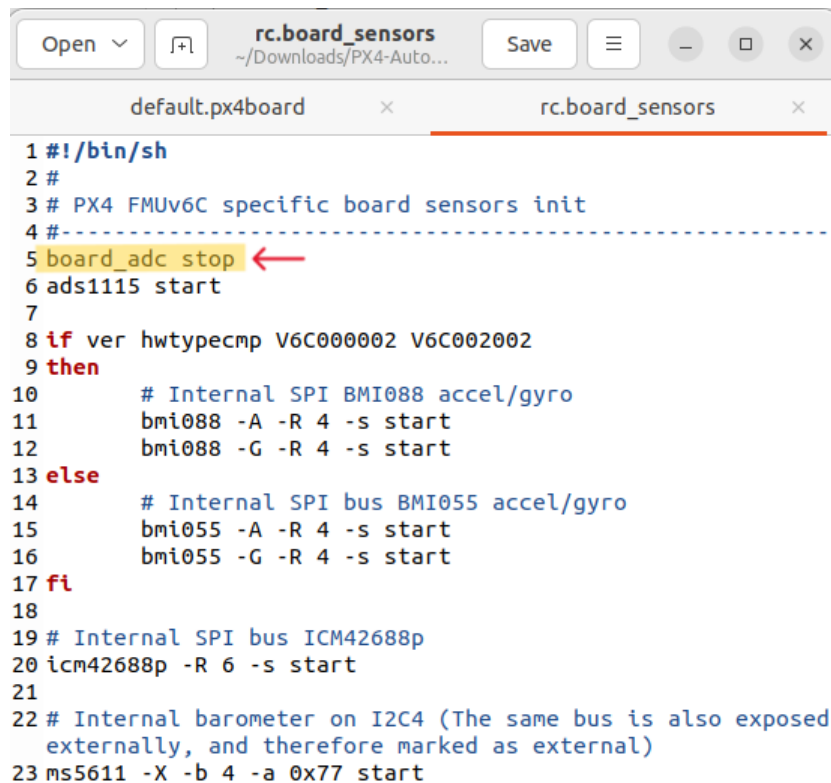
Ensuite en navigant vers le dossier "PX4-Autopilot", nous allons effectuer la compilation du firmware.

- **Mise à jour (flashing) du firmware personnalisé** : Après avoir effectué la compilation, nous allons flasher le firmware PX4 sur notre Pixhawk 6C à l'aide de QGroundControl, les détails existent en Annexe 3.

**Remarque** : Un ADC interne du Pixhawk est activé par défaut, ce ADC est utilisé uniquement pour les données de la batterie et il n'est pas accessible dans notre Pixhawk 6C pour l'utiliser. Comme l'ADS1115 et l'ADC interne partagent le même identifiant, les conflits peuvent être évités en désactivant l'ADC interne avant d'activer l'ADS1115.

Pour réaliser cette étape, nous allons ajouter la commande suivante, **"board\_adc stop"** avant **"ads1115 start"** au script d'initialisation situé au chemin suivant,

**"PX4-Autopilot/boards/px4/fmu-v6c/init/rc.board\_sensors"**, voir figure 3.14.



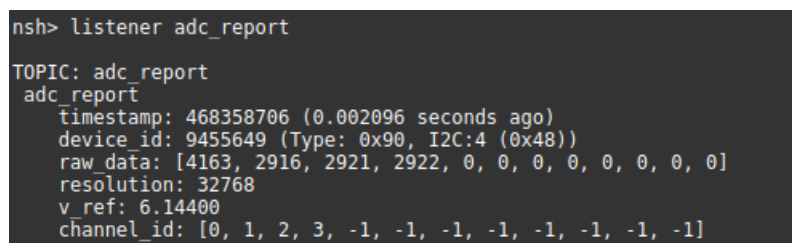
```

1 #!/bin/sh
2 #
3 # PX4 FMUV6C specific board sensors init
4 #-----
5 board_adc stop ←
6 ads1115 start
7
8 if ver hwtypecmp V6C000002 V6C002002
9 then
10     # Internal SPI BMI088 accel/gyro
11     bmi088 -A -R 4 -s start
12     bmi088 -G -R 4 -s start
13 else
14     # Internal SPI bus BMI055 accel/gyro
15     bmi055 -A -R 4 -s start
16     bmi055 -G -R 4 -s start
17 fi
18
19 # Internal SPI bus ICM42688p
20 icm42688p -R 6 -s start
21
22 # Internal barometer on I2C4 (The same bus is also exposed
    externally, and therefore marked as external)
23 ms5611 -X -b 4 -a 0x77 start

```

FIGURE 3.14 – Arrêter l’ADC par défaut

Après avoir terminé la configuration pour le convertisseur ADS1115, nous avons effectué un test rapide afin d’examiner son fonctionnement à travers le console MAVLink de QGroundControl. En écrivant **”listener adc\_report”**, nous allons afficher les données de l’ADS1115 comme indiqué dans la figure 3.15



```

nsh> listener adc_report
TOPIC: adc_report
adc_report
timestamp: 468358706 (0.002096 seconds ago)
device_id: 9455649 (Type: 0x90, I2C:4 (0x48))
raw_data: [4163, 2916, 2921, 2922, 0, 0, 0, 0, 0, 0, 0, 0]
resolution: 32768
v_ref: 6.14400
channel_id: [0, 1, 2, 3, -1, -1, -1, -1, -1, -1, -1, -1]

```

FIGURE 3.15 – L’état de l’ADS1115 dans le console de MAVLink

### 3.4.2.4 Enregistrement des données (Data logging)

Par défaut, l'enregistrement démarre automatiquement pendant la phase d'armement du vol (armement signifie lorsque le contrôleur de vol commence une mission pour les drones) et se termine automatiquement pendant la phase de désarmement. Chaque session d'armement entraîne la création d'un nouveau fichier journal sur la carte microSD.

Nous pouvons manipuler les paramètres de l'enregistrement des données en communiquant directement via le MAVLink console. Par exemple pour vérifier l'état actuel de "data logging", nous écrivons **"logger status"** dans le MAVLink console. De plus, pour commencer immédiatement l'enregistrement, nous pouvons utiliser la commande **"logger on"**. La commande **"logger off"** annule cette opération.

Nous pouvons aussi personnaliser les données à enregistrer, l'état qui déclenche l'enregistrement des données, et d'autres caractéristiques en configurant les paramètres de "data logging" qui existent dans QGroundControl, Les paramètres sont décrites dans l'annexe 3.

#### Data logging pour les données analogiques de l'ADS1115

Nous avons rencontré un problème lors de data logging des données de l'ADS1115, nous avons donc décidé de changer la façon d'enregistrement. Nous avons constaté que nous pouvons personnaliser les sujets enregistrés sur la carte microSD sans avoir besoin d'utiliser les paramètres de data logging qui existent dans QGroundControl. La solution consiste à créer un fichier texte appelé **"etc/logging/logger\_topics.txt"** sur la carte microSD, en listant les sujets que nous souhaitons à enregistrer de la manière suivante,

**<topic\_name> <interval> <instance>**, voir figures 3.16 et 3.17.

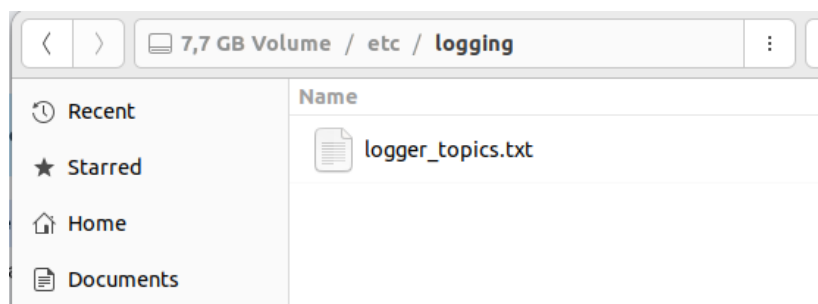


FIGURE 3.16 – Dossier des topics personnalisés

Les données seront stockées dans un fichier uLog dans un dossier appelé **log** qui existe au niveau de la carte microSD.

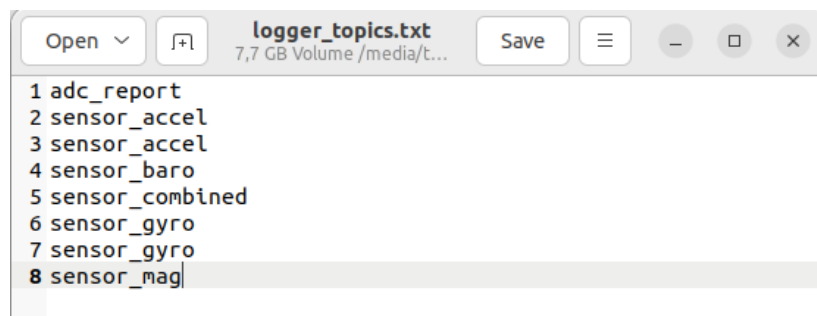


FIGURE 3.17 – Exemple des topics personnalisés

### 3.4.2.5 Télémétrie

Pour la télémétrie, le processus est simple. Tout d'abord, nous allons flasher le firmware de télémétrie (il existe déjà dans QGroundControl, une fois la télémétrie est connectée, le logiciel va le reconnaître directement) sur les deux modules. Ensuite, nous connectons un module au PC et l'autre au Pixhawk. Si les deux LED des modules de télémétrie sont éclairés avec du vert solide comme indiqué dans la figure 3.18, cela signifie que chaque appareil est connecté à l'autre et qu'ils peuvent communiquer ensemble pour envoyer ou recevoir des données.



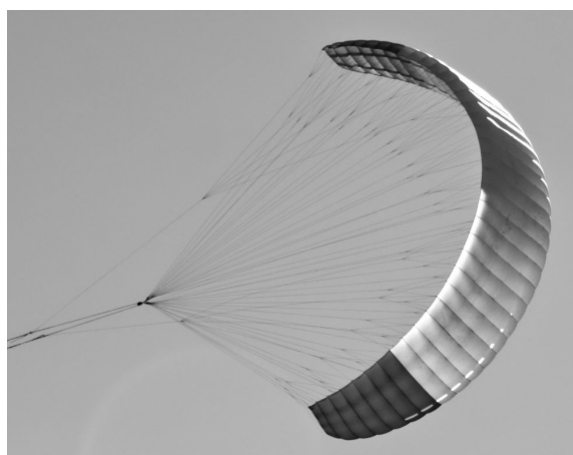
FIGURE 3.18 – Module de télémétrie connecté à l'autre module pour échanger les données

Au cours de cette section, nous avons abordé la configuration et la programmation des principaux capteurs de notre boîtier. Dans la prochaine section, nous allons détailler la partie des tests réalisés.

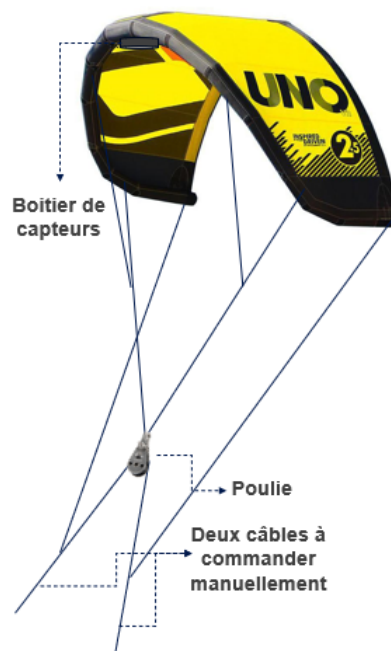
## 3.5 Tests

Après avoir finaliser le boîtier de capteurs, nous avons effectué quelques tests en dehors. Le test consiste à piloter le cerf-volant manuellement. Pour le réaliser, nous avons recours

à une conception de l'entreprise **Kitenergy** avec une petite poulie comme le montre la figure 3.19.



(a) Photo réel du montage de l'entreprise kitenergy, [16]



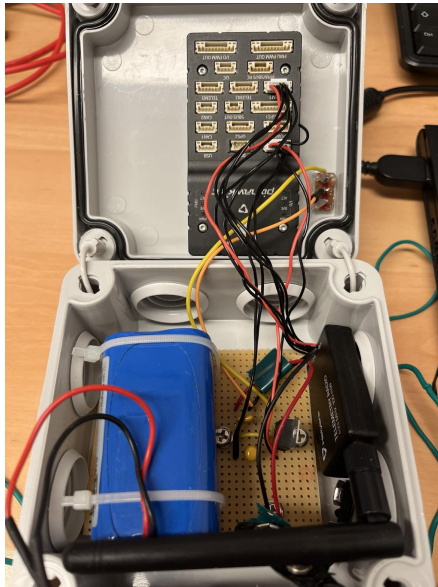
(b) Montage expérimental

FIGURE 3.19 – Montage expérimental pour les tests

Les tests effectués n'étaient pas très fiables vu que les conditions de vent n'étaient pas trop favorables pour voler le cerf-volant. À cause des contraintes matérielles, nous n'avons pas pu tester tous les capteurs mentionnés dans la partie de choix matériel, surtout les jauges contraintes et le capteur de vitesse du vent.

La figure 3.20a montre le boîtier de capteurs, et la figure 3.20b montre la mise en place du boîtier de capteurs sur le cerf-volant.

Les figures 3.21 et 3.22 représentent les accélérations linéaires acquises et les vitesses angulaires acquises par l'IMU au cours d'un test. La lecture des données a été effectuée à l'aide d'un programme Python en utilisant la bibliothèque **pyulog**.



(a) boîtier de capteurs

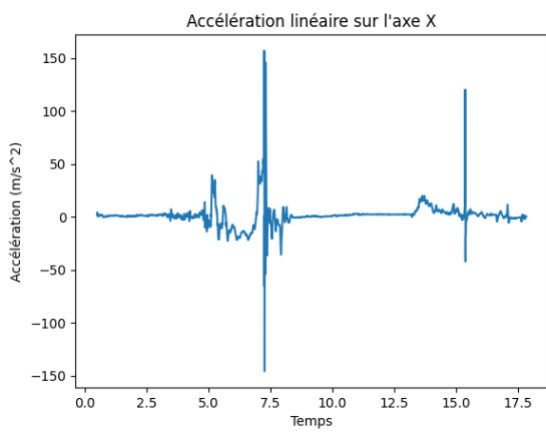


(b) Mise en place du boîtier de capteurs

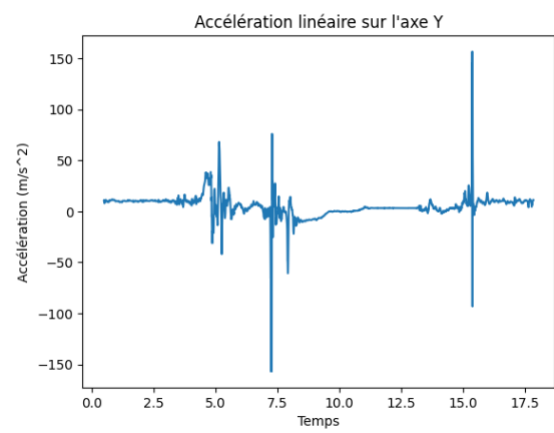
FIGURE 3.20 – Le système d'acquisition de données au cours des tests

## 3.6 Conclusion

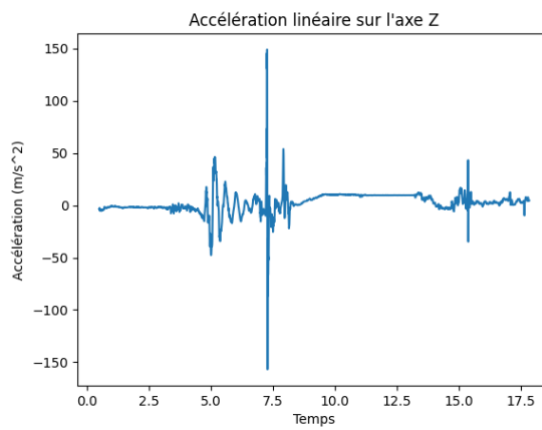
En somme, ce dernier chapitre a mis en oeuvre en premier lieu le choix du matériel, ensuite l'implémentation logicielle pour réaliser le boîtier de capteurs. Nous avons tenter pour effectuer des tests en dehors, mais les conditions du vent n'étaient pas favorable pour valider le fonctionnement des capteurs.



(a) Vitesse angulaire selon l'axe x



(b) Vitesse angulaire selon l'axe y



(c) Vitesse angulaire selon l'axe z

FIGURE 3.21 – Les vitesses angulaires acquises par l'accéléromètre

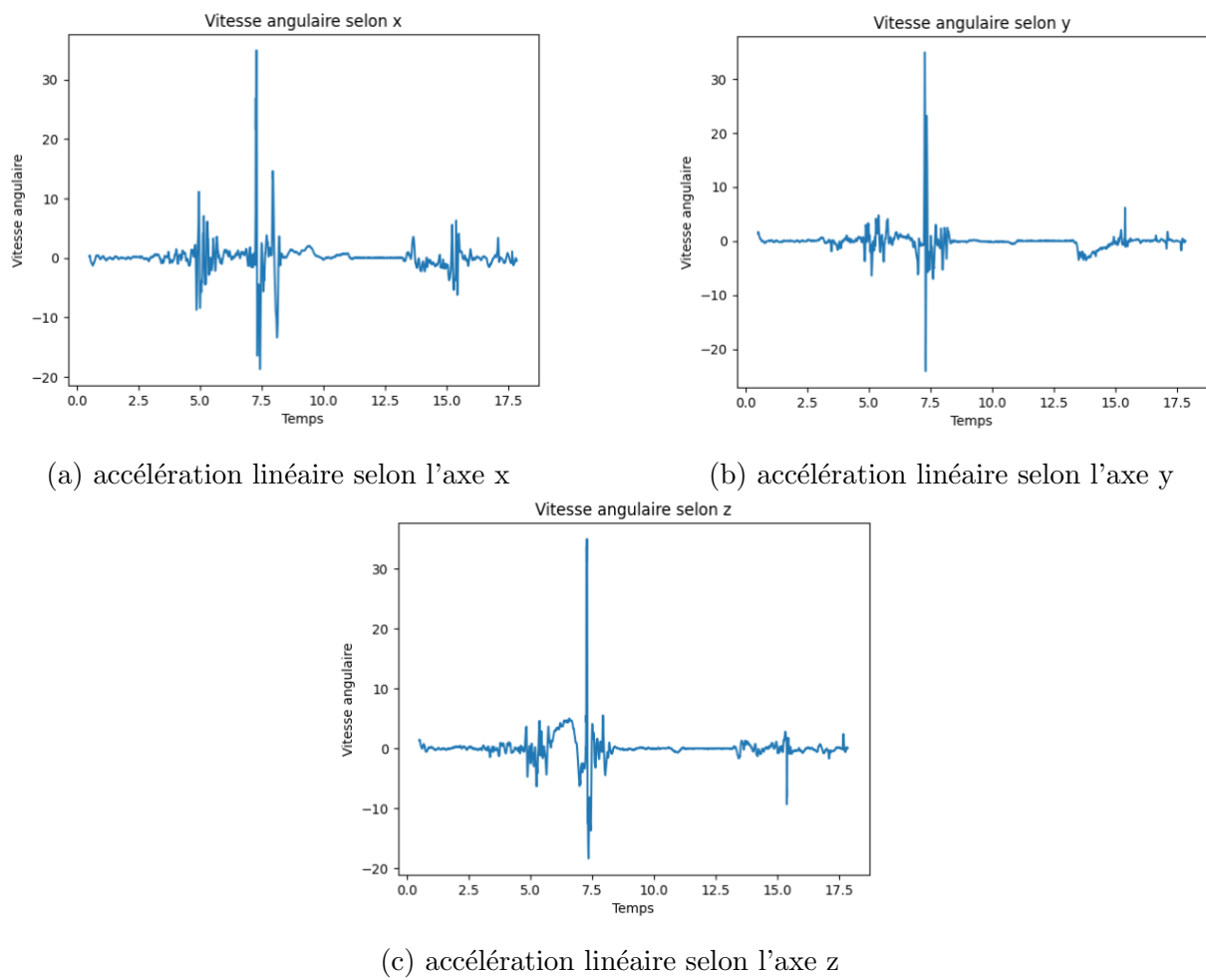


FIGURE 3.22 – Les accélérations linéaires acquises par le gyroscope

# Conclusion générale

En résumé, ce rapport visait à présenter une solution pour concevoir un boîtier d'acquisition de données pour des cerf-volants des éoliennes aéroportées qui est un premier pas dans le projet de la conception d'un prototype open-source pour ce type d'éolienne.

L'objectif principal de ce boîtier est d'avoir des données qui décrivent le comportement de ces systèmes dans l'air, afin de pouvoir valider les modèles dynamiques qui nous permettent par la suite de concevoir des loi de commandes robustes et fiables. L'une des principales limites rencontrées au cours de notre projet était l'absence des conditions de vent favorables pour réaliser des tests en dehors pour valider le fonctionnement des capteurs. Malgré ça, nous avons eu l'opportunité d'effectuer quelques tests en dehors avec des conditions de vent acceptables, ce qui nous a aidé à simuler les difficultés confrontées au cours des tests et les stratégies pour faire décoller le cerf-volant. Au cours de ces tests, nous avons réussi à acquérir les données de l'IMU, les lire et les enregistrer pour les traiter, mais à cause des contraintes matérielles, nous n'avons pas eu la chance pour valider le fonctionnement des autres capteurs.

La commercialisation de ces systèmes fait face à d'autres défis en plus du pilotage et de commande des cerf-volants, tels que le problème de décollage, la durée de vie de ces systèmes, ainsi qu'à l'optimisation des caractéristiques aérodynamiques des dispositifs volants, plusieurs enjeux sont encore en phase de développement, d'où la nécessité de continuer à s'investir dans les recherches scientifiques pour relever ces défis.

# Références Bibliographiques

- [1] PX4 AUTOPILOT. *Code source du driver ADS1115*. <https://github.com/PX4/PX4-Autopilot/tree/main/src/drivers/adc/ads1115>. Dépôt GitHub, Consulté en juin 2024.
- [2] Philip BECHTLE et al. “AWEsome: An open-source test platform for airborne wind energy systems”. In : (2017).
- [3] Roystan Vijay CASTELINO, Yashwant KASHYAP et Panagiotis KOSMOPOULOS. “Airborne Kite Tether Force Estimation and Experimental Validation Using Analytical and Machine Learning Models for Coastal Regions”. In : *Remote Sensing* 14.23 (2022). DOI : 10.3390/rs14236111. URL : <https://doi.org/10.3390/rs14236111>.
- [4] I. CASTRO-FERNÁNDEZ et al. “Automatic testbed with a visual motion tracking system for airborne wind energy applications”. In : *Wind Energy* (2021). DOI : 10.1002/we.2805. URL : <https://onlinelibrary.wiley.com/doi/pdf/10.1002/we.2805>.
- [5] Antonello CHERUBINI et al. “Airborne Wind Energy Systems: A review of the technologies”. In : *Renewable and Sustainable Energy Reviews* 51 (2015). DOI : 10.1016/j.rser.2015.07.026.
- [6] Mitchell COBB, Nihar DEODHAR et Christopher VERMILLION. “Lab-Scale Experimental Characterization and Dynamic Scaling Assessment for Closed-Loop Crosswind Flight of Airborne Wind Energy Systems”. In : *Journal of Dynamic Systems, Measurement, and Control* (2017).
- [7] Tarek N. DIEF et al. “Hardware-in-the-Loop (HIL) and Experimental Findings for the 7 kW Pumping Kite Power System”. In : *Journal of Kite Power Systems* 42.1 (2017).
- [8] Emmanuel DU PONTAVICE. “Propulsion par cerf-volant : envol et pérégrinations”. Thèse de doctorat. France : Université Paris Saclay (COmUE), 2016.

- [9] Lorenzo FAGIANO et Trevor MARKS. “Design of a small-scale prototype for research in airborne wind energy”. In : *IEEE/ASME Transactions on Mechatronics* (2016). URL : <http://ieeexplore.ieee.org/>.
- [10] Lorenzo FAGIANO et al. “A Small-Scale Prototype to Study the Takeoff of Tethered Rigid Aircrafts for Airborne Wind Energy”. In : *Energies* (2015).
- [11] Lorenzo FAGIANO et al. “Automatic crosswind flight of tethered wings for airborne wind energy: modeling, control design and experimental results”. In : (août 2013).
- [12] Lorenzo FAGIANO et al. “Autonomous Airborne Wind Energy Systems: Accomplishments and Challenges”. In : *Annual Review of Control, Robotics, and Autonomous Systems* 5 (2022). First published as a Review in Advance on November 8, 2021. DOI : 10.1146/annurev-control-042820-124658. URL : <https://doi.org/10.1146/annurev-control-042820-124658>.
- [13] All About Circuits FORUM. *ADS1115 Driver for Pixhawk Flight Controller*. Consulté en juin 2024. 2024. URL : <https://forum.allaboutcircuits.com/threads/ads1115-driver-for-pixhawk-flight-controller.164967/>.
- [14] PX4 Discuss FORUM. *ADS1115 for Voltage and Current Reading*. Consulté en juin 2024. 2024. URL : <https://discuss.px4.io/t/ads1115-for-voltage-and-current-reading/25876/4>.
- [15] Kites for FUTURE. *Kites for Future: Flying Wind Turbine*. <https://hackaday.io/project/188136-kites-for-future-flying-wind-turbine>. Hackaday project. 2024.
- [16] KITENERGY. *K100: Innovative Kite-Based Energy Solutions*. [https://kitenrg.com/k100/#gallery-block\\_611ba0167f1f4-1](https://kitenrg.com/k100/#gallery-block_611ba0167f1f4-1).
- [17] Michiel KRUIJFF et al. “A Roadmap Towards Airborne Wind Energy in the Utility Sector”. In : *Renewable and Sustainable Energy Reviews* 87 (2018).
- [18] Baptiste LABAT. *RoboKite: A robotic kite control system*. <https://github.com/baptistelabat/robokite>. GitHub repository. 2021.
- [19] LAGEPP - Laboratoire d’Automatique, de Génie des Procédés et de Génie Pharmaceutique. <https://lagepp.univ-lyon1.fr/le-labo>.
- [20] Rogelio Jr LOZANO. “Etude du vol d’un générateur cerf-volant”. Thèse de doctorat. Grenoble, France : Université de Grenoble, 2014.

- [21] PX4 DOCUMENTATION TEAM. *Accelerometer Setup Guide*. Version 1.12, consulté en Mai 2024. PX4 Autopilot. URL : <https://docs.px4.io/main/en/config/accelerometer.html>.
- [22] PX4 DOCUMENTATION TEAM. *Compass Setup Guide*. Version 1.12, consulté en Mai 2024. PX4 Autopilot. 2024. URL : <https://docs.px4.io/main/en/config/compass.html>.
- [23] Mostafa A. RUSHDI et al. “Towing Test Data Set of the Kyushu University Kite System”. In : (2020).
- [24] Erik SOUKKA. “Chassis Design of a Control Pod for a Kite Power System”. Master Thesis Report at SkySails GmbH, Hamburg. Master Thesis Report. Royal Institute of Technology, Stockholm, 2018.
- [25] Tony Alan WOOD. “Model-Based Flight Control of Kites for Wind Power Generation”. Doctoral thesis. ETH Zurich, 2018.

# Annexe 1

## .1 Détails technologiques des systèmes présentés dans les tableaux de la partie "étude de l'existant" du premier chapitre :

### .1.1 Systèmes ground-gen

#### **Ampyx Power :**

- -durée de vol : 2H
- Poids de l'aéronef : 235 kg

#### **ABB Corporate Research :**

- -type de câble : UHMWPE avec diamètre 3 mm
- Poids de l'aéronef : à peu près 1.2 kg
- Alimentation : 3kW inverter and 12V battery and DC/DC converter 5V.
- Système de propulsion de Glider : Brushless motor D2836, driver, propeller, Li-Poly Battery
- Data-logging : MicroSD (SPI)

#### **Kyushu University :**

- -Poids de KCU(Kite Control Unit) : 3kg
- Durée de vol : 3H
- Protocole de communication : XBEE protocol

#### **SkySails GmbH,Hamburg for Royal Institute of Technology, Stockholm :**

- -châssis en titane
- Pois de l'aéronef : 12.7 kg

## .1. DÉTAILS TECHNOLOGIQUES DES SYSTÈMES PRÉSENTÉS DANS LES TABLEAUX DE LA P

-Charge(Load capacity) : 60 kN

### Universidad Carlos III de Madrid :

- **Autre notes :**

-“visual motion tracking”(VMT) system : 3 caméras + artificial neural network (YO-LOv2) pour suivre la trajectoire du cerf-volant dans l’air.

-Moteurs commandés par les pilotes par un joystick

-On-board data envoyé par télémétrie

-Alimentation : 2 batteries 12V pour le système au sol et une batterie LiPo 7.4V pour le système volant.

### Design of small-scale prototype for research in AWE : Fagiano et Trevor Mark :

- **Actionneurs :**

On-ground steering motor : circuit de commande de moteur : Xenusr XTL-230-36)

center line motors : circuit de commande des moteurs : Xenus Micror XSJ- 230-10)

- **Autre notes :**

-Communication : radio link

-Alimentation : 16 batteries au plomb-acide en parallèle pour construire une alimentation DC 12 V et onduleur sinusoïdal pur 1500W.

## .1.2 Systèmes fly-gen

### University of North Carolina at Charlotte (Mitchell Cobb, Nihar Deodhar, Christopher Vermillion)

- **microcontrôleur** : High performance host/target computer pair for real-time motion capture and closed-loop control (Real time controller using Mathworks (Natick, MA) SIMULINK Real-TimeVR software.

- **Autre notes** : Motion capture system pour suivre le dispositif volant dans l’air : Trois caméras high-speed pour l’acquisition d’images puis ils les traitent avec l’ordinateur.

## Annexe 2

### .2 Les principaux caractéristiques des capteurs à partir des fiches techniques

#### .2.1 Accel/Gyro à 3 axes : BMI055

OUTPUT SIGNAL ACCELEROMETER						
Parameter	Symbol	Condition	Min	Typ	Max	Units
Sensitivity	S <sub>2g</sub>	g <sub>FS2g</sub> , T <sub>A</sub> =25°C		1024		LSB/g
	S <sub>4g</sub>	g <sub>FS4g</sub> , T <sub>A</sub> =25°C		512		LSB/g
	S <sub>8g</sub>	g <sub>FS8g</sub> , T <sub>A</sub> =25°C		256		LSB/g
	S <sub>16g</sub>	g <sub>FS16g</sub> , T <sub>A</sub> =25°C		128		LSB/g

FIGURE 23 – Accéléromètre : BMI055

OUTPUT SIGNAL GYROSCOPE						
Parameter	Symbol	Condition	Min	Typ	Max	Unit
Sensitivity		T <sub>a</sub> =25°C, R <sub>FS2000</sub>		16.4		LSB/°/s
		T <sub>a</sub> =25°C, R <sub>FS1000</sub>		32.8		LSB/°/s
		T <sub>a</sub> =25°C, R <sub>FS500</sub>		65.6		LSB/°/s
		T <sub>a</sub> =25°C, R <sub>FS250</sub>		131.2		LSB/°/s
		T <sub>a</sub> =25°C, R <sub>FS125</sub>		262.4		LSB/°/s

FIGURE 24 – Gyroscope : BMI055

#### .2.2 Accel/Gyro à 3 axes : ICM-42688-P

## .2. LES PRINCIPAUX CARACTÉRISTIQUES DES CAPTEURS À PARTIR DES FICHES TECHNIQUES

ACCELEROMETER SENSITIVITY				
Full-Scale Range	ACCEL_FS_SEL =0		±16	g
	ACCEL_FS_SEL =1		±8	g
	ACCEL_FS_SEL =2		±4	g
	ACCEL_FS_SEL =3		±2	g
ADC Word Length	Output in two's complement format		16	bits
Sensitivity Scale Factor	ACCEL_FS_SEL =0		2,048	LSB/g
	ACCEL_FS_SEL =1		4,096	LSB/g
	ACCEL_FS_SEL =2		8,192	LSB/g
	ACCEL_FS_SEL =3		16,384	LSB/g

FIGURE 25 – Accéléromètre : ICM-42688-P

GYROSCOPE SENSITIVITY				
Full-Scale Range	GYRO_FS_SEL=0		±2000	°/s
	GYRO_FS_SEL=1		±1000	°/s
	GYRO_FS_SEL=2		±500	°/s
	GYRO_FS_SEL=3		±250	°/s
	GYRO_FS_SEL=4		±125	°/s
	GYRO_FS_SEL=5		±62.5	°/s
	GYRO_FS_SEL=6		±31.25	°/s
	GYRO_FS_SEL=7		±15.625	°/s
Gyroscope ADC Word Length	Output in two's complement format		16	bits
Sensitivity Scale Factor	GYRO_FS_SEL=0		16.4	LSB/(°/s)
	GYRO_FS_SEL=1		32.8	LSB/(°/s)
	GYRO_FS_SEL=2		65.5	LSB/(°/s)
	GYRO_FS_SEL=3		131	LSB/(°/s)
	GYRO_FS_SEL=4		262	LSB/(°/s)
	GYRO_FS_SEL=5		524.3	LSB/(°/s)
	GYRO_FS_SEL=6		1048.6	LSB/(°/s)
	GYRO_FS_SEL=7		2097.2	LSB/(°/s)

FIGURE 26 – Gyroscope : ICM-42688-P

### .2.3 GPS M8N

GNSS		GPS & GLONASS	GPS	GLONASS	BeiDou	Galileo
Horizontal position accuracy <sup>3</sup>		2.5 m	2.5 m	4 m	3 m	3 m
	With SBAS	2.0 m	2.0 m	-	-	-

FIGURE 27 – GPS M8N

### .2.4 Jauge de contrainte WIKA F2812

<b>Model F2812</b>	
<b>Rated force <math>F_{nom}</math> N</b>	50 / 100 / 150 / 200 / 300 / 500 / 600 / 1,000
<b>Relative linearity error <math>d_{lin}^{1)}</math></b>	$\pm 0.5 \% F_{nom}$
<b>Relative reversibility error <math>v</math></b>	$\pm 0.5 \% F_{nom}$
<b>Relative repeatability error in unchanged mounting position <math>b_{rg}</math></b>	$\pm 0.25 \% F_{nom}$
<b>Relative deviation of zero signal <math>d_{S,0}</math></b>	$\pm 2 \% F_{nom}$
<b>Temperature effect on zero signal <math>TK_0</math></b>	$\leq \pm 0.2 \% / 10 K$
<b>Temperature effect on characteristic value <math>TK_C</math></b>	$\leq \pm 0.2 \% / 10 K$
<b>Force limit <math>F_L</math></b>	120 % $F_{nom}$
<b>Breaking force <math>F_B</math></b>	200 % $F_{nom}$
<b>Material of the measuring body</b>	Stainless steel
<b>Rated temperature range <math>B_{T, nom}</math></b>	-10 ... +40 °C
<b>Operating temperature range <math>B_{T, G}</math></b>	-20 ... +80 °C
<b>Input resistance <math>R_e</math></b>	700 $\pm$ 30 $\Omega$
<b>Output resistance <math>R_a</math></b>	700 $\pm$ 5 $\Omega$
<b>Insulation resistance <math>R_{is}</math></b>	$\geq 5,000 M\Omega/DC 100 V$
<b>Output signal (rated characteristic value) <math>C_{nom}</math></b>	2.0 $\pm$ 0.2 mV/V
<b>Electrical connection</b>	Measuring cable $\varnothing 3 \times 3,000$ mm
<b>Voltage supply</b>	
Standard	DC 5 ... 10 V
Option	DC 12 ... 28 V integrated or cable amplifier 0 (4) ... 20 mA DC 0 ... 10 V DC 0 ... 5 V
<b>Ingress protection (per IEC/EN 60529)</b>	IP65
<b>Weight in kg</b>	0.1

FIGURE 28 – Jauge de contrainte WIKA F2812

## Annexe 3

### .3 Installation de Git

Pour cloner le dépôt, nous devons installer Git, un système de contrôle de version qui gère et suit les changements dans les projets de développement logiciel.

Pour installer Git, nous écrivons dans la fenêtre de commande :

- Mettre à jour l'index du package **"sudo apt-get update"**
- Installer Git sur Ubuntu par **"sudo apt-get install git"**
- Ensuite, vérifier l'installation par **"git --version"**

### .4 Mise à jour firmware PX4 à l'aide de QGroundControl

- Dans QGroundControl, nous naviguons à la section "Firmware Setup".



FIGURE 29 – Première étape pour la mise à jour de PX4

- Sélectionner "Advanced Settings" pour activer la mise à jour d'un fichier personnalisé du firmware.
- Après nous sélectionnons le fichier **"firmware\_fmuv6c.px4"**.
- Enfin, nous attendons que la mise à jour soit terminée.

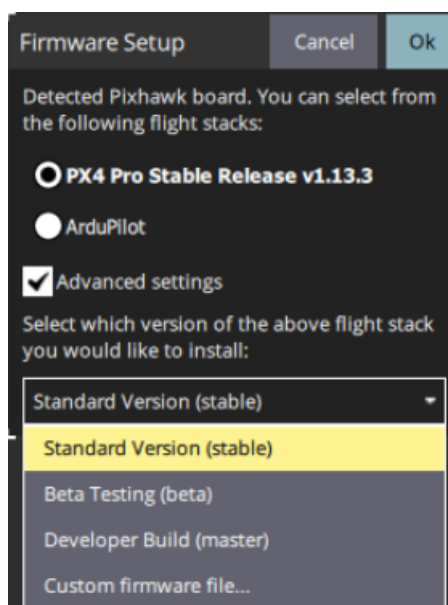


FIGURE 30 – Deuxième étape pour la mise à jour de PX4

## .5 Les paramètres de "SD Logging" qui existe dans QGround-Control

- **SDLOG\_BOOT\_BAT** : Ce paramètre contrôle la possibilité de démarrage automatique le data logging lorsque la batterie est connectée.
- **SDLOG\_DIRS\_MAX** : Le nombre maximum de répertoires pouvant être créés sur la carte SD pour stocker les fichiers ulog.
- **SDLOG\_MISSION** : Ce paramètre détermine si les données de mission sont enregistrées (utilisées en particulier avec les drones).
- **SDLOG\_MODE** : Ce paramètre spécifie le moment où le data logging commence et s'arrête en fonction de l'état du contrôleur de vol.
- **SDLOG\_PROFILE** : Ce paramètre spécifie l'ensemble des sujets et leurs taux de journalisation.
- **SDLOG\_UTC\_OFFSET** : ajuste le décalage horaire UTC appliqué aux horodatages enregistrés dans les fichiers.
- **SDLOG\_UUID** : active ou désactive l'utilisation d'un UUID (Universally Unique Identifier) pour l'identification du fichier journal.