
Mehrzweck-Drohne

Projet 6 : Réunion avec l'expert - n° 1

Membre du projet	Haberkorn Jonas
Client	Nouri T., Prof. Dr. Dip. Eng. Elec./Phys.
Cursus d'études	Bachelor Elektro- und Informationstechnik – I.C.S.
Date	Jeudi, le 17 Mai 2018

Sommaire

1	Introduction	2
2	Définition des tâches.....	3
3	Objectifs	4
3.1	Objectifs obligatoires.....	4
3.2	Objectifs souhaités.....	4
3.3	Non-Objectifs.....	4
4	Gestion du projet	5
4.1	Organisation planifiée.....	5
4.2	Journal	5
4.3	Mindmap.....	6
4.4	Restrictions	7
4.5	Diagramme GANT	8
5	Environnement du projet.....	9
5.1	Le drone	9
5.2	Les logiciels	10
5.3	Le firmware	11
6	Avancement du projet.....	12
6.1	Afficher les données en direct sur Mission Planner	12
6.2	Communication par télémétrie	13
6.3	Réalisation des modules.....	14
6.4	Communication entre l'APM et le contrôleur de module.....	15
6.5	Affichage des données GPS sur une carte.....	15
7	Mesures mises en place pendant le projet.....	17

1 Introduction

Ce document fait office d'introduction aux objectifs, à l'organisation et à l'environnement de la Thèse « Mehrzweck-Drohne ». Il reprend donc la définition des tâches, les objectifs, l'organisation, une présentation simplifiée de l'environnement du projet, ainsi que l'avancement actuel au moment de cette réunion.

Présentation

- Cette thèse a pour objectif de rendre un drone modulable afin que l'on puisse réaliser différentes tâches, sans avoir à changer de drone. Pour ce faire, plusieurs modules interchangeables seront créés.

Dates

- Début : Lundi, le 5 Mars 2018
- Fin : Vendredi, le 17 Août 2018

Domaines impliqués

- Développement de software (Mission planner)
- Programmation de microcontrôleurs (APM 2.5)
- Gestion et travail de données (Données GPS)
- Communication (Télémetrie, Uart)

2 Définition des tâches

La suite de cette page est une traduction fidèle de la définition des tâches du client.

Accompagnateur : Prof. Dr. Taoufik Nouri, IMVS, Nouri@Nouri.ch

Client : Prof. Dr. Taoufik Nouri, Nouri@Nouri.ch

Expert : Prof. Dr. Mauricio Reyes, mauricio.reyes@istb.unibe.ch

Etudiant : Haberkorn Jonas, jonas.haberkorn@students.fhnw.ch

Durée : 19.2.18 -17.8.18

Situation initiale

Ce travail est la succession du P5.

Objectif du travail

Les tâches suivantes doivent être réalisées.

1. Construire un drone (plaque de base, électronique, alimentation, contrôle à distance, avec liaison radio, moteurs, etc.), qui a été réalisé en partie dans le P5, HS17.
2. Etudier le programme de contrôle et vous familiariser avec les modifications possibles du programme.
3. Parcourir une zone et afficher les données GPS sur une carte (pathfollow existe déjà) puis les sauvegarder dans un fichier/BD.
4. **Equipement :** Le drone doit effectuer au moins quatre tâches différentes : transport et dépôt d'objets, y compris les personnes, extincteurs : fumée, gaz, transport d'eau, engrais, etc. Ici, nous avons besoin de construire davantage de matériel et de développer les logiciels.
5. **Mission en temps réel :** les données : La tâche, l'équipement, le temps de vol, le GPS, le produit chargé, la quantité, la durée, l'état (eau ouverte, chargée, vide, etc.) doivent être affichés à l'écran en temps réel. Cette tâche pourrait être considérée comme une extension d'un cockpit existant.
6. **Boîte noire :** Les données du point 4 doivent être stockées dans un fichier/BD, comme un journal.

Technologies/Principaux domaines/Références

Intelligence artificielle, Drone, Electronique, Programmation

3 Objectifs

Les objectifs de ce projet peuvent être divisés en trois catégories différentes (objectifs obligatoires, objectifs souhaités et non-objectifs).

3.1 Objectifs obligatoires

- Monter un drone et le mettre en service.
- Etre capable de définir un plan de vol à l'avance pour que le drone puisse suivre le chemin seul.
- Affichez les données GPS en temps réel sur une carte et les sauvegarder pour une utilisation ultérieure.
- Quatre tâches différentes doivent être possibles avec le drone.
- Les données sur l'état de la tâche et du drone doivent être affichées et stockées en temps réel sur la station de contrôle au sol (GSC).

3.2 Objectifs souhaités

- Affiche les données GPS et les données d'état stockées sur une carte.
- Génération automatique d'un rapport de vol après une mission.

3.3 Non-Objectifs

- Le drone ne doit pas être complètement reconstruit. Le drone du projet 5 est disponible et doit être développé.
- Le contrôleur de vol et les méthodes de suivi de trajectoire ne devraient pas être recréés. Les contrôleurs et méthodes existants sont utilisés.

4 Gestion du projet

Le projet est organisé selon certaines méthodes et obligations.

4.1 Organisation planifiée

Le projet suit un mélange de la méthode du chemin critique et de la méthode « traditionnelle ».

Les tâches sont réparties sur la période du projet en fonction de la date de fin du projet et de la durée estimée. Il n'y a pas de formule exacte pour calculer le temps d'une tâche, mais ce temps a été estimé en fonction de l'expérience personnelle.

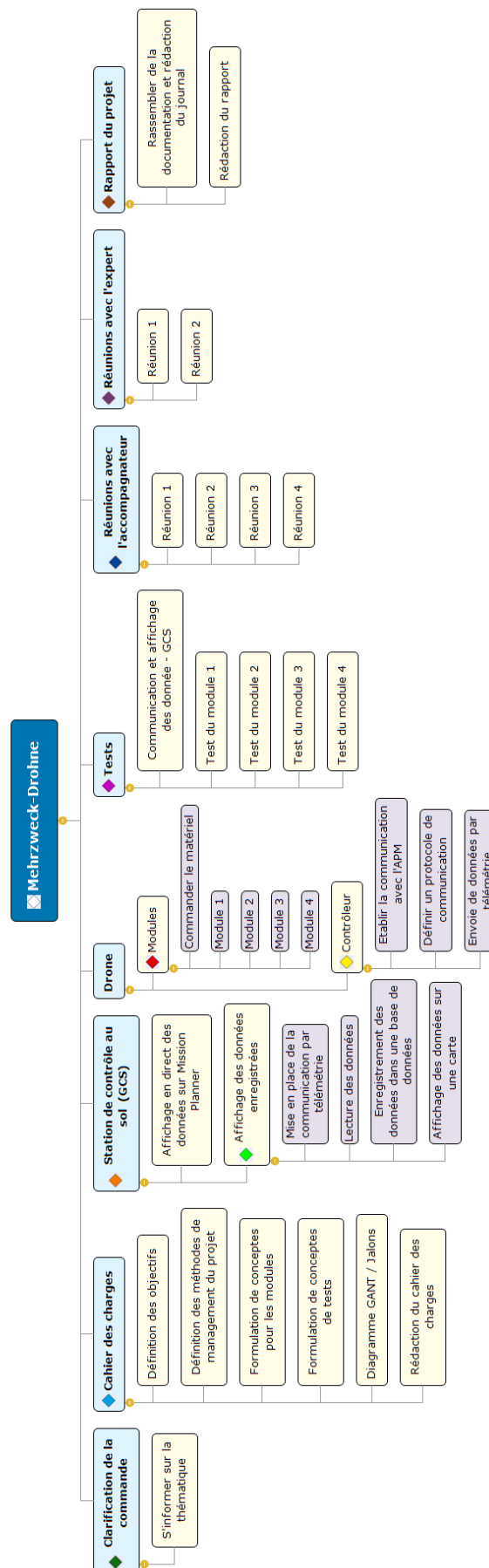
Comme les modules sont construits sur la même base (même support, même contrôleur, même connexion avec le drone, etc.), il faut un peu moins de temps pour implémenter chaque module après les premiers.

Des jalons seront atteints avec la validation de chaque test.

4.2 Journal

Au cours du projet, un journal personnel est rédigé, qui contient les nouvelles et les expériences les plus importantes. Le déroulement du projet sera donc plus facilement compréhensible et les tâches de mise en œuvre ou de protocole peuvent être commentées.

4.3 Mindmap



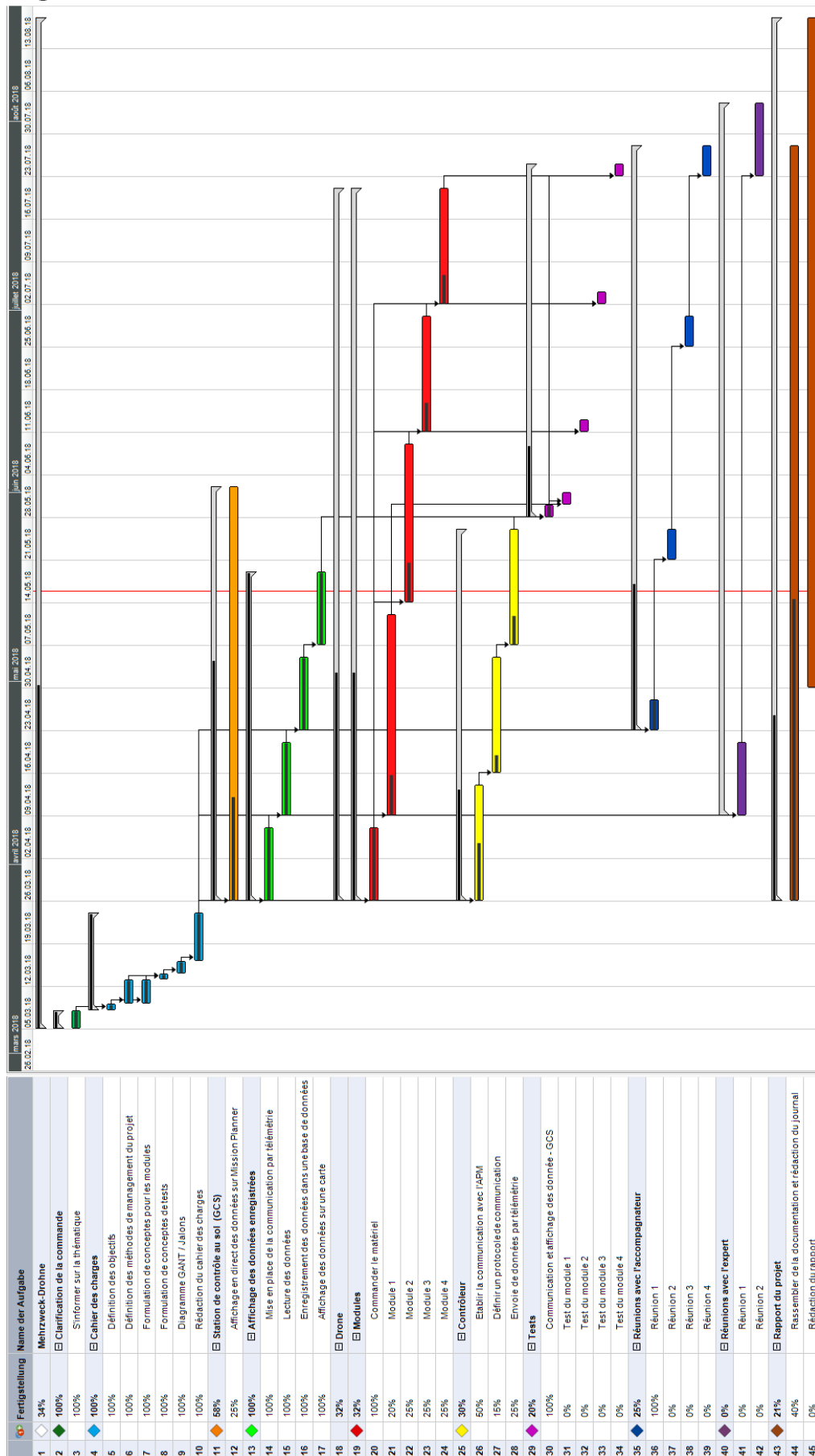
4.4 Restrictions

Ce projet est soumis aux restrictions suivantes :

- Fin du projet : 17.08.2018
- Budget : pas encore connu
- Cours et examens pendant le semestre.

Celles-ci influencent le déroulement du projet ainsi que le choix des variantes et des équipements. Il est donc difficile de définir un bon calendrier à l'avance, mais les méthodes de gestion de projet mentionnées ci-dessus constituent une bonne base pour réaliser une estimation générale.

4.5 Diagramme GANT



5 Environnement du projet

L'environnement du projet joue un rôle très important, il peut grandement influencer la mise en œuvre du projet.

5.1 Le drone

Le drone est un hexacopteur avec un ArduPilot Mega (APM) 2.5 comme contrôleur de vol. Il se compose, entre autres, d'un GPS externe, d'une batterie 4000 mAh, de 6 moteurs brushless, de 6 variateurs électroniques de vitesse et d'un module de puissance (pour alimenter le contrôleur avec la puissance appropriée).

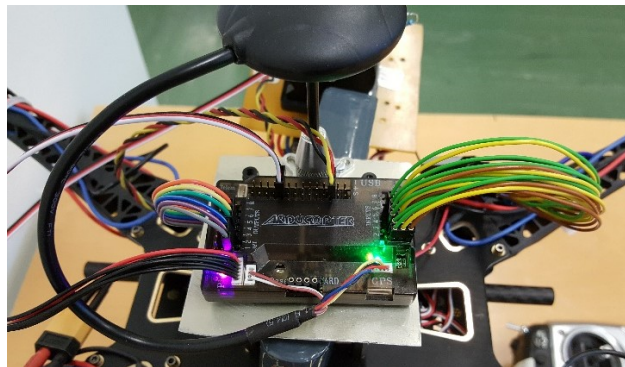


Image 1 : Partie commande du drone

Le drone peut être contrôlé par une télécommande (la Graupner GR-12 HoTT). Le récepteur est un récepteur à 6 canaux. Pour déplacer le drone, 4 canaux sont nécessaires, il y a donc 2 canaux disponibles pour contrôler les modules.

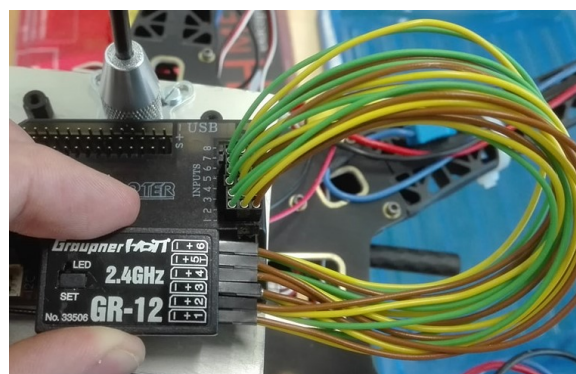


Image 2 : récepteur à 6 canaux

5.2 Les logiciels

Le logiciel « Mission Planner » est utilisé comme station de contrôle au sol. Très utile et pratique, ce logiciel permet d'effectuer le calibrage, d'afficher les données du vol en temps réel et de générer des plans de vol.



Image 3 : Mission Planner

L'IDE Eclipse et Java 8 sont utilisés pour tout programme Java. Eclipse est très pratique pour gérer différents projets.

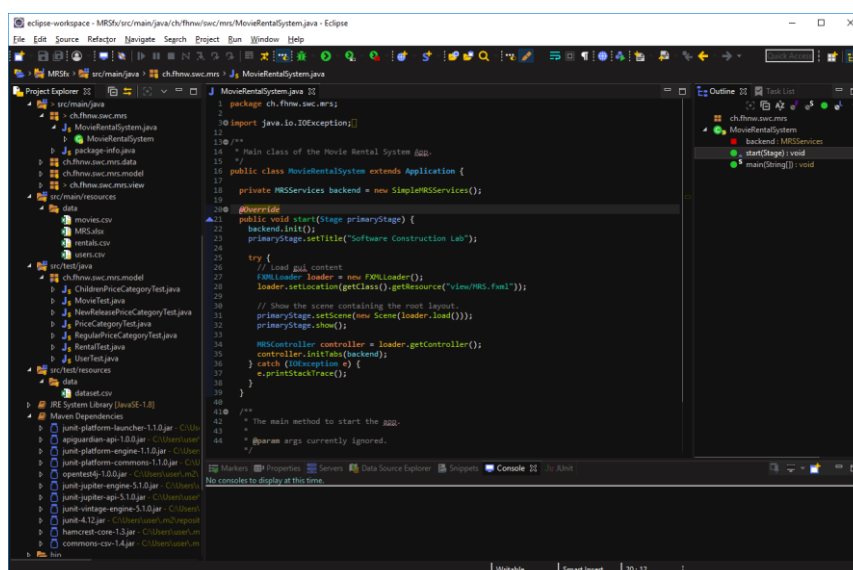


Image 4 : Eclipse IDE

5.3 Le firmware

La dernière version du firmware ArduCopter compatible avec un APM 2.5 est la 3.2.1. Le firmware est chargé depuis les projets ArduCopter 3.2.1 vers le contrôleur en utilisant une toolchain (MHV AVR) et le Makefile présent.

```
fhnw@ubuntu:~/Desktop/ardupilot/ArduCopter$ sudo make upload
fatal: ambiguous argument 'HEAD': unknown revision or path not in the working tree.
Use '--' to separate paths from revisions, like this:
'git <command> [<revision>...] -- [<file>...]'
// BUILDROOT=/tmp/ArduCopter.build HAL_BOARD=HAL_BOARD_APM2 HAL_BOARD_SUBTYPE= TOOLCHAIN=AVR
EXTRAFLAGS=-DGIT_VERSION="HEAD"
building /tmp/ArduCopter.build/ArduCopter.cpp
%% ArduCopter.cpp
/usr/bin/avrdude -c wiring -p atmega2560 -P /dev/ttyACM0 -b115200 -U flash:w:/tmp/ArduCopter
.build/ArduCopter.hex:i

avrdude: AVR device initialized and ready to accept instructions

Reading | ##### | 100% 0.02s

avrdude: Device signature = 0x1e9801 (probably m2560)
avrdude: NOTE: "flash" memory has been specified, an erase cycle will be performed
To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "/tmp/ArduCopter.build/ArduCopter.hex"
avrdude: writing flash (243054 bytes):

Writing | ##### | 68% 26.50s
```

Image 5 : téléversement d'ArduCopter 3.2.1 grâce au makefile

6 Avancement du projet

Ici est exposé, dans son ensemble, l'avancement actuel du projet pour différentes tâches.

6.1 Afficher les données en direct sur Mission Planner

- Le projet open source de Mission Planner a été récupéré et compilé localement.
- Le projet étant très vaste, seul une petite partie a été étudiée pour l'instant.
- L'affichage de texte sur le cockpit a été réussi, ainsi que l'ouverture d'une fenêtre si l'on clique sur ce texte (pour l'affichage potentiel de données autre que le module actuel).

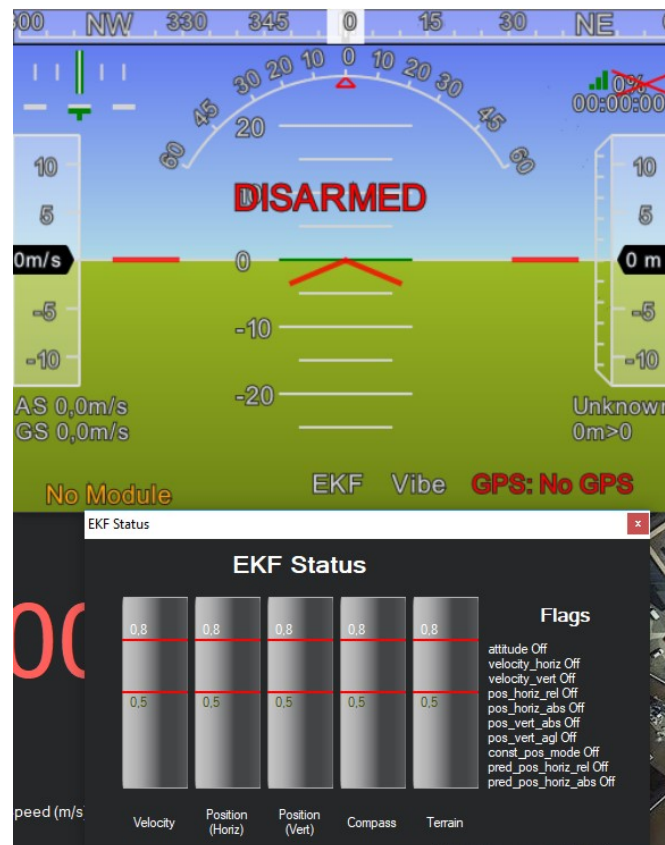


Image 6: Affichage de texte et ouverture de fenêtre sur le cockpit

- La récupération de données provenant de la télémétrie ne fonctionne pas encore, les valeurs affichées sont statiques et artificielles pour l'instant.

6.2 Communication par télémétrie

- Comme l'APM possède un port de télémétrie à 5 pins, et l'APM un de 6 pins, il a fallu réaliser un adaptateur.
- Les antennes arrivent à se trouver et se connectent.
- Il est possible d'établir une connexion par protocole MAVLink entre Mission Planner et l'APM.
- L'envoi de données autres que celles déjà présente dans la télémétrie est mis en place.

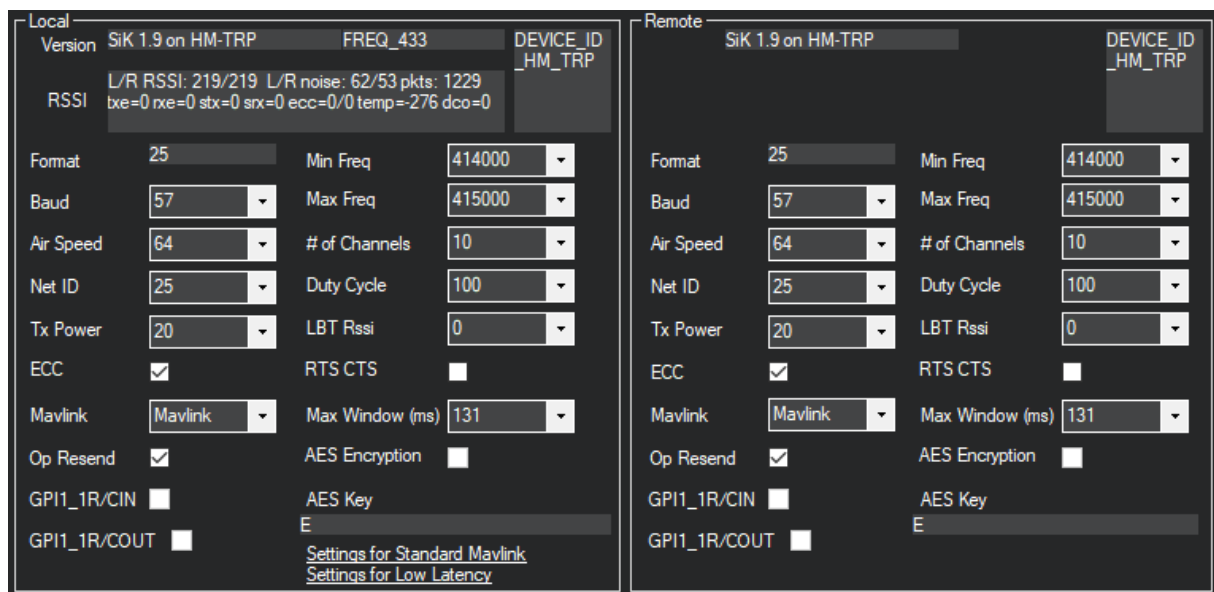


Image 7 : Configuration des antennes

6.3 Réalisation des modules

- Le matériel principal (moteurs, pompe et contrôleurs) a été commandé il y a plus d'un mois, mais certains colis sont à l'heure actuelle toujours en transit.
- Des solutions techniques ont été étudiées afin d'anticiper le besoin de certaines pièces.
- La base des modules sera en plexiglass et est déjà réalisée.
- La programmation des contrôleurs est prise en main.

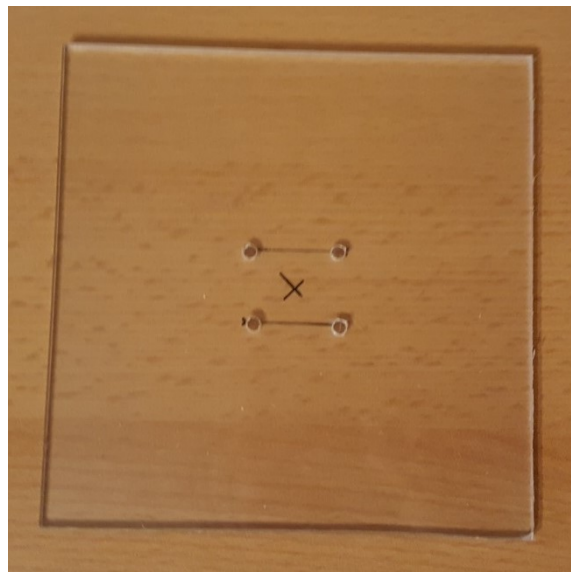


Image 8 : Base des modules

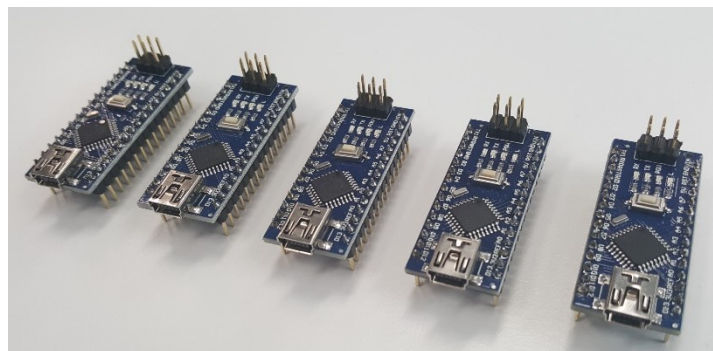


Image 9 : Contrôleurs des modules

6.4 Communication entre l'APM et le contrôleur de module

- L'APM possède des pins déjà soudés sur le port de l'UART 2, ceux-ci sont donc utilisés.
- Pour le Contrôleur, la communication se fera par le port série (Rx, Tx).
- Les messages envoyés de l'APM vers les contrôleurs sont bien réceptionnés, mais dans le sens inverse, des problèmes persistent.

6.5 Affichage des données GPS sur une carte

- Les données, en provenance de l'APM, récupérés par Mission Planner, sont enregistré dans un fichier de logs.
- Grâce à un programme Java développé pour ce projet, ces données sont d'abord convertis en fichier .csv, puis en .kml (un format de fichier utilisé par Google Earth pour l'affichage de trajets et/ou formes sur la carte).
- Google Earth permet également l'affichage du trajet dans la vue « Street view ».

444,7.356025,47.91786,4.170000	ID	LONGITUDE	LATITUDE	ALTITUDE
445,7.356027,47.91786,4.200000	444	7.356025	47.91786	4.17
446,7.356029,47.91786,4.220000	445	7.356027	47.91786	4.2
447,7.356030,47.91786,4.300000	446	7.356029	47.91786	4.22
448,7.356031,47.91786,4.450000	447	7.35603	47.91786	4.3
449,7.356031,47.91786,4.560000	448	7.356031	47.91786	4.45
451,7.356030,47.91786,4.580000	449	7.356031	47.91786	4.56
452,7.356027,47.91786,4.580000	451	7.35603	47.91786	4.58
453,7.356024,47.91786,4.580000	452	7.356027	47.91786	4.58
454,7.356019,47.91787,4.560000	453	7.356024	47.91786	4.58
455,7.356015,47.91787,4.550000	454	7.356019	47.91787	4.56
456,7.356009,47.91787,4.550000	455	7.356015	47.91787	4.55
457,7.356003,47.91787,4.550000	456	7.356009	47.91787	4.55
458,7.355996,47.91787,4.570000	457	7.356003	47.91787	4.55
460,7.355984,47.91787,4.570000	458	7.355996	47.91787	4.57
462,7.355978,47.91788,4.540000	460	7.355984	47.91787	4.57
464,7.355976,47.91788,4.500000	462	7.355978	47.91788	4.54
466,7.355977,47.91788,4.500000				
468,7.355980,47.91788,4.510000				
470,7.355984,47.91788,4.510000				
472,7.355988,47.91789,4.500000				
474,7.355992,47.91788,4.480000				
476,7.355997,47.91788,4.450000				
478,7.356001,47.91788,4.430000				

Image 10 : Exemple de fichier .csv généré et extrait de la base de données

```
<Placemark>
  <name>Stage number 1</name>
  <visibility>0</visibility>
  <description>Track generated with the converter.</description>
  <LookAt>
    <longitude>7.356025</longitude>
    <latitude>47.91786</latitude>
    <altitude>4.17000050</altitude>
    <heading>0</heading>
    <tilt>0</tilt>
    <range>550</range>
  </LookAt>
  <styleUrl>#1</styleUrl>
```

Image 11 : Exemple de fichier .kml généré

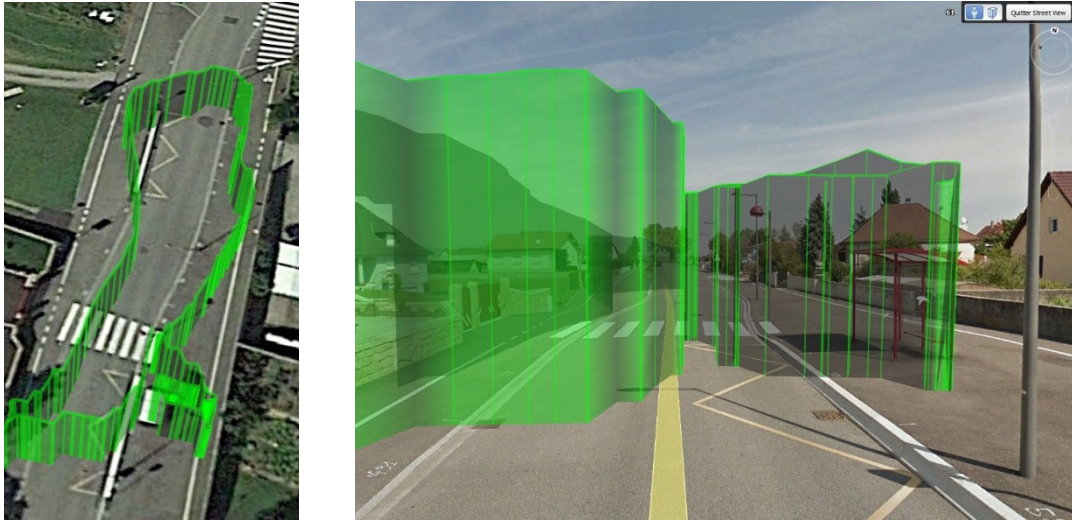


Image 12: Exemple d'affichage de trajets généré sur Google Earth

7 Mesures mises en place pendant le projet

- Comme beaucoup de temps est perdu lors d'essai de différentes méthodes de réalisation dans le programme du contrôleur ou de Mission Planner. Et comme une majeure partie d'entre elles sont infructueuses, une partie du matériel a été apporté au domicile pour pouvoir y travailler le soir ou les week-ends (le matériel n'étant pas encombrant, il est facilement transportable lors du travail à l'université).
- Comme, lors de la semaine de projet, les pièces pour les modules n'étaient pas encore arrivées, le travail s'est concentré sur d'autres tâches qui n'étaient pas encore prévu (en l'occurrence, l'affichage des données sur une carte).
- Etant donné la complexité et le nombre d'erreurs de compilation qui sont survenues lors de l'édition du code de Mission Planner, un dépôt sur GitHub a été ouvert, afin d'avoir un contrôle de version efficace et pouvoir revenir facilement en arrière en cas de problème.
- Vu que l'avancement du projet est en retard par rapport aux planifications, le travail se concentre à présent plus sur une tâche, plutôt que sur plusieurs tâches en parallèle. Cela permet d'avancer plus rapidement sur un des objectifs au lieu de passer d'un travail à un autre tous les quelques jours.