

ECOLE DES MÉTIERS DE LAUSANNE  
ECOLE SUPÉRIEURE

GÉNIE ÉLECTRIQUE

SYSTÈME D'ENREGISTREMENT DE TRAJECTOIRES DE VOL

---

## Boîte noire miniaturisée

---

*Auteur*

Ali ZOUBIR

*Superviseur*

Juan José MORENO

*Mandataire*

ASSOCIATION POUR LE MAINTIEN  
DU PATRIMOINE AÉRONAUTIQUE

11 août 2023



# Table des matières

<b>Acronymes</b>	<b>4</b>
<b>Glossaire</b>	<b>4</b>
<b>1 Cahier des charges</b>	<b>5</b>
1.1 Introduction . . . . .	5
1.2 Aperçu . . . . .	5
1.3 Tâches à réaliser . . . . .	5
1.4 Schéma de principe . . . . .	6
1.5 Planification . . . . .	7
1.6 Livrable . . . . .	7
<b>2 Pré-étude</b>	<b>8</b>
2.1 Fonctionnement du système . . . . .	8
2.2 Schéma bloc . . . . .	8
2.3 Description des blocs . . . . .	8
2.4 Choix des composants et technologies . . . . .	9
2.4.1 Microcontrôleur . . . . .	9
2.4.2 Centrale inertielle . . . . .	10
2.4.3 GPS / GNSS . . . . .	11
2.4.4 Carte SD . . . . .	12
2.5 Batterie, autonomie, charge et régulation . . . . .	13
2.6 Systèmes d'économie d'énergies . . . . .	15
2.7 Diagramme des états du système . . . . .	15
2.8 Estimation des coûts . . . . .	16
<b>3 Conclusion de la Pré-étude</b>	<b>16</b>
<b>4 Développement du schéma électronique</b>	<b>17</b>
4.1 Blocs développés . . . . .	17
4.2 Microcontrôleur . . . . .	18
4.2.1 Connexion . . . . .	18
4.2.2 Programmateur et reset . . . . .	18
4.2.3 LED de vie . . . . .	19
4.3 Périphériques . . . . .	20
4.3.1 Carte SD . . . . .	20
4.3.2 Centrale inertielle . . . . .	20
4.3.3 GNSS . . . . .	21
4.3.4 USB - FTDI . . . . .	21
4.4 Alimentations . . . . .	22
4.4.1 Chargeur de batterie . . . . .	22
4.4.2 Système d'enclenchement . . . . .	23
4.5 Conclusion et perspectives de l'étude . . . . .	23

<b>5 Développement du PCB</b>	<b>24</b>
5.1 Bill of materials . . . . .	24
5.2 Mécanique du projet . . . . .	24
5.3 Placement des composants . . . . .	24
5.4 Mécanique du PCB . . . . .	24
5.5 Routage . . . . .	24
<b>6 Développement firmware</b>	<b>24</b>
<b>7 Validation du design</b>	<b>24</b>
7.1 Liste de matériel . . . . .	24
7.2 Contrôle des alimentations . . . . .	24
7.3 Communication UART . . . . .	24
7.4 Communication SPI, carte SD . . . . .	24
<b>8 Caractéristiques du produit fini</b>	<b>24</b>
<b>9 Conclusion</b>	<b>25</b>
<b>10 Bibliographie</b>	<b>26</b>
<b>11 Annexes</b>	<b>27</b>

## Acronymes

**MCU** microcontrôleur.

**PCB** circuit imprimé.

**IMU** centrale inertielle.

**RF** radio-fréquence.

**GPS** global Positioning System.

**GNSS** global navigation satellite systems.

## Glossaire

**Centrale inertielle** Instrument utilisé en navigation, capable d'intégrer les mouvements d'un mobile (accélération et vitesse angulaire) pour estimer son orientation (angles de roulis, de tangage et de cap), sa vitesse linéaire et sa position.

**timestamp** Enregistrement de l'heure et/ou la date d'un événement.

**GPS** Le système de positionnement global (GPS) est un service public américain qui fournit aux utilisateurs des services de positionnement, de navigation et de synchronisation (PNT). Ce système se compose de trois segments : le segment spatial, le segment de contrôle et le segment utilisateur. L'U.S. Space Force développe, entretient et exploite les segments spatial et de contrôle.

**GNSS** Le système mondial de navigation par satellite (GNSS) est un terme général décrivant toute constellation de satellites qui fournit des services de positionnement, de navigation et de synchronisation (PNT) à l'échelle mondiale ou régionale.

**PIC32** Famille de microcontrôleur 32-bits de Microchip.

**FTDI** Composant Future Technology Devices International. Ici sert de convertisseur USB to UART.

**harmony** Configurateur graphique / générateur de code pour les microcontrôleurs de Microchip.

**datasheet** Document du fabricant fournissant les spécifications d'un produit.



# Boîte noire miniaturisée 2023, 1942B

## 1 Cahier des charges

### 1.1 Introduction

Ce projet vise à stocker les données de mesures et de localisation d'un avion en utilisant une centrale inertuelle et un GPS/GNSS. En combinant ces technologies, nous pouvons enregistrer des informations précises sur les caractéristiques du vol et la trajectoire de l'avion. En cas d'accident, ces enregistrements permettent de déterminer les causes potentielles. En somme, ce système de collecte et de stockage de données fournit une compréhension approfondie des vols et des données essentielles. Le cahier des charges détaillé est disponible en annexe.

### 1.2 Aperçu

- Sauvegarde des données inertielles chaque 500ms par défaut.
- Sauvegarde des données de localisation chaque 5'000ms par défaut.
- Possibilité de configurer les temps de sauvegarde.
- Résistance aux chocs.
- Bonne autonomie / Low power.
- [Global Positioning System. \(GPS\)](#)
- [Global navigation satellite systems. \(GNSS\)](#).
- [Timestamp](#) par satellite.
- [Centrale inertuelle](#) :
  - Accéléromètre 3-axes.
  - Gyroscope 3-axes.
- Charge, lecture et config. par USB-C.

### 1.3 Tâches à réaliser

Développement et intégration d'un PCB avec capteurs et logging sur carte SD dans un boîtier compact.

- Développement schématique
  - Fonctionnement MCU.
  - Périphériques de mesures et de sauvegarde / Bus de communication.
  - Gestion batterie
- Routage pour intégration dans boîtier résistant aux chocs.
- Programmation mesure et sauvegarde des données.
  - Configuration MCU.
  - Configuration du périphérique de mesure pour [centrale inertuelle. \(IMU\)](#).
  - Configuration du périphérique de sauvegarde (Carte SD).
  - Configuration du périphérique de localisation [GPS/GNSS](#).
  - Configuration et communication avec l'interface.
  - Communication et traitement des données mesurées.

## 1.4 Schéma de principe



FIGURE 1 – Schéma de principe  
Source: Auteur

Ce système électronique de mini boîte noire pour avion, serait capable d'enregistrer des informations récentes sur les données inertielles et la position d'un vol, dans une mémoire non volatile (carte microSD). Le dispositif, abrité dans un boîtier plastique pour assurer une réception optimale des données GPS et une installation compacte. Celui-ci fournirait des données via un port USB pour l'extraction des mesures sur la carte microSD ou pour configurer des paramètres tels que les intervalles de mesures. Les mesures sur les trois axes d'accélération et de vitesse angulaire seraient recueillies par défaut toutes les 500 ms et les données de position GPS toutes les 5000 ms. Des intervalles d'enregistrement plus longs sont envisageables pour optimiser la durée de vie de la carte SD, selon la taille et l'organisation des données. Le dispositif sauvegarderait les données des 15 dernières minutes de vol (ou plus) dans un fichier CSV pour traitement ultérieur, selon le principe FIFO. L'objectif principal du prototype est de privilégier la compacité pour minimiser son encombrement à bord de l'avion.

## 1.5 Planification

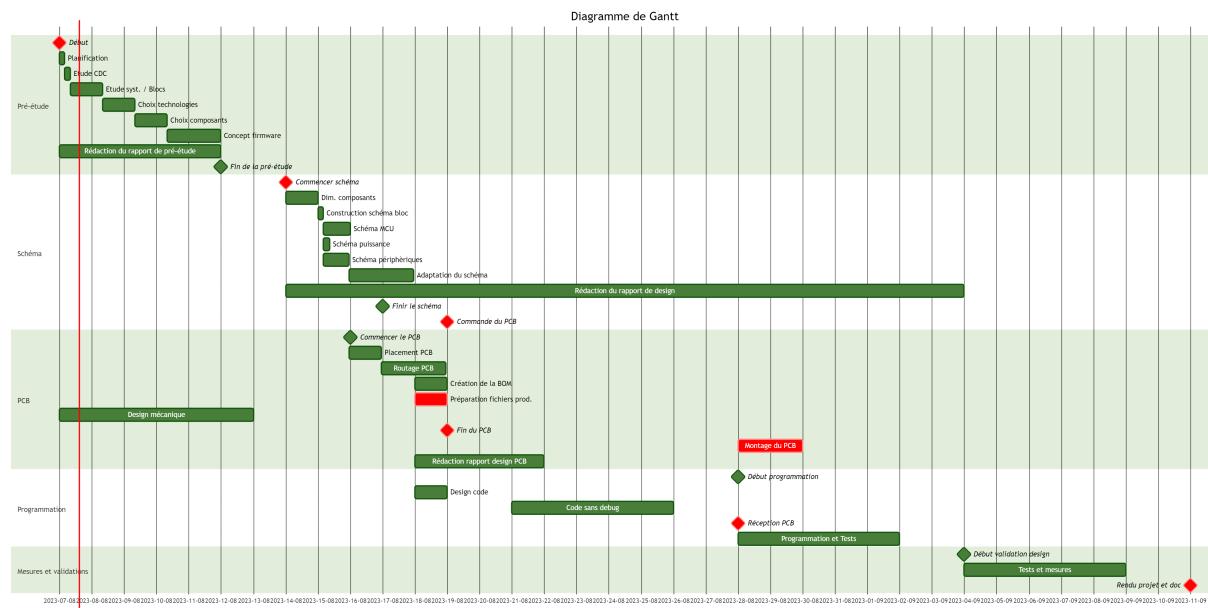


FIGURE 2 – Planification - Diagramme de gantt

Source: Auteur

No jour de travail	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36					
Date	lundi 7. aout 2023	mardi 8. aout 2023	mercredi 9. aout 2023	vendredi 11. aout 2023	samedi 12. aout 2023	dimanche 13. aout 2023	lundi 14. aout 2023	mardi 15. aout 2023	mercredi 16. aout 2023	vendredi 18. aout 2023	dimanche 19. aout 2023	lundi 21. aout 2023	mardi 22. aout 2023	mercredi 23. aout 2023	vendredi 25. aout 2023	samedi 26. aout 2023	dimanche 27. aout 2023	lundi 28. aout 2023	mardi 29. aout 2023	mercredi 30. aout 2023	vendredi 1. septembre 2023	samedi 2. septembre 2023	dimanche 3. septembre 2023	lundi 4. septembre 2023	mardi 5. septembre 2023	mercredi 6. septembre 2023	vendredi 8. septembre 2023	samedi 9. septembre 2023	dimanche 10. septembre 2023	lundi 11. septembre 2023											
Cahier des charges																																									
Pré-étude																																									
Dimensionnement + design + schéma																																									
Design de parties mécaniques																																									
Montage PCB																																									
Design software																																									
Réalisation des softwares																																									
Mise en service et tests																																									
Caractéristiques et mesures																																									
Préparation et présentation																																									
Rédaction rapport																																									
Finalisation/corrections/documentation																																									

FIGURE 3 – Planification théorique

## 1.6 Livrable

- Les fichiers sources de CAO électronique des PCB réalisés
- Tout le nécessaire à fabriquer un exemplaire hardware de chaque :
  - fichiers de fabrication (GERBER) / liste de pièces avec références pour commande / implantation
  - Prototype fonctionnel
  - Modifications / dessins mécaniques, etc
  - Les fichiers sources de programmation microcontrôleur (.c / .h)
  - Tout le nécessaire pour programmer les microcontrôleurs (logiciel ou fichier .hex)
  - Un calcul / estimation des coûts
  - Un rapport contenant les calculs - dimensionnement de composants - structogramme, etc.

## 2 Pré-étude

### 2.1 Fonctionnement du système

Le microcontrôleur interagît avec 4 périphériques principaux : Avec le **GNSS**, il partage une communication qui lui permet d'obtenir les informations de localisation par le biais de plusieurs systèmes de satellites. Il y a ensuite, la centrale inertuelle qui lui donne accès de une multitude de mesures sur 9 axes, or, ici les mesures gyroscopiques et d'accélération sont exploitées. La carte SD, permet quant-à-elle, de stocker toutes ces données pour avoir minimum les information des 15 dernières minutes de vol. Le dernier périphérique principal **FTDI**, permet d'avoir une interface avec un ordinateur via connexion USB-C.

### 2.2 Schéma bloc



FIGURE 4 – Schéma bloc

Source: Auteur

### 2.3 Description des blocs

Bloc	Description
<b>GNSS.</b>	Récepteur <b>Radio-fréquence, (RF)</b> avec antenne interne/externe et communication UART.
<b>Microcontrôleur. (MCU).</b>	Microcontrôleur PIC32, intelligence du système, basse consommation.
<b>IMU.</b>	Centrale inertuelle, accélération, gyroscope...
Carte SD	Stockage des données de vol.
<b>FTDI.</b>	Convertit la communication USB en série.
Régulateurs.	Le régulateur de charge gère la charge de l'accu. et un régulateur 3.3V le suit.
Batterie.	La batterie est un accu que l'on peut charger par USB et permet une bonne autonomie.

## 2.4 Choix des composants et technologies

L'objectif de la pré-étude consiste en grande partie à sélectionner méthodiquement les technologies et les composants du projet. Cette partie du travail est essentielle et critique.

### 2.4.1 Microcontrôleur

Le microcontrôleur nécessite au minimum les périphériques suivants :

**2 UART** **1 SPI** **1 I2C**

Il est préférable que le **MCU** dispose de différentes configurations de gestion de puissance, notamment des modes d'économie d'énergie, afin d'avoir une maîtrise de la consommation et de permettre une meilleure autonomie. Enfin, le standard de l'école veut que les familles de microcontrôleurs PIC32 (Microchip) sont préférées.

	Memory Flash/SRAM	Automotive	Connectivity	Functional Safety	Graphics	Motor Control	Security	Ultra-Low Power	Touch
<b>PIC32MZ EF FPU</b> MIPS32® M-Class, 252 MHz	512-2048 KB/ 128-512 KB	●	●	●	●		●		●
<b>PIC32MZ DA</b> MIPS32 microAptiv™, 200 MHz	1024-2048 KB/ 256-640 KB		●	●	●		●		●
<b>PIC32CX SG</b> Arm® Cortex®-M4F, 120 MHz	1024 KB/ 256 KB	●	●	●	●	●	●		●
<b>PIC32MK</b> MIPS32 microAptiv, 120 MHz	256-1024 KB/ 128-256 KB	●	●	●	●	●			●
<b>PIC32MX 3/4</b> MIPS32 M4K™, 80-120 MHz	32-512 KB/ 8-128 KB		●	●					
<b>PIC32MX 5/6/7</b> MIPS32 MM, 80 MHz	64-512 KB/ 16-128 KB	●	●						
<b>PIC32MX 1/2 XLP</b> MIPS32 M4K, 72 MHz	128-256 KB/ 32-64 KB		●	●				●	
<b>PIC32MX 1/2/5</b> MIPS32 M4K, 50 MHz	16-512 KB/ 4-64 KB		●	●					
<b>PIC32CM JH</b> Arm® Cortex®-M0+, 48 MHz	128-512 KB/ 16-64 KB	●	●	●		●	●		●
<b>PIC32CM Lx</b> Arm® Cortex®-M23, 48 MHz	256-512 KB/ 32-16 KB		●				●	●	●
<b>PIC32CM MC</b> Arm® Cortex®-M0+, 48 MHz	64-128 KB/ 8-16 KB			●		●			
<b>PIC32MM</b> MIPS32 microAptiv UC, 25 MHz	16-256 KB/ 4-32 KB	●					●		

FIGURE 5 – Familles PIC32

Source: [Microchip, familles](#)

Sur la figure 5 le **MCU** sélectionné appartient à la gamme MX (Baseline performance-mémoire) et à la famille XLP qui offre notamment la fonctionnalité "Ultra low power" qui est celle qui nous intéresse.

PIC32MX274F256D



FIGURE 6 – Caractéristiques du PIC32 choisi

Source: Auteur

## 2.4.2 Centrale inertielle

Pour la centrale inertielle, il existe un composant avec lequel j'ai déjà acquis une certaine expérience et eu l'occasion d'utiliser et de créer des librairies pour le firmware en C. Celui-ci est performant et très utilisé dans l'industrie. Il est hautement configurable et a le grand avantage de calculer déjà une fusion de capteurs ainsi qu'une compensation de la dérive par les mesures de température. Cela permet notamment d'accéder à des données plus poussées, telles que les quaternions et les angles d'Euler. Il s'agit du **BNO055** de BOSCH.

### Caractéristiques importantes :

Résolution gyroscope	:	16	[bits]
Résolution accéléromètre	:	14	[bits]
Résolution magnétomètre	:	~0.3	[ $\mu$ T]
$I_{DD}$	:	12.3	[mA]
Dérive de température	:	$\pm 0.03$	[%/K]
Dérive accéléromètre	:	0.2	[%/V]
Dérive gyroscope	:	<0.4	[%/V]

Afin de simplifier l'implémentation de ce composant dans le projet, sachant qu'il s'agit d'un boîtier de composant difficile à souder ou à mettre au four, il est possible d'utiliser la carte miniature d'Adafruit, cette carte comprend tous les composants passifs requis. Cela facilite ainsi son montage sur le PCB.



FIGURE 7 – Carte d'extension, centrale inertielle  
Source: [Digikey, 4646](#)

### Données disponibles :

Température	
Vecteur gravité	XYZ
Orientation compensées, quaternion	WXYZ
Orientation compensée, angle de Euler	HPR
Données gyroscopiques	XYZ
Intensité du champ magnétique	XYZ
Accélération	XYZ

TABLE 1 – Liste des données accessibles

### 2.4.3 GPS / GNSS

Pour le **GPS/GNSS**, différents critères entrent en jeu dans le cadre de ce projet : le prix, la facilité d'implémentation, la complexité (par complexité, nous entendons le nombre de fonctionnalités), la consommation et la performance.

Il existe un très grand nombre de récepteurs **RF** pour la navigation. Parmi les plus utilisés dans l'industrie, dont l'implémentation est la plus simple et la documentation la plus complète, il y a plusieurs gammes chez le fabricant **ublox**. Deux composants ont principalement été pris en considération : Le **CAM-M8C-0** (BeiDou, GLONASS, GNSS, GPS, QZSS) avec une antenne omnidirectionnelle interne au composant et différents modes de puissance, ainsi que le **MAX-M10M-00B** (BeiDou, Galileo, GLONASS, GNSS, GPS) sans antenne interne mais avec une consommation de base plus faible.

**MAX-M10M-00B**      **CAM-M8C-0**

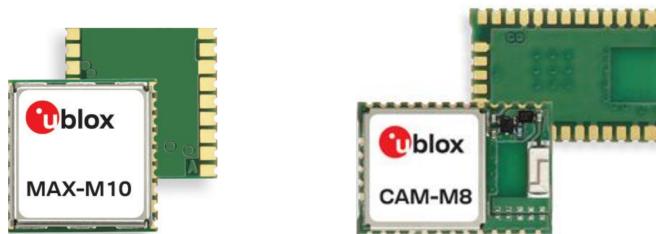


FIGURE 8 – Illustration des deux GNSS

Source: [Digikey, MAX-M10 et CAM-M8C](#)

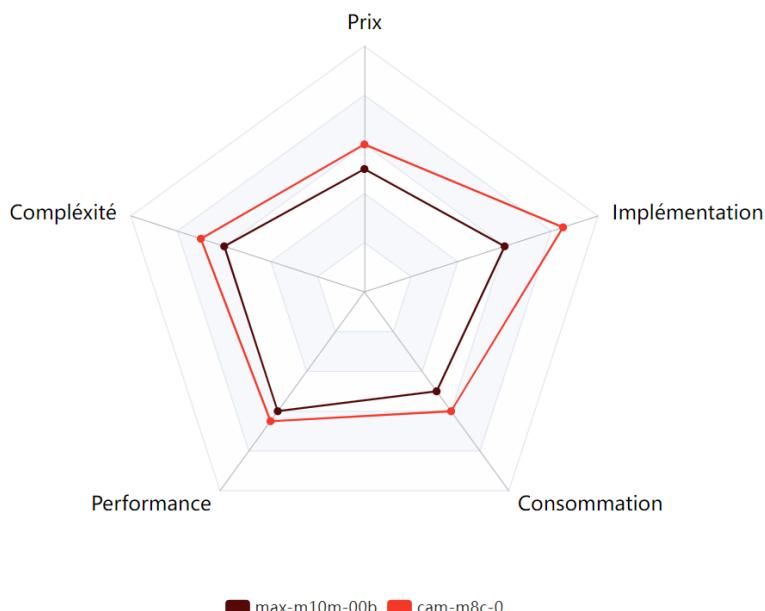


FIGURE 9 – Comparaison GNSS

Source: Auteur

Malgré les différents avantages que présente le MAX-M10M-00B sur la figure 9, le choix s'est porté sur CAM-M8C-0 grâce à sa facilité d'implémentation et à la garantie du fabricant sur son antenne omnidirectionnelle de qualité.

#### 2.4.4 Carte SD

Les pilotes de carte SD disponibles dans [harmony](#) ne permettent pas une gestion des capacités de stockage trop importantes. Cela signifie, que nous ne pouvons pas avoir des cartes SD de trop grandes capacités, c'est pour cela que nous allons baser nos dimensionnements sur une carte de **256MB**.



FIGURE 10 – Illustration carte SD

Source: <https://www.oempcworld.com/OEMPCworld-com/017045.html>

**Estimation de la capacité** Admettons les paramètres suivants où  $S$  représente une taille de données et  $T$  un temps :

$S_{SD}$	256	[KB]
$S_{gyro}$	16	[Bytes]
$S_{accel}$	16	[Bytes]
$S_{gnss}$	~ 100	[Bytes]
$T_{inertiel}$	0.5	[s]
$T_{gnss}$	5	[s]
$T_{mesMin}$	900	[s]

Nous pouvons déduire la taille mémoire que prendra 5 secondes d'enregistrement avec les paramètres par défaut du système :

$$S_{single} = \frac{T_{gnss}}{T_{inertiel}}(S_{gyro} + S_{accel}) + S_{gnss} = \frac{5}{0.5}(16 + 16) + 100 = 420 \text{ [Bytes]}$$

Nous pouvons enfin calculer à partir de cela, la taille mémoire que prendra 15 minutes d'enregistrement :

$$S_{mesures} = \frac{S_{single}}{T_{gnss}} * T_{mesMin} = \frac{420}{5} * 900 = 75'600 \text{ [Bytes]} = 75.6 \text{ [KB]}$$

Nous pouvons déduire avec cette estimation qu'une carte SD de 256MB est largement suffisante et permet de mesurer jusqu'à un temps calculable de cette façon :

$$T_{mesures} = \frac{S_{SD} * T_{gnss}}{S_{single}} = \frac{256'000 * 5}{420} = \sim 508 \text{ Minutes} = \sim 8.5 \text{ Heures}$$

Nous estimons donc, que les données de vol des 8.5 dernières heures sont enregistrées dans la carte SD.

## 2.5 Batterie, autonomie, charge et régulation

Afin de dimensionner une batterie pour le projet, il faut considérer les différentes consommation :

Liste des consommations principales

Microcontrôleur	24	[mA]	Typ.
Carte-SD	100	[mA]	Max.
Carte-SD	60	[mA]	Moyenne
IMU	12.3	[mA]	Typ.
GNSS	71	[mA]	Max.
GNSS	29	[mA]	Typ.
Totale max	<u>207.3</u>	[mA]	Max.
Totale moyennes	<u>125.3</u>	[mA]	Moyenne

TABLE 2 – Tableau des consommations de courant

En examinant les consommations typiques et moyennes du tableau 2 à régime constant, et en visant une autonomie de **10 heures**, nous aurions besoin d'une batterie d'une capacité de **1253 mAh**.

**Technologies** Concernant la technologie de la batterie, notre objectif, dans un souci d'ergonomie, est de permettre son chargement via un connecteur USB. La technologie offrant actuellement la meilleure densité d'énergie est le **Lithium Ion**<sup>1</sup>. Elle présente cependant des inconvénients, énumérés dans le tableau 3.

Avantages	Inconvénient
Haute densité d'énergie	Risque d'éclatement
Poids léger	Risque d'enflammement avec l'eau
Haute durée de vie	Sensible à la température
Charge rapide	Décharge complète altérante

TABLE 3 – Tableau avantages/inconvénient LI-ION

Les inconvénients listés dans le tableau 3 posent un risque pour l'intégrité des données de la carte-SD. C'est pourquoi il est essentiel de bien dimensionner le boîtier ainsi que l'intégration du circuit et de la batterie pour pallier ces dangers. Malgré les risques du Li-Ion, il s'agit d'une technologie très utilisée, y compris dans des domaines sensibles.

1. <https://www.epectec.com/batteries/cell-comparison.html>

**Choix de la batterie** La batterie doit avoir une capacité d'au moins **1260 mAh** et être suffisamment compacte pour être intégrée dans un boîtier de petite taille. Une batterie répond à ces critères : il s'agit de la **PICPAL36** de chez Farnell<sup>2</sup>. Les dimensions et caractéristiques de cette batterie sont visibles sur les figures 11 et 12.

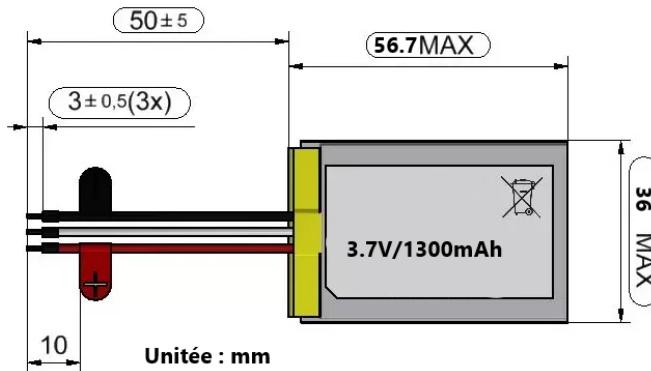


FIGURE 11 – Dimension de la batterie

Source: Auteur

<input type="checkbox"/> <b>Battery Voltage:</b>	3.7V	<input type="checkbox"/> <b>External Width:</b>	6.5mm
<input type="checkbox"/> <b>Battery Technology:</b>	Lithium Ion	<input type="checkbox"/> <b>Pack Quantity:</b>	1
<input type="checkbox"/> <b>Battery Capacity:</b>	1.3Ah	<input type="checkbox"/> <b>External Depth:</b>	35mm
<input type="checkbox"/> <b>Battery Terminals:</b>	Connector	<input type="checkbox"/> <b>Weight:</b>	27g
<input type="checkbox"/> <b>Battery Size Code:</b>	-	<input type="checkbox"/> <b>Battery IEC Code:</b>	-
<input type="checkbox"/> <b>External Diameter:</b>	-	<input type="checkbox"/> <b>Battery NEDA Code:</b>	-
<input type="checkbox"/> <b>External Height:</b>	55.7mm	<input type="checkbox"/> <b>Product Range:</b>	-

FIGURE 12 – Caractéristiques de la batterie

**Régulateur de charge** Le choix du régulateur de charge s'est orienté vers un composant déjà utilisé par l'école, dont les caractéristiques et le montage sont bien connus. C'est un composant performant et fréquemment employé. De plus, il est disponible dans le stock de composants de l'école. Il s'agit du **MCP73871T-2CCI/ML**. Il est configurable et permet la charge des batteries Li-ion et Li-po.

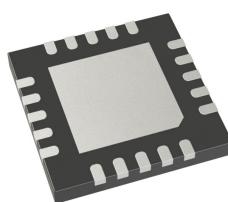


FIGURE 13 – MCP73871T-2CCI/ML

Source: [Digikey IC BATT CHG LI-ION 1CELL 20QFN](#)

2. Distributeur de composants électroniques. [Lien de la batterie](#).

## 2.6 Systèmes d'économie d'énergies

Au-delà des 10 heures d'autonomie estimées, il est envisageable, pour maximiser le temps de logging, d'adopter des mécanismes d'économie d'énergie. Ces mécanismes pourraient être :

- **Modes de puissance du PIC32** Le microcontrôleur peut basculer entre différents modes de puissance où il réduit à la fois sa consommation et ses fonctionnalités. On pourrait envisager de passer en mode d'énergie restreinte entre les périodes d'enregistrement ou de limiter l'utilisation de certains périphériques.
- **Modes de puissance du GNSS** À l'instar du **MCU**, le composant de navigation dispose de plusieurs modes de fonctionnement qui permettent d'économiser de l'énergie. Un mode particulièrement utile permet de limiter sa cadence de données à 1Hz, étant donné qu'il n'est pas nécessaire qu'il fournisse des informations de localisation aussi fréquemment.
- **Alimentation du GNSS** On pourrait également envisager de ne pas alimenter le **GNSS** lorsqu'il n'est pas en service, afin de supprimer complètement sa consommation entre chaque mesure.
- **Modes de puissance IMU** La centrale inertuelle dispose également de différents modes. Ces modes permettent, par exemple, de limiter les mesures à certains axes ou de la mettre en veille. Elle se met d'ailleurs automatiquement en veille après un certain temps entre les mesures. Ces modes peuvent être utilisés efficacement.
  - **Alimentation de la carte SD** Il est envisageable d'éteindre entièrement la carte SD entre les écritures. Cependant, étant donné que les écritures seront fréquentes, l'intérêt de cette démarche est limité.
- **Mise en veille automatique du système** Si aucun mouvement n'est détecté après un certain délai, il est envisageable que le système s'éteigne entièrement.

## 2.7 Diagramme des états du système

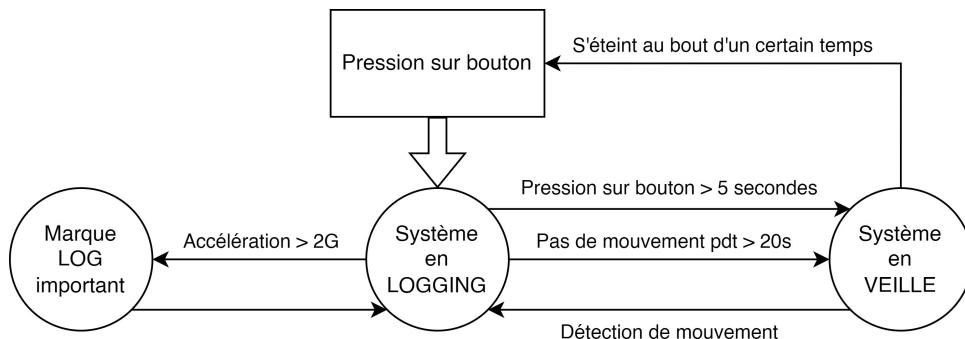


FIGURE 14 – Diagrammes des états (Sans USB)  
Source: Auteur

Sur la figure 14 les états de communication par le port USB ne sont pas représentés, il s'agit des modes principaux.

## 2.8 Estimation des coûts

À partir des composants et des technologies définis lors de la pré-étude, nous pouvons estimer le coût du projet. Cette estimation est présentée dans le tableau 4.

<u>Estimation des coût</u>	
Élément	Prix estimé
circuit imprimé. (PCB)	60.-
Batterie	36.-
IMU	26.-
GNSS	24.-
Boîtier	20.-
Composants divers	15.-
MCU	6.-
Régulateurs	6.-
USB & FTDI	3.-
Total	<u>196.-</u>

TABLE 4

Cette estimation est relativement élevée. Il est à noter que le coût du **PCB** pourrait être réduit en cas de commande groupée dans un panel<sup>3</sup>. Par ailleurs, les prix du boîtier et des composants semblent également être assez élevés.

## 3 Conclusion de la Pré-étude

**Fonctionnement du système :** La conception du système semble possible, le microcontrôleur s'interface avec quatre périphériques essentiels : un **GNSS** pour la localisation, une centrale inertielle pour des mesures sur 9 axes (priorisant les mesures gyroscopiques et d'accélération), une carte SD pour le stockage (conservant au moins 15 minutes de données de vol), et un périphérique **FTDI** pour l'interface avec un ordinateur via USB-C.

**Choix des composants et technologies :**

- *Microcontrôleur* : Requiert au moins 2 UART, 1 SPI, et 1 I2C. Le PIC32MX274F256D est privilégié car il offre divers modes d'économie d'énergie.
- *Centrale inertielle* : Le BNO055 de BOSCH a été choisi, car il offre une variété de mesures avancées et une facilité d'implémentation grâce à la carte Adafruit.
- *GPS/GNSS* : Le CAM-M8C-0 d'ublox a été retenu pour sa facilité d'implémentation et son antenne interne omnidirectionnelle.
- *Carte SD* : Une capacité de 256MB a été choisie à cause des limites du pilote. Une telle carte permettrait d'enregistrer jusqu'à 8,5 heures de données de vol.
- *Batterie, charge, et régulation* : Une capacité de 1253 mAh est nécessaire pour atteindre l'autonomie souhaitée de 10 heures. Une batterie compact de 1300 mAh a été retenue.

**Estimation des coûts :** L'estimation totale pour le projet s'élève à environ 196 CHF. Néanmoins, il existe des possibilités d'économie, en particulier par une commande groupée du **PCB**.

En conclusion, cette pré-étude a décortiqué le projet de façon méthodique pour sélectionner les composants et technologies appropriés. Elle a également fourni des estimations de coûts et identifié des mécanismes d'économie d'énergie potentiels. La prochaine étape consistera à entamer la phase de conception en utilisant ces informations utiles.

3. Regroupement de plusieurs circuits imprimés en un seul, pour diminuer le coût.

## 4 Développement du schéma électronique

Dans cette section, nous décrirons la phase principale du développement ainsi que la démarche suivie pour élaborer le schéma électronique du projet.

### 4.1 Blocs développés

Pour faciliter le développement et la lecture du schéma, il est judicieux de diviser le système en plusieurs blocs. Une structure a ainsi été définie, divisant le circuit en trois blocs principaux : **Microcontrôleur 4.2** (Intelligence du système, connexion du programmeur et LED de vie.), **Périphériques 4.3** (GNSS, IMU, FTDI, connecteur USB, Carte SD.) et **Alimentations 4.4** (Connecteur batterie, gestion de charge, régulateurs de tension et système ON/OFF.).

Nous pouvons sur la figure 15 observer les différentes interactions entre les blocs, elles sont par la suite décrites dans le tableau 5.

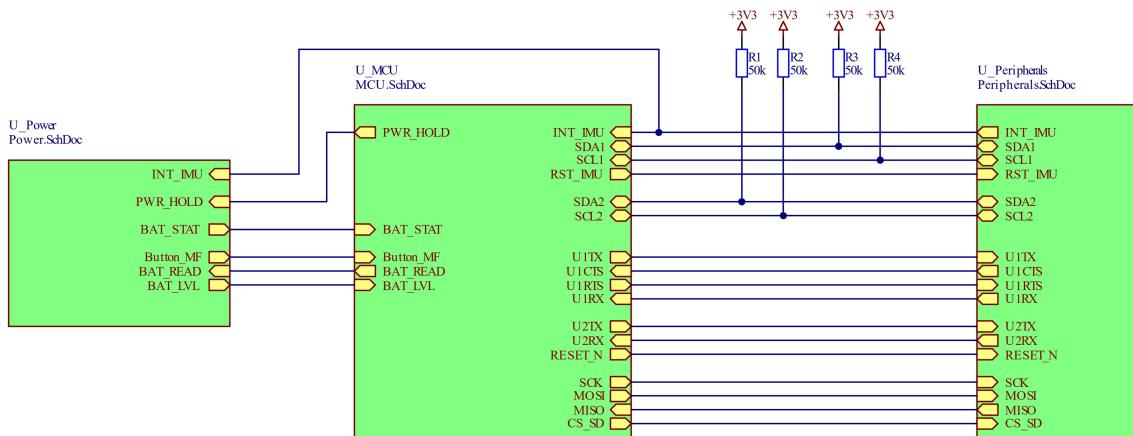


FIGURE 15 – Blocs du système

Source: Auteur

Tableau des interactions entre les blocs

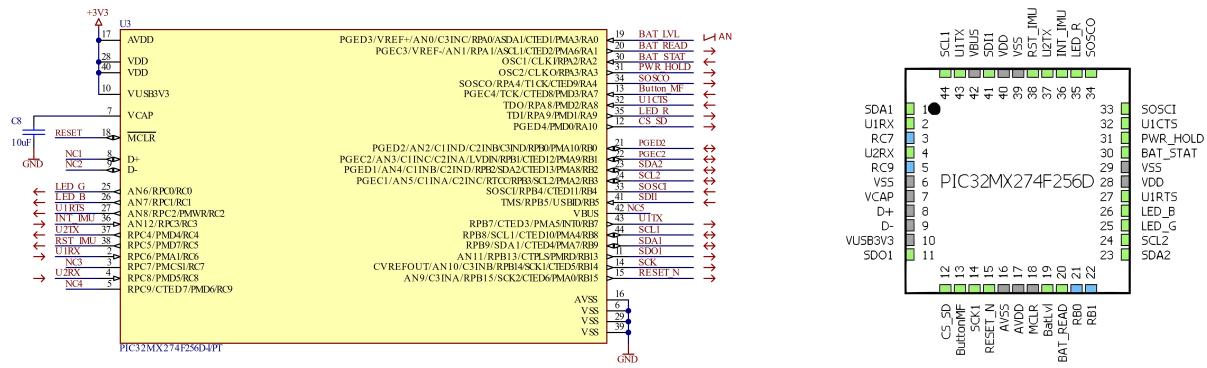
Connexion/s	Description
INT_IMU	Interruption de la centrale inertuelle, informe le <b>MCU</b> et peut allumer le système.
RST_IMU	Permet de réinitialiser la <b>IMU</b> .
PWR_HOLD	Le <b>MCU</b> peut se maintenir alimenté par cette connexion.
BAT_STAT	Fournit le statut de la batterie au <b>MCU</b> .
Button_MF	Fournit le niveau logique du bouton au <b>MCU</b> .
BAT_READ	Le <b>MCU</b> peut activer la lecture de la tension de la batterie.
BAT_LVL	Lecture analogique de la tension de batterie.
SDA1, SCL1	Communication I2C avec l' <b>IMU</b> .
SDA2, SCL2	Communication I2C optionnelle avec le <b>GNSS</b> .
U1TX/RX...	Communication UART avec le <b>FTDI</b> pour l'USB.
U2TX/RX	Communication UART avec le <b>GNSS</b> .
RESET_N	Permet de réinitialiser le <b>GNSS</b> .
SCK, MOSI...	Communication SPI avec la carte SD.
R1,2,3,4	Résistances de PULL-UP pour communication I2C.

TABLE 5 – Description des connexions

## 4.2 Microcontrôleur

### 4.2.1 Connexion

Pour utiliser le microcontrôleur, il est nécessaire de définir ses entrées/sorties en se référant à son [datasheet](#). Ce dernier permet de connaître les connexions dédiées à certains bus de communication ou à des entrées analogiques.



(a) Connexions du microcontrôleur

(b) Config. [harmony](#).

FIGURE 16 – Configuration des PINs du microcontrôleur

Source: Auteur

Pour valider les connexions de la figure 16a, une vérification a été réalisée à l'aide du configurateur graphique [harmony](#), comme le montre la figure 16b. Cette étape a confirmé la possibilité d'attribuer certaines fonctions à des PINs spécifiques. Les PINs non utilisés du microcontrôleur sont redirigées vers un connecteur. Dans le contexte du prototype, cela permet de les exploiter ailleurs sur la carte.

### 4.2.2 Programmateur et reset

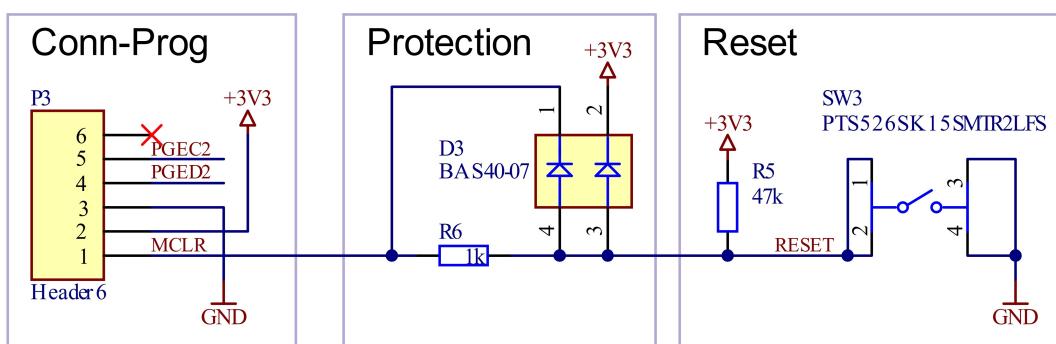


FIGURE 17 – Schéma programmateur et reset

Source: Auteur

Sur la figure 17, nous pouvons observer le connecteur de programmation **P3**. La connexion **MCLR** (Master Clear), qui permet de réinitialiser le **MCU** lors de sa programmation, est suivie d'un circuit de protection du **MCU** et d'un bouton pour permettre une réinitialisation manuelle.

#### 4.2.3 LED de vie

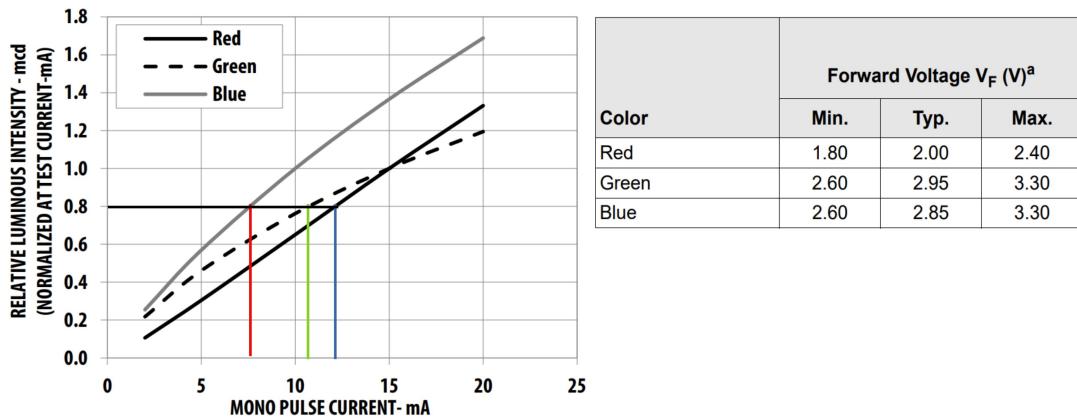


FIGURE 18 – Données de la LED  
Source: [datasheet ASMB-KTF0-0A306](#)

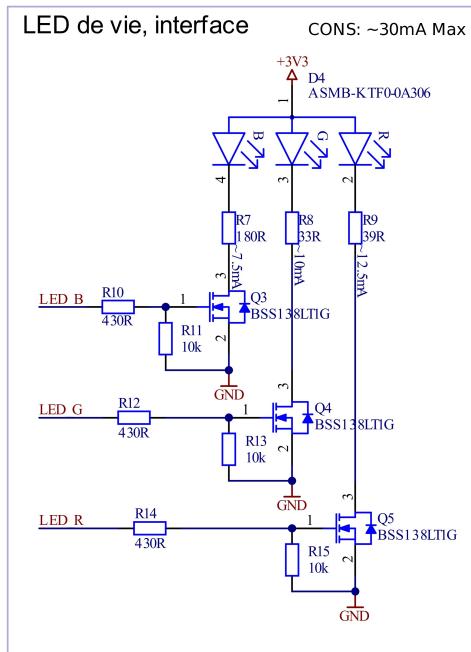


FIGURE 19 – Schéma de la LED de vie  
Source: Auteur

Les résistances de la figure 19 ont été dimensionnées en respectant les caractéristiques des LEDs pour chacune des couleurs (voir figure 18). Ces dernières éclairent à 80% de leur luminosité nominale dans un souci d'économie d'énergie, leur utilité étant réduite à ce stade de prototypage.

#### Exemple dimensionnement résistance LED bleue

$$R_{blue} = \frac{(Vcc - V_{blue})}{I_b} = \frac{(3.3 - 2.85)}{12 * 10^{-3}} = 37.5\Omega$$

## 4.3 Périphériques

Dans cette section, nous allons décrire les schématiques des périphériques du système.

### 4.3.1 Carte SD

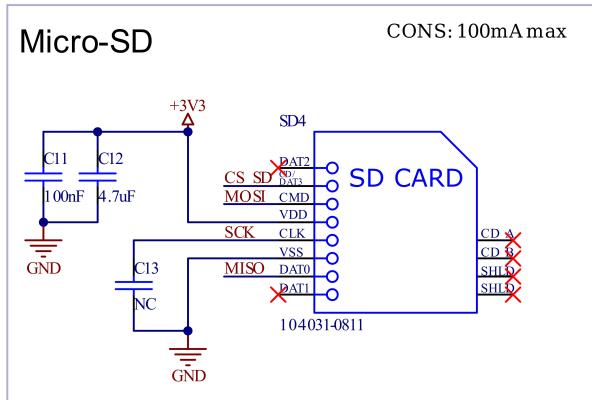


FIGURE 20 – Branchement carte SD

*Source: Auteur*

La carte SD de la figure 20 est branchée en respectant la connectique pour une communication SPI simple. De plus, un condensateur est connecté entre **SCK** et **GND**. Toutefois, ce condensateur ne sera pas monté sauf si un filtrage sur la ligne de clock s'avère nécessaire.

### 4.3.2 Centrale inertie

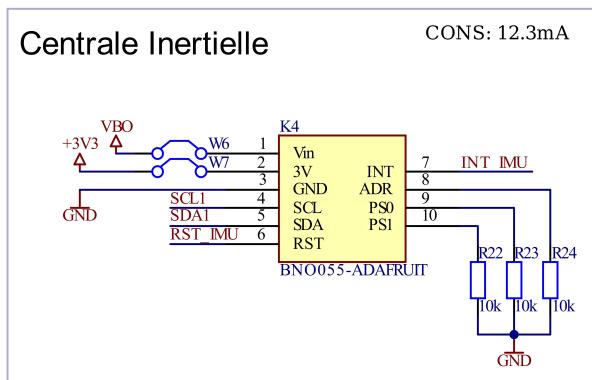


FIGURE 21 – Schéma centrale inertie

*Source: Auteur*

La centrale inertie, étant déjà montée sur une carte d'extension, ne nécessite pas de composants passifs additionnels. Trois résistances de PULL-DOWN sont présentes, respectivement sur **ADR**, **PS0** et **PS1**. Elles permettent de configurer l'adresse et l'interface de communication de l'**IMU**. De plus, un jumper<sup>4</sup> est présent sur les alimentations. Ceci permet de choisir si la centrale inertie est alimentée indépendamment du reste du système (pour pouvoir l'allumer séparément) ou avec la même alimentation que le reste du système.

4. Connecteur permettant de faire ou de rompre une connexion.

### 4.3.3 GNSS

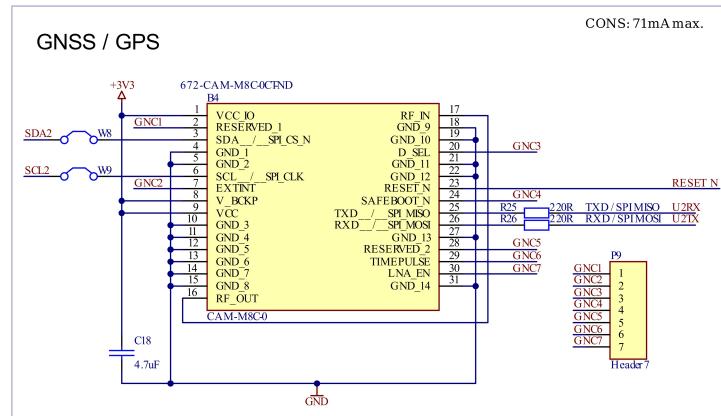


FIGURE 22 – Schéma du GNSS

*Source: Auteur*

Le **GNSS** est connecté selon un schéma présenté dans son [Hardware Integration Manual](#), à la section **2.2**. Ce schéma représente un design minimal pour recevoir des données via l’interface UART. Par rapport à la figure [22](#), les différences sont les suivantes : l’interface I2C est également disponible, la PIN **RESET\_N** est connectée au **MCU** et les PINS non-utilisées sont reliées à un connecteur afin de permettre leur utilisation éventuelle dans le cadre du prototype.

#### 4.3.4 USB - FTDI

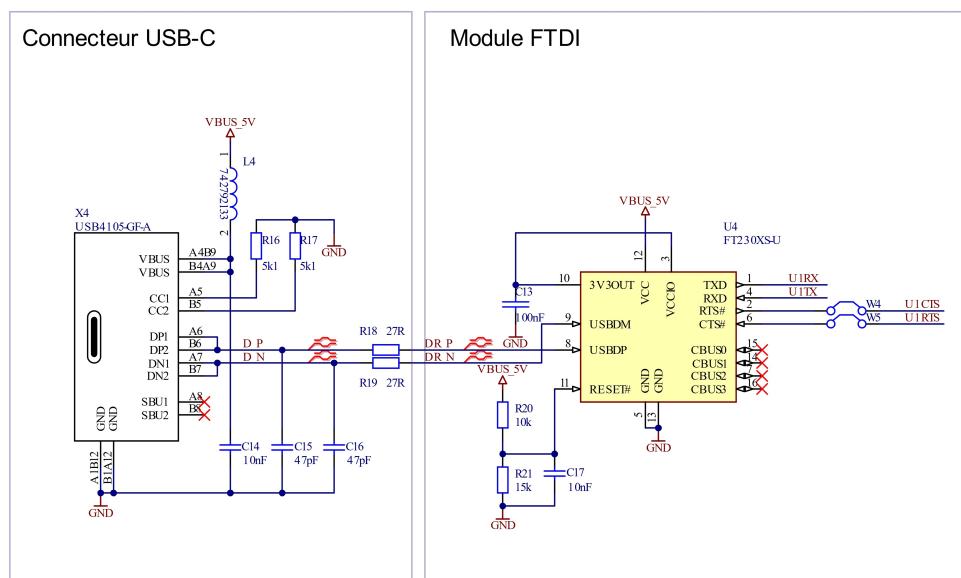


FIGURE 23 – Schéma connecteur USB et FTDI

*Source: Auteur*

Le schéma du connecteur USB-C est conçu pour présenter une résistance aux interférences. Son signal 5V est filtré par une *ferrite bead*. Les pistes différentielles de l'USB sont ensuite reliées au module [FTDI](#), qui est interfacé en UART et est alimenté par l'USB.

## 4.4 Alimentations

Dans cette section, nous allons décrire les fonctionnement et dimensionnements principaux du bloc alimentation comprenant les différents régulateurs et la batterie.

### 4.4.1 Chargeur de batterie

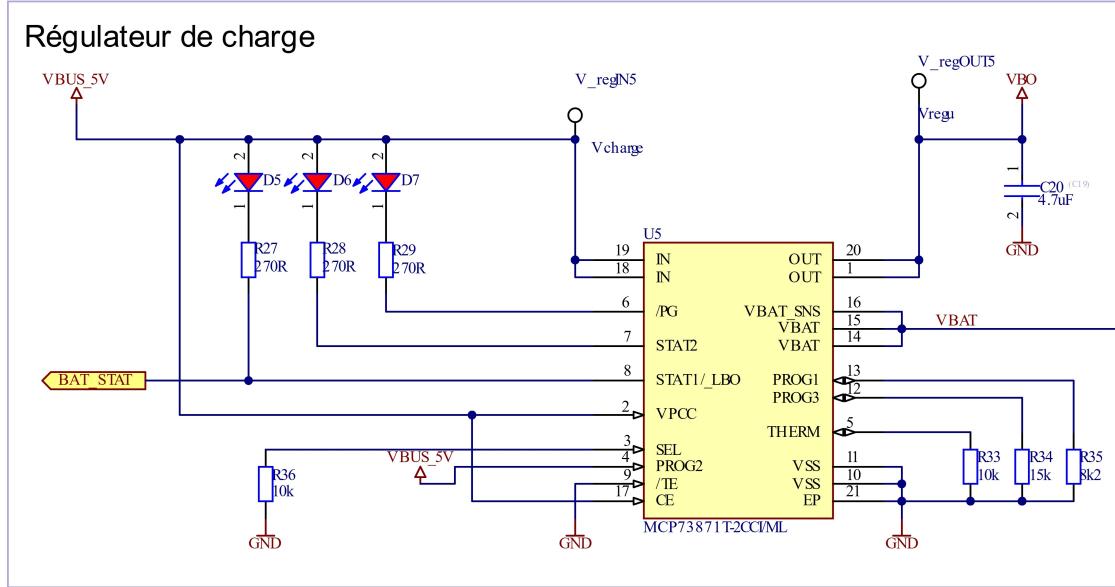


FIGURE 24 – Schéma chargeur de batterie

Source: Auteur

Le composant **MCP73871T-2CCI/ML** est configurable pour délivrer un courant de charge programmable à la batterie. Ce courant se sépare en deux phases : la charge normale et la charge de terminaison. Cette dernière est activée lorsque la batterie approche de sa capacité maximale. Pour déterminer ces courants, il convient de se baser sur la capacité de la batterie et de certains ratios standards associés.

Où :

$C = 1300mAh$  est la capacité de la batterie.

$k_{term} = 0.05$  est le ratio de fin de charge.

$k_{chg} = 0.1$  est le ratio de charge.

Le courant de terminais peut être calculé ainsi :

$$I_{term} = C * k_{term} = 1300 * 0.05 = 65 mA$$

Par conséquent, d'après la [datasheet](#) ([Equation 4-2](#)) la résistance de programmation de fin de charge vaudrait :

$$R_{prog3} = \frac{1000V}{I_{term}} = \frac{1000}{65 * 10^{-3}} = 15.38 k\Omega \Rightarrow 15 k\Omega$$

Nous pouvons de la même façon calculer la résistance de charge **Rprog1** :

$$R_{prog3} = \frac{1000V}{C * k_{chg}} = \frac{1000}{1300 * 0.1} = 7.69 k\Omega \Rightarrow 8,2 k\Omega$$

#### 4.4.2 Système d'enclenchement

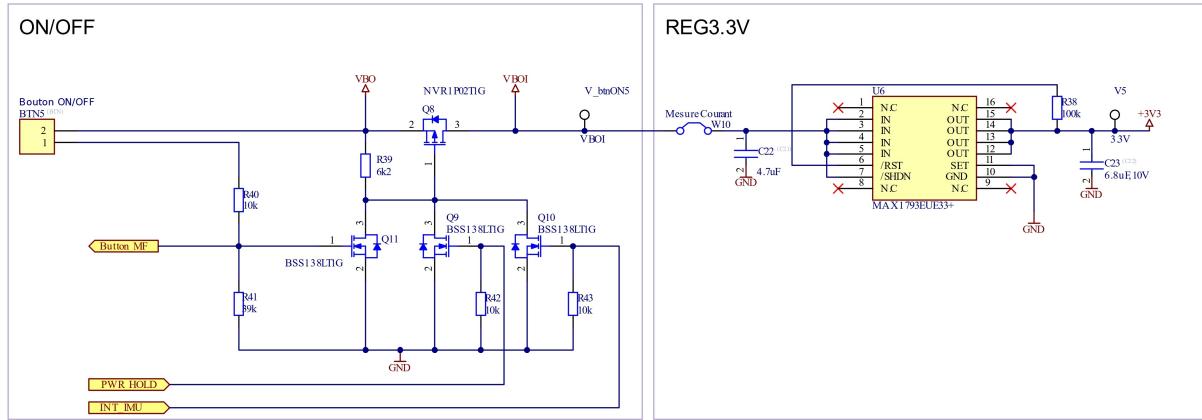


FIGURE 25 – Schéma allumage du système

Sur la figure 25, nous observons les différentes manières de connecter la tension régulée de la batterie au régulateur linéaire 3.3V. Quand cette connexion n'est pas réalisée, le système n'est pas alimenté, à l'exception de l'**IMU** qui peut être alimenté de manière autonome. Pour mettre en marche le système, trois options sont possibles :

- Le microcontrôleur peut maintenir le système alimenté avec **PWR\_HOLD**.
- L'**IMU** peut 'allumer le système avec son interruption **INT\_IMU**.
- Un bouton peut être pressé pour enclencher la carte **Button\_MF**.

#### 4.5 Conclusion et perspectives de l'étude

Tout au long de cette étude, nous avons décomposé et analysé le processus de développement du schéma électrique du projet. La démarche adoptée a mis l'accent sur la modularité, avec une division du système en trois blocs : *Microcontrôleur*, *Périphériques* et *Alimentations*. Cette approche a permis une meilleure lisibilité et une facilité d'adaptation quant au prototypage. Les différentes interactions et connexions entre les composants ont été détaillées.

Par la suite, le circuit imprimé sera développé, les composants placés et routés le tout dans un design petit et compact. La modularité du projet, permettra des tests empiriques sur les différents blocs du système, lors de la mise en service.

## 5 Développement du PCB

### 5.1 Bill of materials

### 5.2 Mécanique du projet

### 5.3 Placement des composants

### 5.4 Mécanique du PCB

### 5.5 Routage

## 6 Développement firmware

## 7 Validation du design

### 7.1 Liste de matériel

### 7.2 Contrôle des alimentations

Méthode de mesure

Mesures

### 7.3 Communication UART

Méthode de mesure

Mesures

### 7.4 Communication SPI, carte SD

Méthode de mesure

Mesures

## 8 Caractéristiques du produit fini

## 9 Conclusion

## 10 Bibliographie

### Références

- [1] P. Kordowski, Z. Jakielaszek, M. Nowakowski, and A. Panas, “Miniaturized flight data recorder for unmanned aerial vehicles and ultralight aircrafts,” in *2018 5th IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pp. 484–488, 2018.

## 11 Annexes