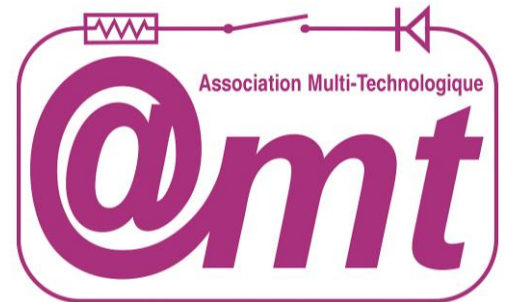


MF58 : EZSPACE Easy SPACE



C'Space 2025



Mouadh MASSOURI
Louis DAMAY STACHEL

AMT Lyon



12/08/2025

Table des matières

Liste des figures.....	3
Remerciements	4
I. Introduction	5
II. Outils logiciels.....	6
1. FreeCAD	6
2. Solidworks	6
3. OpenRocket.....	7
4. StabTraj.....	7
5. Arduino	8
III. Conception de la partie mécanique	9
1. Ogive	9
2. Tube principal	12
3. Bague de retenue du propulseur Pro-24.....	15
4. Module parachute avec trappe d'éjection.....	16
5. Ailerons	17
6. Module électronique	18
7. Propulseur Pro-24	20
IV. Conception de la partie électronique	21
1. L'unité de contrôle principale	21
a. Architecture matérielle.....	21
b. Architecture logicielle	22
i. Initialisation et lecture des paramètres	23
ii. Mode Configuration.....	23
iii. Passage en mode Vol.....	24
iv. Détection du départ et séquence de déclenchement	24
v. Réinitialisation après le vol.....	24
vi. Indications visuelles et auditives	24
2. L'unité de gestion des expériences	26
c. Architecture matérielle.....	26
d. Architecture logicielle	27

V.	Simulation et Intégration	29
1.	Simulation	29
a.	Stabilité.....	30
b.	Trajectoire.....	31
2.	Intégration.....	31
VI.	Analyse de l'expérience.....	33
1.	Contexte et objectif de l'expérience	33
2.	Problème rencontré.....	33
c.	Synchronisation GPS au lancement.....	33
d.	Cause probable.....	33
3.	Données obtenues après atterrissage.....	34
a.	Acquisition post-vol	34
b.	Précision en position fixe	34
4.	Analyse des performances du système embarqué.....	37
a.	Charge processeur.....	37
b.	Cadence de 100 ms.....	38
5.	Leçons apprises	38
6.	Pistes d'amélioration pour EZSPACE-X (C'Space 2026).....	38
a.	Améliorations matérielles.....	38
b.	Améliorations logicielles	38
c.	Améliorations opérationnelles	39
VII.	Conclusion et perspectives.....	40
	Bibliographie	42

Liste des figures

Figure 1 - Ogive ellipsoïdale : modèle 3D (FreeCAD) et pièce correspondante imprimée en 3D.....	11
Figure 2 - Ogive Haack : modèle 3D (SolidWorks) et pièce correspondante imprimée en 3D.....	13
Figure 3 - Transition Haack et tube du bas : modèle 3D (SolidWorks) et pièce correspondante imprimée en 3D.....	14
Figure 4 – Système de retenue du propulseur Pro-24 : modèle 3D (SolidWorks) et pièce correspondante imprimée en 3D.....	15
Figure 5 – Parachute et trappe d'éjection.....	17
Figure 6 - Aileron: modèle 3D (SolidWorks) et pièce correspondante imprimée en 3D. .	19
Figure 7 - Schéma en bloc de la carte de contrôle.....	22
Figure 8 - Interface web de configuration via WiFi.....	25
Figure 9 - Interface série et traces du système de contrôle.....	26
Figure 10 - Schéma en bloc de la carte d'expérience.....	28
Figure 11 - Vue de face de la minifusée sur OpenRocket.....	30
Figure 12 - Vue 3D en perspective de la minifusée sur OpenRocket.....	30
Figure 13 - Stabilité de la minifusée sur StabTraj.....	30
Figure 14 - Trajectoire de la minifusée sur StabTraj.....	31
Figure 15 - Position d'atterrissage de la minifusée.....	35
Figure 16 - Latitude en fonction de la Longitude.....	36
Figure 17 - Altitude en fonction du temps.....	36
Figure 18 - Trajectoire 3D (Longitude, Latitude, Altitude).....	37

Remerciements

Nous tenons à exprimer notre profonde gratitude à l'AMT et à ses membres (Rafik, Thierry, Lilian...) pour leur accueil chaleureux et leur soutien tout au long de l'année.

Nos remerciements s'adressent également aux membres de Planète Sciences, ainsi qu'aux contrôleurs des RCE et du C'Space, pour le temps qu'ils nous ont consacré et la richesse de leurs conseils.

Nous remercions aussi nos camarades de l'AMT, et plus particulièrement les membres de l'équipe ARCETHIE, pour leur travail, leur engagement et leur bonne humeur, qui ont rendu cette année particulièrement agréable.

Un grand merci au FabLab de la Doua pour leur accueil et pour nous avoir permis d'imprimer nos différentes productions 3D, ainsi qu'au club de modélisme de Jonage pour leur disponibilité et leurs explications, qui nous ont permis de mieux comprendre le vol d'un aéronef.

Nos remerciements vont également au CLES-FACIL, et en particulier à Rayan, Keshev et Zacharias, pour leur aide dans la prise en main des logiciels OpenRocket et SolidWorks, ainsi que pour leur accompagnement dans le paramétrage avancé des imprimantes BambuLab.

Nous adressons aussi nos remerciements à Grégory, de l'entreprise DASPLET EXPOS, pour son accueil et pour nous avoir initiés aux techniques de fabrication en composites et à l'impression 3D.

Nous remercions également Mathieu pour son aide précieuse dans la découpe vinyle des autocollants destinés à la décoration de notre minifusée.

Enfin, nous exprimons toute notre reconnaissance à l'entreprise bgplastic et à l'ensemble de son personnel pour leur accueil chaleureux et la visite guidée de leur usine spécialisée dans l'injection plastique. Grâce à eux, nous avons pu découvrir et comprendre en détail l'ensemble du processus de production, depuis la conception et la modélisation 3D jusqu'à la fabrication et le contrôle qualité.

I. Introduction

Le projet EZSPACE a débuté en octobre 2024. Son objectif principal était de concevoir, fabriquer et faire voler une mini-fusée conforme au cahier des charges établi par Planète Sciences [1, 2, 3], afin de participer à la campagne nationale C'Space 2025.

Il s'agit de notre toute première expérience dans ce domaine, et les défis sont considérables : les lois de la physique qui régissent le vol d'une fusée, analyse et compréhension du cahier des charges, conception de la structure mécanique et sa décomposition en plusieurs modules, conception complète de la partie électronique, et surtout la réalisation et l'intégration finale pour obtenir un système fonctionnel et fiable.

Le projet est mené sous le suivi de l'AMT et de Planète Sciences. Planète Sciences organise les Rencontres Club Espace (RCE) pour assurer le suivi de notre projet et la campagne C'Space en collaboration avec le CNES, qui fournit le propulseur.

Le cahier des charges impose notamment la stabilité du vol, un système de récupération fiable, ainsi qu'une absence totale de danger pour les personnes et l'environnement.

Pour répondre à ces exigences, nous nous sommes fixés les objectifs suivants :

- Comprendre les lois qui régissent le vol de fusée à petite échelle.
- Prendre en main les outils matériels et logiciels qui aident à la conception et la mise en œuvre d'une minifusée ainsi que les techniques de fabrication.
- Réussir à intégrer l'ensemble des éléments et démontrer la faisabilité durant les vérifications des RCEs et le C'Space.
- Assurer un vol stable avec récupération de la fusée en parfait état.
- Intégrer un système électronique embarqué accompagné d'expériences scientifiques.

- Respecter l'ensemble des règles de sécurité, de dimension et de masse définies par la réglementation.

II. Outils logiciels

1. FreeCAD

Description : FreeCAD est un logiciel libre et open source de conception assistée par ordinateur (CAO) en 3D [6].

Utilisation dans le projet :

- Modélisation initiale des composants de la minifusée, notamment l'ogive ellipsoïdale, Haack, les bagues, la transition, les ailerons et les patins.
- Assemblage des différentes pièces pour vérifier la cohérence dimensionnelle et l'intégration mécanique.
- Génération de fichiers STL et STEP pour l'impression 3D.

Avantages :

- Gratuit et accessible.
- Compatible avec de nombreux formats de fichiers (STEP, STL, IGES, etc.).
- Paramétrique, permettant de modifier facilement les dimensions.

2. Solidworks

Description : SolidWorks est un logiciel professionnel de CAO 3D largement utilisé en ingénierie [7].

Utilisation dans le projet :

- Conception avancée d'ogives et de transition optimisées (ex. profil Haack) ainsi que le système de retenu du propulseur (bouchon et bague filetés).

- Assemblages complexes intégrant la structure mécanique et les systèmes internes.
- Simulation de contraintes mécaniques et validation dimensionnelle.

3. OpenRocket

Description : OpenRocket est un simulateur open source dédié à la conception et à l'analyse de fusées [8].

Utilisation dans le projet :

- Modélisation 3D de la fusée complète (structure, masse, propulsion).
- Simulation des performances de vol : altitude, vitesse maximale, accélération, stabilité.
- Optimisation de la position du centre de gravité et du centre de poussée pour assurer un vol stable.

Avantages :

- Gratuit, conviviale et facile à prendre en main.
- Simulations réalistes basées sur la physique du vol.
- Permet de tester virtuellement des modifications avant fabrication.

4. StabTraj

Description : StabTraj est un outil de simulation spécialisé dans la stabilité et la trajectoire des fusées.

Utilisation dans le projet :

- Vérifier la stabilité de la fusée par le diagramme des critères de stabilité. Nous avons optimisé notre conception pour se rapprocher au mieux du centre du diagramme (portance en fonction de la marge statique)
- Étude de l'influence des paramètres aérodynamiques (forme de l'ogive, taille des ailerons).

- Optimisation des trajectoires pour atteindre les objectifs de vol en respectant les contraintes de sécurité.

5. Arduino

Description : Arduino est une plateforme open source composée d'un microcontrôleur et d'un environnement de développement (IDE) [9].

Utilisation dans le projet :

- Programmation et gestion du module électronique embarqué.
- Acquisition de données (altitude, température, accélération) via des capteurs.
- Contrôle du déclenchement du système de récupération (parachute).

Avantages :

- Grande variété de modules et capteurs compatibles.
- Communauté active et nombreuses ressources pédagogiques.
- Facile à programmer même pour des débutants.

III. Conception de la partie mécanique

Une mini-fusée est composée des éléments suivants :

- Ogive
- Tube principal
- Bague de retenue du propulseur Pro-24
- Module parachute avec trappe d'éjection
- Ailerons
- Module électronique
- Propulseur Pro-24

1. Ogive

L'ogive, placée à l'extrémité avant du tube principal, joue un rôle essentiel dans l'aérodynamisme et la protection de la fusée. Sa forme effilée réduit la traînée et protège les éléments embarqués, tout en servant de point d'ancrage au système de récupération.

Les principales fonctions de l'ogive sont les suivantes :

- Améliorer l'aérodynamisme en réduisant la traînée, c'est-à-dire la force qui s'oppose au mouvement d'un objet dans un fluide. Pour une fusée, cette traînée correspond à la résistance exercée par l'air sur l'ensemble de sa structure pendant le vol, limitant ainsi sa vitesse et son altitude maximale.
- Protéger les charges utiles, qu'il s'agisse de composants électroniques, de capteurs ou d'expériences scientifiques embarquées, contre les chocs, vibrations et contraintes extérieures.

Il existe plusieurs types d'ogives [4], chacune présentant des caractéristiques spécifiques en termes d'aérodynamisme, de fabrication et de performances :

1. Ogive conique

- **Description** : forme simple en cône droit.
- **Avantages** : fabrication rapide et facile, faible coût en matériaux et en temps.
- **Limites** : moins performante sur le plan aérodynamique, générant une traînée plus importante à grande vitesse.

2. Ogive ogivale ou ellipsoïdale

- **Description** : profil arrondi évoquant une balle ou un projectile.
- **Avantages** : excellente pénétration dans l'air, bonne stabilité en vol, réduction notable de la traînée à vitesse modérée.
- **Limites** : fabrication plus complexe, nécessitant un outillage précis, un moulage adapté ou impression 3D haute qualité.

3. Ogive parabolique

- **Description** : profil en courbe parabolique, optimisé pour un écoulement d'air fluide et régulier.
- **Avantages** : traînée minimale à haute vitesse, idéale pour les vols supersoniques et les projets visant des altitudes maximales.
- **Limites** : conception et fabrication plus exigeantes, souvent réalisées par usinage de précision ou impression 3D haute qualité.

Nous avons d'abord modélisé, sous FreeCAD, une ogive ellipsoïdale. Cette première version a été imprimée à l'aide d'une imprimante 3D Bambu Lab, afin de tester la faisabilité et d'évaluer la qualité du résultat.

La Figure 1 illustre, d'une part, la modélisation 3D de l'ogive réalisée sous FreeCAD et, d'autre part, le prototype physique issu de l'impression 3D, permettant de valider la géométrie et la qualité de fabrication.



Figure 1 - Ogive ellipsoïdale : modèle 3D (FreeCAD) et pièce correspondante imprimée en 3D.

Par la suite, nous avons conçu, cette fois sur SolidWorks, une ogive de type Haack. Ce profil particulier offre des avantages notables par rapport aux formes ellipsoïdales ou coniques, notamment une réduction optimale de la traînée grâce à une géométrie issue de calculs aérodynamiques précis.

- **Description** : le profil Haack est défini par une courbe mathématique optimisant la pénétration dans l'air tout en minimisant les pertes d'énergie dues aux frottements.
- **Avantages** : excellente performance aérodynamique, idéale pour les vols à vitesse élevée, compromis optimal entre fabrication et efficacité.

- **Formule de la courbe** : la forme Haack est généralement définie par une équation paramétrique dérivée des travaux de von Kármán et Haack, visant à minimiser la traînée de forme. Il existe deux types principaux d'ogives Haack, définis par la valeur du coefficient C :
 - **LV-Haack ($C=1/3$)** : profil optimisé pour les lanceurs de satellites, offrant un compromis entre réduction de la traînée et volume interne.
 - **LD-Haack ($C=0$)** : profil dit *Low-Drag*, spécifiquement optimisé pour minimiser la traînée aérodynamique.

Dans notre projet, nous avons choisi d'utiliser une ogive LD-Haack, car elle permet de réduire au maximum la résistance de l'air et d'améliorer la stabilité en vol. L'expression mathématique décrivant la courbe de son rayon en fonction de l'abscisse x est donnée par :

$$r(x) = \frac{R}{\sqrt{\pi}} \sqrt{\theta - \frac{1}{2} \sin(2\theta)}$$

où R est le rayon maximal, L la longueur totale de l'ogive, x la distance depuis la pointe et $\theta = \arccos(1 - \frac{2x}{L})$. La Figure 2 illustre le modèle 3D de l'ogive LD-Haack conçu sur Solidworks et l'ogive correspondante après impression 3D.

2. Tube principal

Le tube principal est le corps cylindrique constituant la structure centrale de la fusée. Il a pour rôle de supporter et de relier tous les modules (ogive, module parachute, propulseur et ailerons) tout en offrant un logement pour les éléments internes tels que l'électronique et le parachute. Il assure également la rigidité et la résistance mécanique nécessaires au maintien de l'intégrité de la fusée durant le vol.

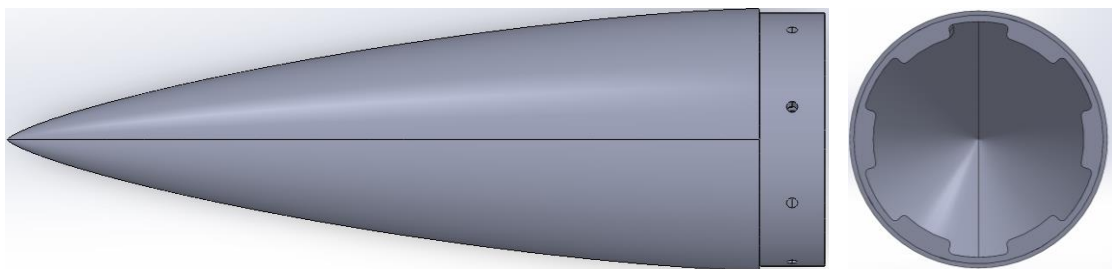
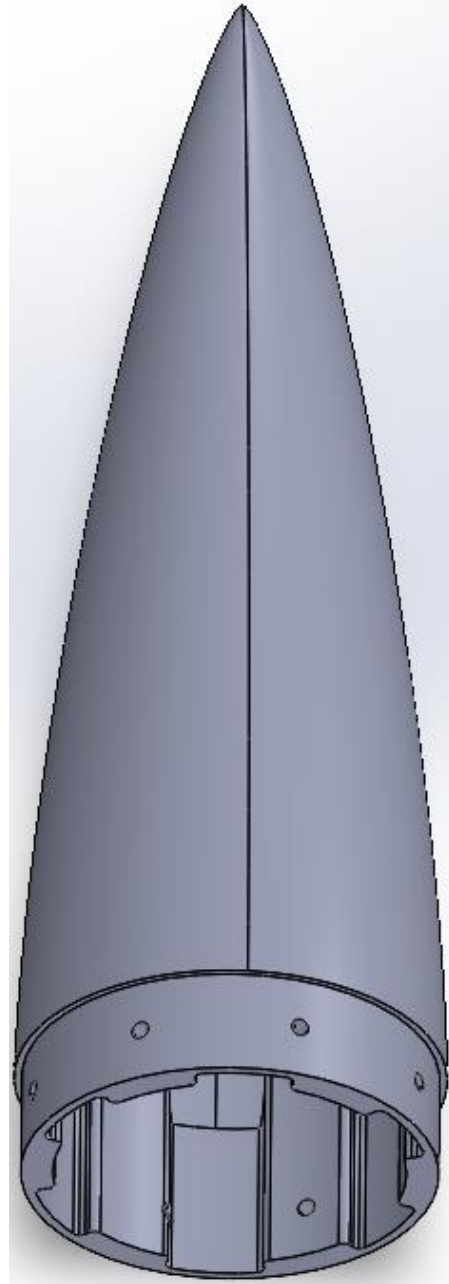


Figure 2 - Ogive Haack : modèle 3D (SolidWorks) et pièce correspondante imprimée en 3D.

Les matériaux couramment utilisés pour sa fabrication sont le carton haute densité, la fibre de verre, la fibre de carbone ou l'aluminium léger. Dans notre projet, nous avons retenu un tube en PVC de 80 mm de diamètre, offrant un bon compromis entre rigidité, poids et facilité d'intégration de la partie électronique et le parachute.



Figure 3 - Transition Haack et tube du bas : modèle 3D (SolidWorks) et pièce correspondante imprimée en 3D.

3. Bague de retenue du propulseur Pro-24

La bague de retenue est une pièce circulaire interne, fixée à l'intérieur du tube principal, qui maintient le propulseur solidement en place et assurer la reprise de poussé qui doit se faire par le bas.

Son rôle est double : d'une part, empêcher tout déplacement du moteur lors de l'allumage et de l'accélération, et d'autre part, faciliter son insertion et retrait. Nous avons mis en œuvre un système vissable comme un bouchon pour maintenir le propulseur (sans outils supplémentaires, Figure 4). Il permet également de faire plusieurs tests d'insertion et de retrait d'un faux propulseur modélisé et imprimé en 3D afin de simplifier et optimiser l'intégration finale.

Cette pièce est conçue et dimensionnée spécifiquement pour le moteur Pro-24, afin d'assurer un ajustement précis et une fixation fiable.



Figure 4 – Système de retenue du propulseur Pro-24 : modèle 3D (SolidWorks) et pièce correspondante imprimée en 3D.

4. Module parachute avec trappe d'éjection

Le module parachute est un compartiment destiné à loger le système de récupération. Il est équipé d'une trappe d'éjection actionnée par un dispositif mécanique.

Sa mission est d'assurer la récupération en douceur de la fusée après le vol, tout en protégeant les éléments électroniques et la structure lors de l'atterrissage.

Nous avons dimensionné la porte d'éjection pour permettre d'intégrer facilement le parachute et la ligne complète incluant les bagues de fixation et les cordes d'attache.

Nous avons utilisé un système d'attache à double cordes pour porter la fusée exactement par son centre de gravité (Figure 5).

Nous avons opté pour un mécanisme d'éjection mécanique, actionné par un solénoïde et commandé électriquement sous une tension comprise entre 9 et 12 V. Ce dispositif permet de libérer la trappe du module parachute de manière rapide et fiable. Un module de conversion DC/DC est utilisé pour générer la tension d'alimentation de 9 V. Ce module est activé par une sortie numérique de la carte Arduino lorsque le compte à rebours (timer) atteint « 0 s », déclenchant ainsi l'actionnement du solénoïde pour l'éjection du parachute.



Figure 5 – Parachute et trappe d'éjection..

5. Ailerons

Les ailerons sont des surfaces planes fixées à la base du tube du bas (réducteur). Leur rôle est de stabiliser la fusée en vol, et de maintenir une orientation verticale en compensant les effets du vent latéral.

On distingue plusieurs formes d'ailerons, chacune avec ses avantages :

- **Trapézoïdaux** : très répandus, offrant un bon compromis entre performance et facilité de fabrication.
- **Elliptiques** : procurent un meilleur rendement aérodynamique, mais sont plus difficiles à fabriquer.
- **Rectangulaires** : simples à réaliser, mais moins performants sur le plan aérodynamique.

Nous avons opté pour quatre ailerons trapézoïdaux à profil symétrique, spécialement conçus avec un système de fixation adapté au diamètre du tube inférieur de 50 mm. Leur géométrie a été soigneusement optimisée afin d'améliorer l'aérodynamisme de la fusée et d'assurer sa stabilité en vol (Figure 6).

Une première version des ailerons a été modélisée sous FreeCAD, puis retravaillée dans SolidWorks afin d'affiner le profil aérodynamique et de renforcer la structure. Les pièces finales ont été imprimées en 3D et soumises à plusieurs séries de tests.

La fabrication s'est révélée particulièrement chronophage, en raison des difficultés rencontrées pour obtenir une impression 3D de qualité avec différents matériaux (PLA, ABS, PETG, etc.). Plusieurs itérations ont été nécessaires, en ajustant progressivement les paramètres d'impression (température, remplissage, orientation des couches, vitesse), afin d'obtenir des ailerons suffisamment résistants.

Malgré ces efforts, les ailerons demeurent perfectibles. Leur résistance mécanique devra être encore améliorée pour mieux supporter :

- les contraintes aérodynamiques générées par la vitesse,
- les charges mécaniques liées au poids et aux efforts en vol,
- les vibrations subies lors du décollage et de la propulsion.

Ces limites ouvrent des pistes d'amélioration, notamment l'utilisation de matériaux composites (fibre de verre, fibre de carbone) ou d'une impression 3D optimisée avec renforts structurels intégrés.

6. Module électronique

Le module électronique regroupe l'ensemble des circuits, capteurs et systèmes embarqués dans la fusée. Il permet de déclencher l'ouverture du parachute au moment idéale de l'apogée et réaliser des expériences scientifiques embarquées. Les composants typiques incluent : les batteries, un BMS, un ou plusieurs microcontrôleur et processeurs, une centrale inertielle (IMU), un altimètre barométrique, un GPS et un émetteur radio pour la transmission des données pour la télémétrie.

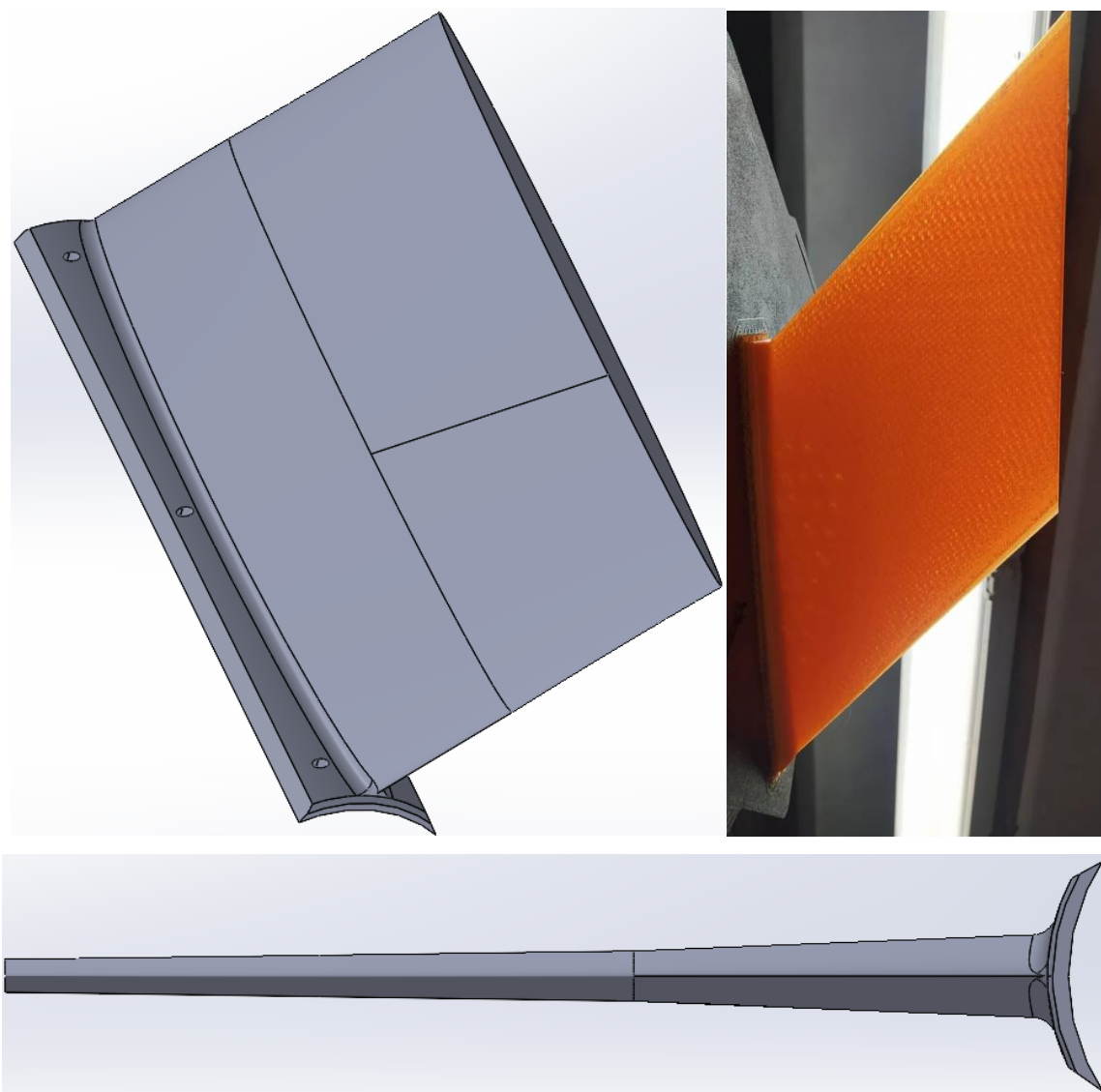


Figure 6 - Aileron: modèle 3D (SolidWorks) et pièce correspondante imprimée en 3D.

Notre système électronique embarqué se compose des éléments suivants :

- **Alimentation** : batterie LiPo 3 cellules (3S) d'une capacité de 2200 mAh, offrant une autonomie d'environ 2 heures. Elle est associée à un BMS (*Battery Management System*) qui assure la gestion sécurisée de la charge et de la décharge, afin de prévenir tout risque de surcharge, de sous-tension ou de surchauffe.

- **Unité de contrôle principale** : une carte Arduino dédiée à la gestion de la minuterie et au déclenchement du système d'éjection du parachute. Cette carte permet :
 - La configuration du compte à rebours via une connexion Wi-Fi et une interface web embarquée.
 - L'affichage en temps réel du timer sur des afficheurs 7 segments multiplexés.
- **Unité de gestion des expériences** : une carte Luckfox, chargée de superviser les expériences embarquées, comprenant :
 - Une caméra pour l'enregistrement vidéo en vol.
 - Un GPS pour le suivi de position.
 - Une IMU (centrale inertielle) pour mesurer l'accélération et l'orientation.
 - Un capteur barométrique pour relever la pression et la température.
 - Un système de stockage sur carte microSD pour l'enregistrement des données.

7. Propulseur Pro-24

Le propulseur Pro-24 [5] est un moteur à propergol solide standardisé, largement utilisé dans le cadre des minifusées, notamment lors de la campagne nationale C'Space, organisée par Planète Sciences et le CNES. Il constitue le système de propulsion principal, fournissant la poussée nécessaire au décollage et à l'ascension de la fusée.

IV. Conception de la partie électronique

La partie électronique de la minifusée regroupe l'ensemble des systèmes nécessaires au contrôle du vol, à la gestion du parachute et à la réalisation d'expériences scientifiques embarquées. Elle se compose de deux sous-ensembles principaux :

1. **L'unité de contrôle principale (minuterie)** : basée sur une carte Arduino Uno R4 WiFi, programmée en C++, chargée de gérer le déclenchement du parachute.
2. **L'unité de gestion des expériences** : basée sur une carte Luckfox Mini Pico B, programmée en Python, dédiée à l'acquisition et à l'enregistrement des données expérimentales.

1. L'unité de contrôle principale

a. Architecture matérielle

Cette unité est basée sur une carte Arduino Uno R4 WiFi [10] et intègre les composants suivants (Figure 7) :

- **Carte shield 4 digits / 7 segments** : affiche en millisecondes la valeur du timer, permettant un contrôle visuel précis.
- **Buzzer** : signale par des tonalités différentes:
 - La détection du jack d'arrachage au décollage.
 - La fin du compte à rebours (à l'apogée), indiquant l'ouverture de la porte du parachute.
- **4 diodes LED** :
 - Indicateur du mode actif (**Configuration** ou **Vol**).
 - État du jack.
 - État de la porte du parachute.
- **3 boutons poussoirs** :
 1. Forcer l'ouverture de la porte du parachute.
 2. Activer le **mode configuration** (WiFi et serveur web).

3. Activer le **mode vol**, désactiver le WiFi et arrêter le serveur web.

- **Entrées numériques :**
 - Détection de l'arrachage du jack (début du vol).
 - Détection de l'état de la porte du parachute.
- **Sortie numérique :** déclenche l'ouverture de la porte et active le module DC/DC 9 V qui alimente le solénoïde.
- **EEPROM I²C :** stocke la configuration (nom et identifiant du projet, valeur du timer en ms, paramètres système).

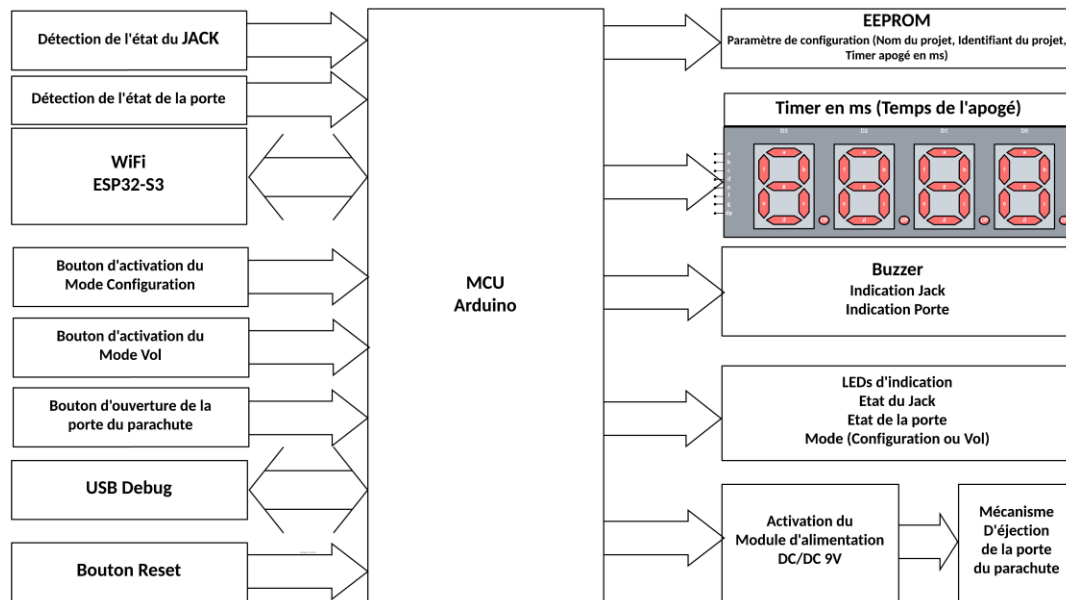


Figure 7 - Schéma en bloc de la carte de contrôle.

b. Architecture logicielle

Le logiciel est développé en C++ avec programmation orientée objet pour une modularité optimale. Les fonctionnalités implémentées incluent :

- Gestion de l'afficheur 7 segments.
- Lecture des boutons et gestion des LED.
- Commande du buzzer.
- Gestion de l'EEPROM (lecture, écriture, mise à jour des paramètres).

- Détection de l'état du jack et de la porte du parachute.
- Décrémentation du timer et déclenchement de l'ouverture à l'apogée.
- Gestion des modes **Configuration** et **Vol**.
- Connexion WiFi et serveur web embarqué pour la configuration.
- Lecture et mise à jour des paramètres stockés en EEPROM via l'interface web.

Le programme embarqué sur la carte Arduino Uno R4 WiFi gère la séquence de déclenchement du parachute ainsi que les modes de fonctionnement de la minifusée.

Il repose sur deux modes principaux : **CONFIGURATION** et **VOL**, et utilise un mécanisme simple de détection de départ par arrachage d'un **jack**.

i. Initialisation et lecture des paramètres

Au démarrage, le programme initialise tous les modules matériels (afficheurs 7 segments, LEDs, boutons, buzzer, interface WiFi, EEPROM, entrées/sorties de contrôle).

Les paramètres principaux (identifiant du projet, nom du projet, délai parachute) sont lus depuis l'EEPROM (Figure 8). Le système affiche ensuite l'état actuel (mode actif, IP WiFi, état de la porte parachute) sur le port série (Figure 9).

ii. Mode Configuration

Le système démarre par défaut en mode CONFIG.

Dans ce mode :

- Le WiFi est activé et connecté au réseau prédéfini (SSID *EZSPACE*).
- Un serveur web intégré permet de configurer le délai parachute et de passer au mode VOL.
- L'adresse IP locale est affichée sur le port série, permettant un accès via navigateur.

- Les LEDs et l'afficheur 7 segments informent visuellement de l'état du système.

iii. Passage en mode Vol

Le changement de mode s'effectue via l'interface web par une requête HTTP (/set?mode=1) comme illustré dans la Figure 9.

En **mode VOL** :

- Le WiFi et le serveur web sont désactivés pour libérer toutes les ressources au traitement des événements de vol.
- Le système se met en attente de l'événement de départ (arrachage du jack).

iv. Détection du départ et séquence de déclenchement

Le jack agit comme un contact de sécurité :

- Jack retiré : le système interprète cet événement comme le début du vol et lance le compte à rebours programmé (ex. 5000 ms).
- Compte à rebours terminé : la sortie de commande active le module DC/DC 9V, alimentant le solénoïde pour libérer la trappe du parachute.
- Un buzzer signale également la fin du compte à rebours.

v. Réinitialisation après le vol

Une fois le jack remis en place, le compte à rebours est réinitialisé et le système revient en attente. Cette réinitialisation permet de préparer rapidement la fusée pour un nouvel essai.

vi. Indications visuelles et auditives

- **LEDs** : affichent le mode actif, l'état du jack et l'état de la porte parachute.

- **Buzzer** : émet un signal lors de la détection du départ et à l'ouverture du parachute.
- **Afficheurs 7 segments** : montrent en temps réel la valeur du compte à rebours.

System Info

Project ID: MF58

Project Name: EZSPACE

Board: Arduino UNO R4 WiFi

Current Delay: 5000 ms

IP Address: 10.220.87.252

System Mode: Config

Set Configuration

Project ID:

Project Name:

Parachute Delay (5000 to 10000 ms):

System Mode

Current Mode: **CONFIG**

Figure 8 - Interface web de configuration via WiFi.

```
Connected to COM17 (9600 bps) [Icons] Baud rate: 9600
??System CLI Ready. Type 'help'.

=====
Project ID      : MF58
Project Name    : EZSPACE
Parachute Delay : 5000 ms
=====

EZSPACE - FLIGHT SYSTEM INITIALIZED
=====

Current Time      : 2025-08-14T16:41:36
WiFi IP Address   : 0.0.0.0
Project ID        : MF58
Project Name      : EZSPACE
Parachute Delay   : 5000 ms
System State      : Idle
Boot Mode         : CONFIG
=====

# The parachute door is closed!
=====

Attempting to connect to SSID: EZSPACE

WiFi connected
To access, open: http://10.220.87.252
Web server started on ESP-AT
JACK removed - starting countdown
At countdown end
Opening the Parachute Door...
JACK inserted - reset countdown
Client connected
Request: GET /set?mode=1 HTTP/1.1
Mode switched to FLIGHT
Client disconnected

—
```

Figure 9 - Interface série et traces du système de contrôle.

2. L'unité de gestion des expériences

c. Architecture matérielle

Cette unité repose sur une carte Luckfox Pico Mini B [11] et intègre (Figure 10):

- IMU (centrale inertielle) MPU-9250 pour mesurer l'accélération et l'orientation.
- Capteur barométrique BMP280 pour mesurer la pression et la température.
- GPS NEO-6 pour la localisation en vol.
- Caméra pour l'enregistrement vidéo.
- Carte microSD pour le stockage des données et vidéos.

d. Architecture logicielle

Le développement logiciel de l'unité de gestion des expériences repose sur une architecture de type micro-services, permettant de séparer les fonctions en modules indépendants et spécialisés.

Il comprend :

- Application Python – GPS : acquisition et enregistrement des données de position fournies par le récepteur GPS.
- Application Python – Centrale inertielle (IMU) : acquisition et enregistrement des mesures d'accélération et de rotation, avec datation synchronisée à l'heure et à la date issues du GPS.
- Application Python – Capteur barométrique (BMP280) : acquisition et enregistrement des données de pression atmosphérique et de température, également datées en fonction de l'heure et de la date GPS.
- Scripts Bash : automatisation du lancement des services Python au démarrage de la carte.
- Système de gestion des fichiers : organisation automatique des données dans des dossiers horodatés, facilitant le classement et l'analyse post-vol.

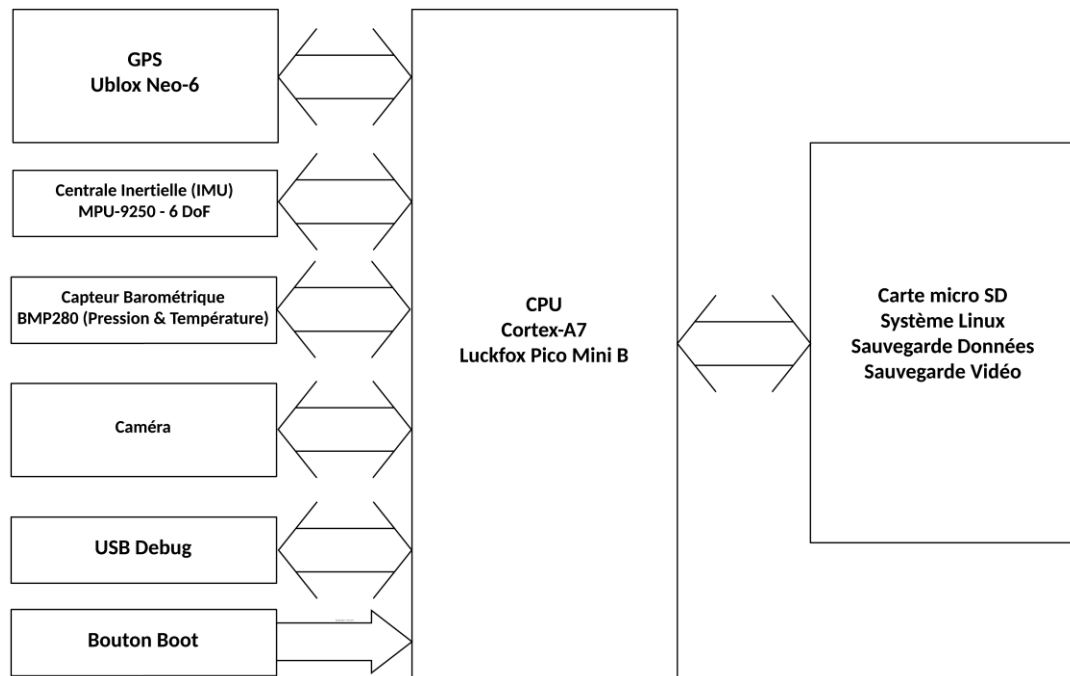


Figure 10 - Schéma en bloc de la carte d'expérience.

V. Simulation et Intégration

1. Simulation

La minifusée a été modélisée et simulée à l'aide du logiciel **OpenRocket** afin de déterminer son apogée ainsi que la position de son centre de masse. OpenRocket permet d'intégrer l'ensemble des composants mécaniques avec leurs dimensions, leurs masses en fonction des matériaux utilisés, ainsi que les éléments électroniques et de récupération.

Dans notre modèle, nous avons inclus :

- L'ogive
- Le tube principal
- La transition
- Le tube inférieur
- Les bagues de fixation
- Les ailerons
- Le système parachute
- Le propulseur Pro-24

Le tube principal possède un diamètre de 80 mm, tandis que le tube inférieur est de 50 mm.

La Figure 11 et la Figure 12 illustrent respectivement la vue en plan et la représentation 3D de la minifusée sur OpenRocket.

Les données obtenues (structure, dimensions, masse, centre de gravité et apogée) ont ensuite été importées dans StabTraj afin de vérifier la stabilité de la fusée et d'analyser sa trajectoire.

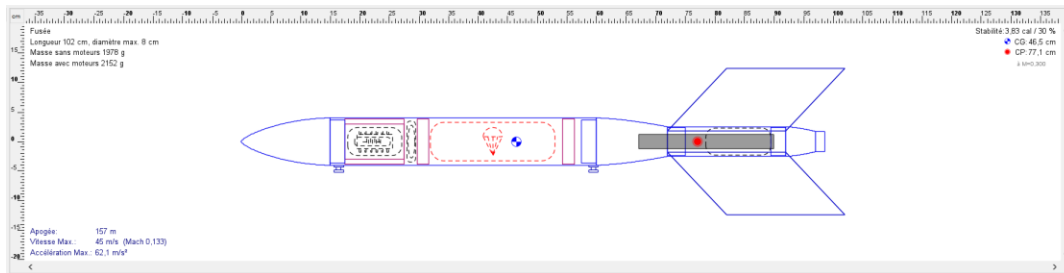


Figure 11 - Vue de face de la minifusée sur OpenRocket.



Figure 12 - Vue 3D en perspective de la minifusée sur OpenRocket.

a. Stabilité

La Figure 13 présente la fusée modélisée dans StabTraj. Les résultats confirment que la configuration retenue est statiquement stable.

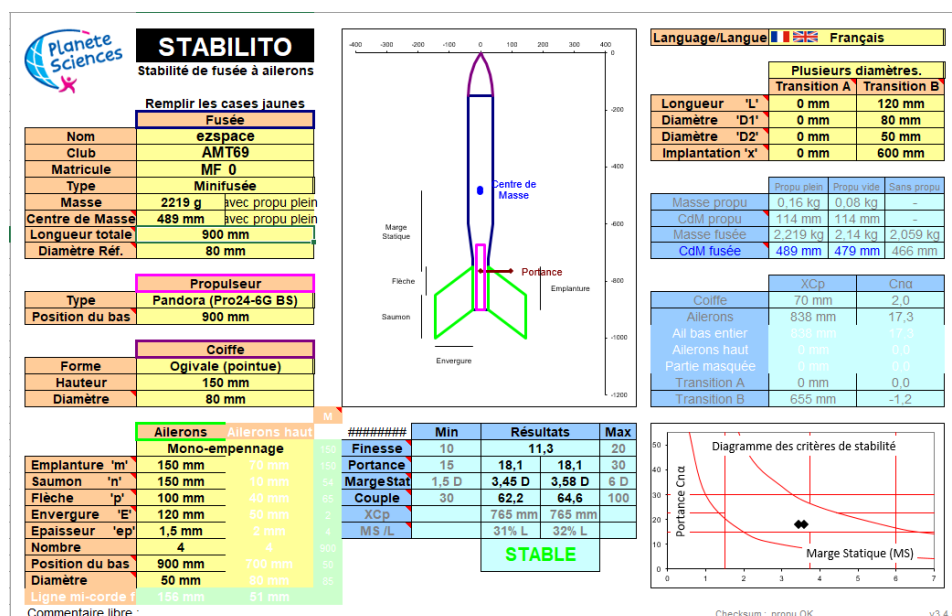


Figure 13 - Stabilité de la minifusée sur StabTraj.

b. Trajectoire

La Figure 14 montre la trajectoire simulée dans StabTraj, permettant d'anticiper le comportement en vol et de valider les paramètres de conception.

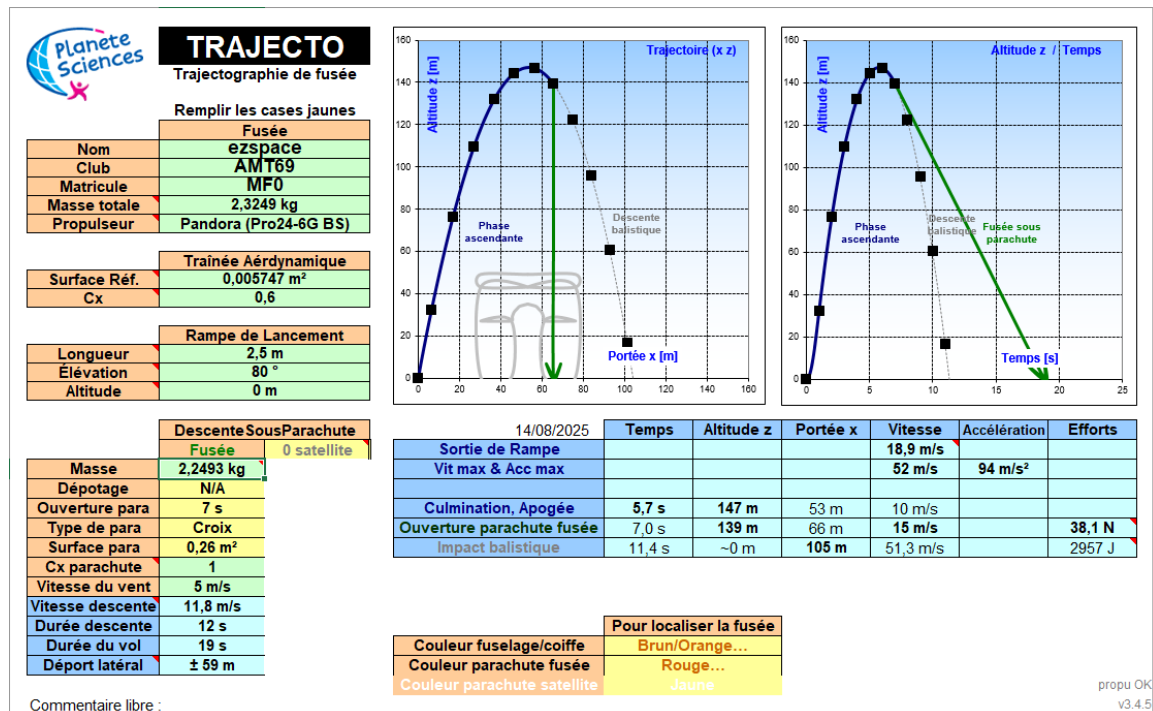


Figure 14 - Trajectoire de la minifusée sur StabTraj.

2. Intégration

Après validation par simulation, nous avons procédé à la mise en œuvre et à l'intégration physique de la minifusée.

Les étapes principales ont été les suivantes :

1. Découpe des tubes supérieur (80 mm) et inférieur (50 mm).
2. Découpe de la trappe d'éjection du parachute.
3. Marquage et perçage des points de fixation pour l'ogive, les bagues, la transition et les ailerons. Nous avons conçu et imprimé des pièces en 3D pour simplifier le perçage et l'alignement avec précision.
4. Impression 3D des pièces : ogive, transition, bagues, le système de retenue, ailerons et patins.

5. Installation des systèmes électroniques.
6. Montage des bagues, du mécanisme d'éjection et du parachute.
7. Assemblage de la transition, du tube inférieur, des ailerons, des bagues et le système de retenue du propulseur.
8. Programmation des cartes électroniques.
9. Tests et validation fonctionnelle.
10. Peinture et décoration de la fusée.

VI. Analyse de l'expérience

1. Contexte et objectif de l'expérience

L'expérience visait à enregistrer en vol, à une cadence de 100 ms, les données issues de plusieurs capteurs :

- GPS NEO-6 (latitude, longitude, altitude, date, heure)
- IMU MPU9250 (accélération, orientation)
- Capteur barométrique BMP280 (pression, température)
- Caméra embarquée (flux vidéo)

L'ensemble devait être horodaté grâce à l'horloge interne synchronisée par le GPS, afin de générer des fichiers et vidéos portant un nom de fichier basé sur la date et l'heure précises du vol.

2. Problème rencontré

c. Synchronisation GPS au lancement

Lors du vol, le GPS n'a pas pu acquérir un *fix* (position + heure UTC) avant le décollage.

Ce problème a eu plusieurs conséquences :

- Les fichiers générés par l'IMU, le BMP280, le GPS lui-même et la caméra n'ont pas été horodatés correctement.
- La vidéo et les logs n'ont pas pu être associés directement dans une chronologie fiable.

d. Cause probable

L'analyse des conditions laisse penser que :

- L'antenne GPS était mal positionnée, installée au niveau de la transition et orientée vers le bas. Au moment de la mise sous tension, le GPS n'a pas pu se synchroniser aux satellites.

- La plateforme RV1103 était fortement sollicitée (capture vidéo + acquisitions multiples à 100 ms), ce qui a pu retarder le traitement des données. Nous n'avons malheureusement pas pu avoir accès à la diode LED d'indication du *fix* GPS à cause des contraintes d'intégration
- Aucun mécanisme de pré-synchronisation ou de stockage temporaire de l'heure GPS n'était en place pour assurer un démarrage horodaté même en l'absence de *fix* immédiat.

3. Données obtenues après atterrissage

a. Acquisition post-vol

Quelques minutes après l'atterrissage, le GPS a réussi à se synchroniser et à enregistrer la date, l'heure et les coordonnées du lieu d'atterrissage.

b. Précision en position fixe

Malgré la position fixe de la fusée, les mesures montrent :

- Précision horizontale : variation des coordonnées de quelques mètres autour du point réel.
- Précision verticale : altitude moyenne ~438,8 m, avec un écart-type d'environ ± 3 m, et des extrêmes compris entre 423 m et 448 m.

Ces variations sont normales pour un GPS à utilisation civil et reflètent les limitations dues:

- à la géométrie instantanée des satellites visibles,
- aux erreurs de propagation dans l'atmosphère,
- au bruit de mesure du module.

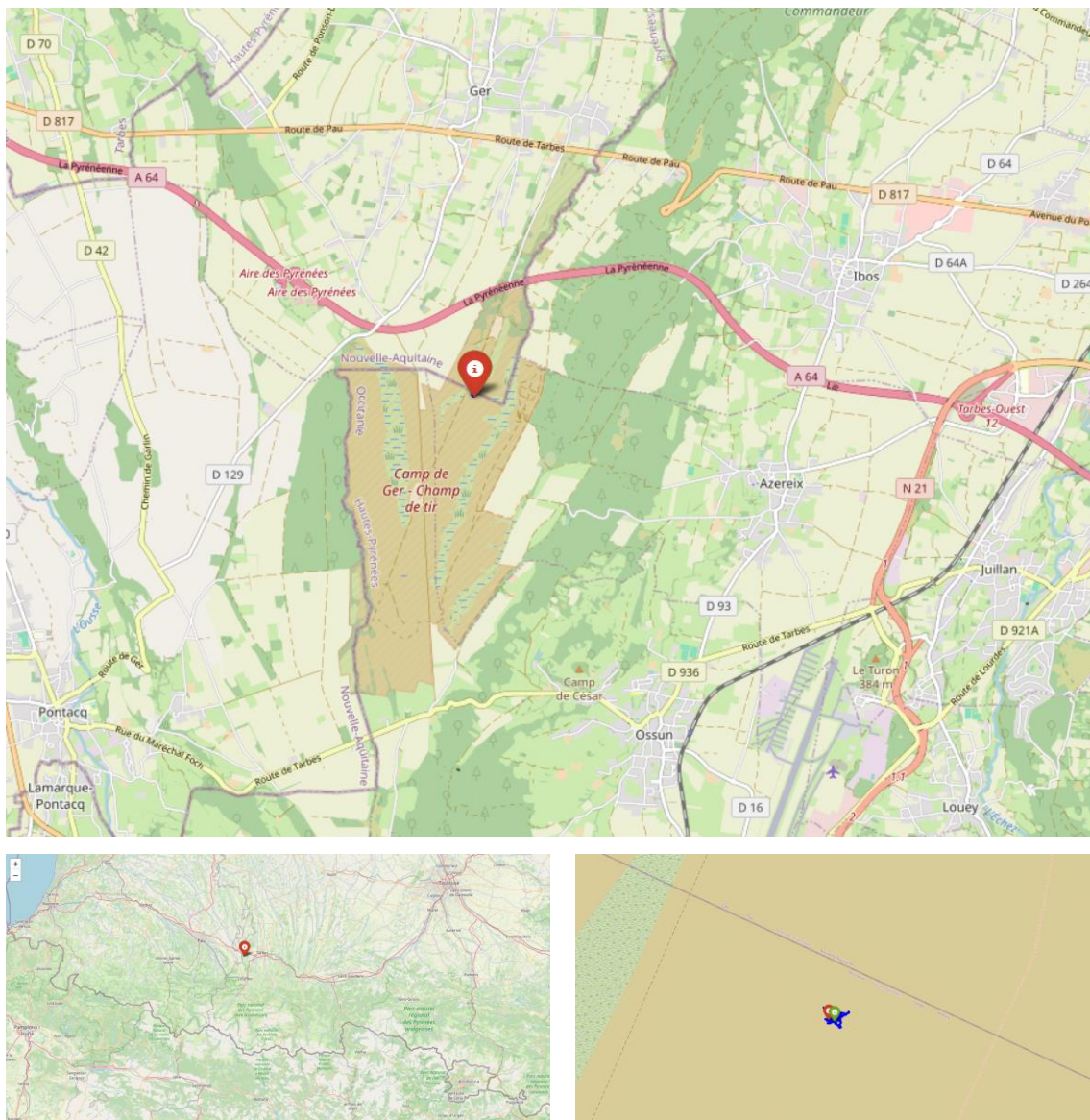


Figure 15 - Position d'atterrissage de la minifusée.

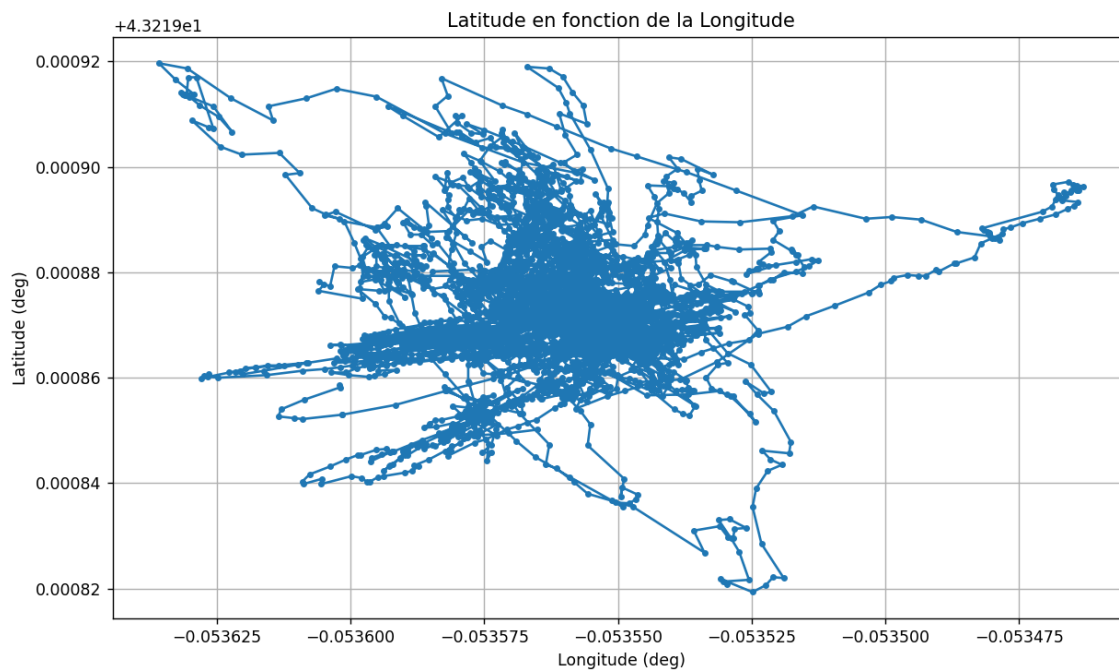


Figure 16 - Latitude en fonction de la Longitude.

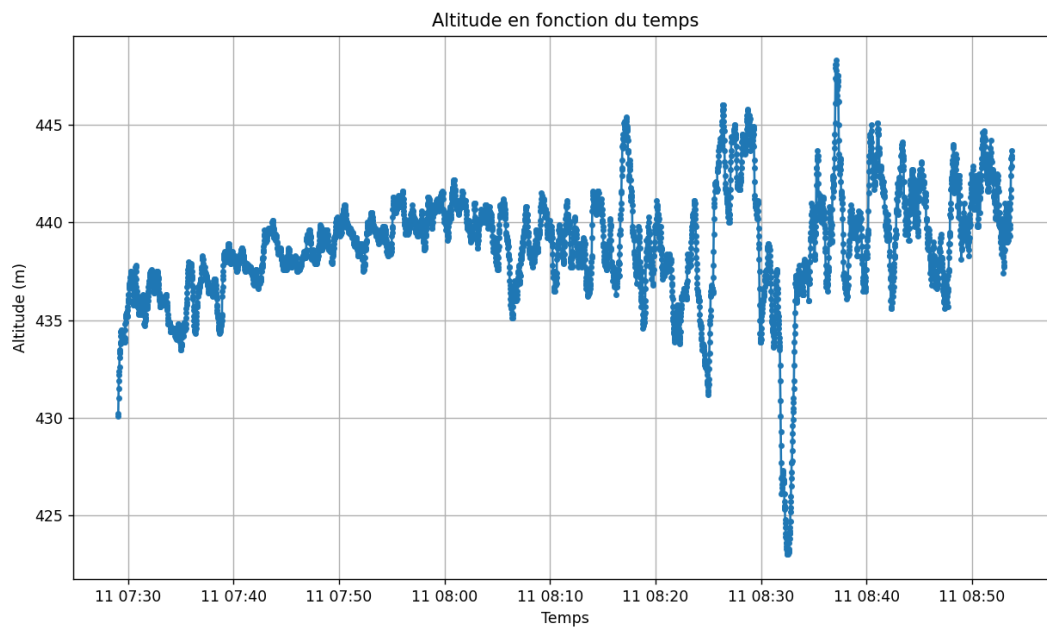


Figure 17 - Altitude en fonction du temps.

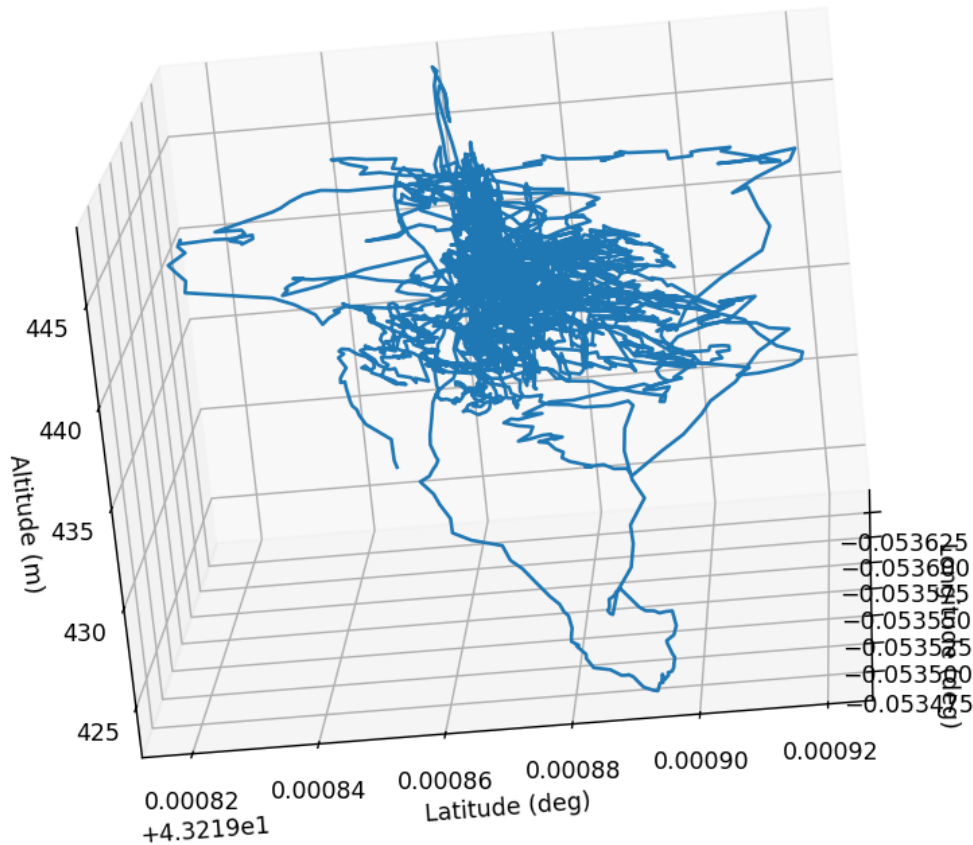


Figure 18 - Trajectoire 3D (Longitude, Latitude, Altitude).

4. Analyse des performances du système embarqué

a. Charge processeur

Le processeur RV1103 exécutait simultanément :

- Un service d'acquisition vidéo haute résolution
- La lecture et l'enregistrement des données GPS, IMU et BMP280
- L'écriture simultanée sur carte microSD
- Le traitement à haute fréquence (10 Hz)

Cette charge élevée a pu :

- Ralentir certains services, notamment la lecture GPS
- Retarder le traitement du *Time To First Fix*
- Créer des risques de pertes de données ou de décalages temporels.

b. Cadence de 100 ms

Si elle permet une haute résolution temporelle, cette cadence :

- Accroît considérablement le volume de données
- Augmente les accès mémoire et microSD
- Peut saturer les ressources CPU/IO sur une carte à faible puissance.

5. Leçons apprises

1. Importance de la synchronisation initiale GPS : sans *fix* au lancement, l'exploitation des données devient difficile.
2. Précision verticale : le GPS montre des variations exploitables pour estimer l'incertitude, mais insuffisantes pour remplacer un altimètre barométrique dans les phases critiques du vol.
3. Limites matérielles : le RV1103 atteint ses limites quand il doit gérer simultanément plusieurs flux haute fréquence.

6. Pistes d'amélioration pour EZSPACE-X (C'Space 2026)

a. Améliorations matérielles

- Utiliser une plateforme plus puissante avec plus de cœurs et de RAM.
- Ajouter un module GPS avec mémoire RTC interne (Timekeeping) pour fournir l'heure immédiatement après mise sous tension.
- Employer un altimètre barométrique dédié et indépendant pour l'enregistrement critique de l'altitude.

b. Améliorations logicielles

- Mettre en place un mode de pré-initialisation GPS associé à une horloge temps réel (RTC) dédiée, permettant d'alimenter le module avant le lancement afin qu'il acquière un *fix* et l'heure GPS en amont du décollage.
- Réduire la fréquence d'acquisition des capteurs à 10 – 20 Hz pour économiser les ressources, ou utiliser deux cartes séparées (une pour la vidéo, une pour les capteurs).

- Synchroniser tous les capteurs via un temps système unique et prévoir une sauvegarde des données dans un format unique horodaté.

c. Améliorations opérationnelles

- Positionner l'antenne GPS de manière optimale, à ciel ouvert, avant le lancement et prévoir une indication lumineuse, affichage sur écran et confirmation audio de la synchronisation GPS.
- Tester la latence de *fix* dans les conditions réelles (antenne, météo, orientation).

VII. Conclusion et perspectives

Le vol réalisé lors de la campagne C'Space 2025 s'est déroulé de manière nominale, avec un décollage réussi, une ouverture correcte du parachute et une descente sous parachute parfaitement au centre de gravité comme attendue pour la récupération de la fusée.

Cette année de travail a été l'occasion d'acquérir de nombreuses compétences, notamment en modélisation 3D, en simulation numérique, en fabrication additive (impression 3D), la prise en main du système électronique (minuterie et expérience) fournis par l'AMT ainsi que le travail collaboratif au sein d'une équipe motivée et pluridisciplinaire.

Notre premier prototype immatriculé MF58 du projet EZSPACE et la participation à la campagne C'Space 2025 nous ont permis de valider le concept de la minifusée et de démontrer sa faisabilité technique. Cependant, l'expérience a également mis en évidence plusieurs axes d'amélioration :

- Amélioration aérodynamique et réduction de masse : affiner la forme générale et alléger la structure pour réduire la traînée et améliorer les performances en vol.
- Ailerons renforcés : concevoir des pièces capables de supporter des contraintes mécaniques plus élevées, tout en maintenant un profil optimisé (explorer et tester de nouveaux matériaux).
- Intégration électronique : compacter et organiser les modules électroniques pour limiter l'encombrement et réduire la complexité du câblage.
- Télémétrie temps réel : intégrer un système de transmission permettant un suivi instantané des paramètres critiques du vol.
- Développement de cartes électroniques dédiées : créer des circuits sur mesure, adaptés aux besoins spécifiques du projet.

- Adopter une plateforme plus puissante (processeur multi-cœurs, mémoire accrue) afin de gérer simultanément l'acquisition vidéo, les capteurs et le stockage sans surcharge.
- Ajouter un module GPS avec mémoire RTC interne pour fournir immédiatement l'heure après mise sous tension.
- Mettre en place un mode de pré-initialisation GPS avec RTC dédiée, permettant l'alimentation et la synchronisation avant le décollage pour obtenir un *fix* et l'heure GPS en amont.
- Placer l'antenne GPS dans une position dégagée et optimale, à ciel ouvert, pour accélérer l'acquisition des signaux satellites.
- Réaliser des tests de latence de *fix* dans des conditions représentatives (météo, orientation, positionnement) avant le lancement.
- Employer un altimètre barométrique dédié et indépendant pour garantir un enregistrement fiable de l'altitude.
- Optimiser la fréquence d'acquisition des capteurs (10 – 20 Hz) ou séparer les tâches sur deux unités (une pour la vidéo, une pour les capteurs) afin d'alléger la charge processeur.
- Synchroniser l'ensemble des capteurs sur un temps système unique, avec un format de sauvegarde unifié et horodaté.

Ces évolutions viseront à accroître la fiabilité, la précision des mesures et la performance globale en vue de la participation au C'Space 2026. L'édition C'Space 2025 était une expérience formidable et riche en découverte et apprentissage. Elle nous a permis de voir le fruit de notre travail décoller et atterrir avec succès. Cette expérience nous a motivé à poursuivre pour le C'Space 2026 avec **EZSPACE-X**, une minifusée **EZSPACE** eXpérimentale qui a également pour ambition de recruter de nouveaux membres de l'AMT !

Bibliographie

- [1] <https://www.planete-sciences.org/espace/Minifusee/Presentation>
- [2] https://www.planete-sciences.org/espace/IMG/pdf/cahier_des_charges_minif_pro2x.pdf
- [3] https://www.planete-sciences.org/espace/IMG/pdf/fiche_controle_minif_v3_1.pdf
- [4] https://fr.wikipedia.org/wiki/A%C3%A9rodynamique_de_la_pointe_avant
- [5] <https://pro38.com/product-category/pro-24/>
- [6] <https://www.freecad.org/index.php?lang=fr>
- [7] <https://www.solidworks.com/fr>
- [8] <https://openrocket.info/>
- [9] <https://www.arduino.cc/>
- [10] <https://docs.arduino.cc/hardware/uno-r4-wifi/>
- [11] <https://wiki.luckfox.com/Luckfox-Pico/Luckfox-Pico-quick-start/>