

Rapport de vol

Projet MF63 - SubSonic

Antoine Jeanson - Eva Rangayen - Rémi Leriche



Remerciements	3
Réalisation du projet Subsonic	4
Simulation OpenRocket et Stabtraj	4
Electronique embarquée	7
Expérience	7
Prototypage	7
Séquenceur (Système de récupération)	7
Prototypage	7
Réalisation sur breadboard à pastilles (“perma-proto” ou “perfboard”)	8
Panneau de contrôle	8
Éléments n’ayant pas été mis en œuvre	8
Déclenchement du système de récupération par altimètre barométrique	8
Conception des pièces et du modèle 3D	9
Réflexion général	9
Partie “haute”	9
L’ogive	9
L’anneau	10
Module batterie	10
Partie “basse”	11
Tube alu	11
Ailerons	11
Compte rendu du vol	13
Traitement de données	13
Déroulement du vol	13
Retour sur expérience	14
Problèmes rencontrés lors du projet	14
Amélioration à voir pour un futur projet	14

Remerciements

Nous tenons dans un premier temps à remercier l'école ELISA Aerospace, sans qui ce projet n'aurait jamais pu avoir lieu.

L'association Virtu'ELISA a elle aussi une place importante dans le projet, de par les nombreux prêts de matériel qu'elle nous a accordés (Désolé pour la meuleuse Erwan !).

L'association ELISA Space nous a fourni la structure nécessaire au bon déroulement du projet.

Bien entendu, rien de tout cela ne serait possible sans le support du CNES et de Planète Sciences, qui nous ont apporté connaissances technique et matériel pour mener à bien le travail de notre équipe, de la conception de la fusée à son lancement.

Nous tenons enfin à remercier le 1er régiment de hussards parachutistes et l'ensemble du personnel du camp de Ger pour leur accueil.

Réalisation du projet Subsonic

Simulation OpenRocket et Stabtraj

Le club utilise depuis quelques années, en complément de Stabtraj, le logiciel OpenRocket.

Ce projet n'y a pas fait exception.

OpenRocket nous permet de concevoir la fusée dans son ensemble très rapidement, et d'aller jusqu'à son optimisation. La maquette numérique ainsi obtenue nous permet de suivre les performances de la fusée tout au long de sa conception.

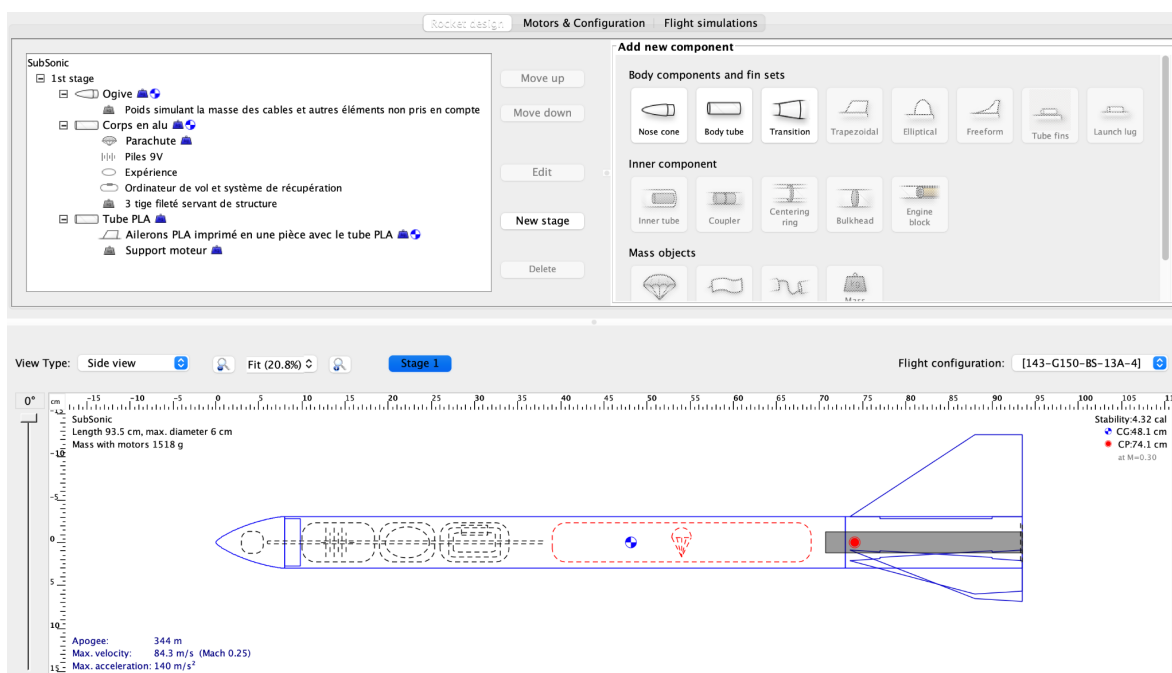
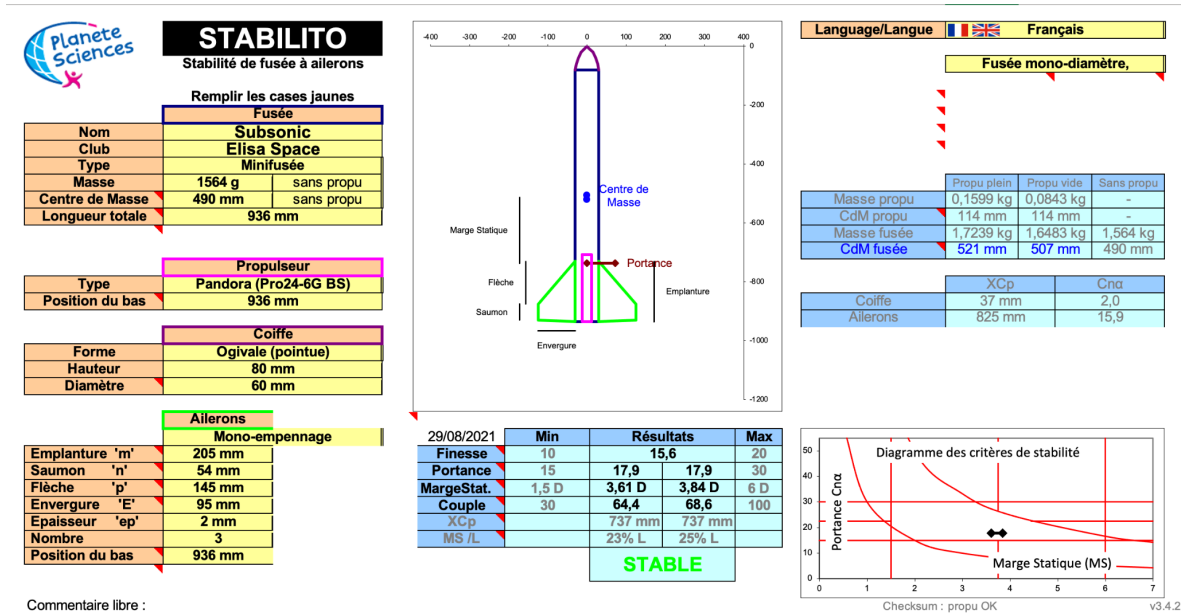


Figure 1 : Maquette numérique de la fusée sous OpenRocket (masse estimée)

De gauche à droite dans le corps de la fusée :

- Module batterie composé de deux anneaux, d'une plaque supportant les batteries et d'un système permettant des les retirer facilement grâce à une clé à laine.
- Module expérience composé lui aussi de deux anneaux et deux breadboards à pastilles comportant la carte Arduino et les capteurs.
- Module séquenceur composé à nouveau de deux anneaux en d'une breadboard à pastilles comportant une carte Arduino.

Rapport de vol - MF63 SubSonic



Rapport de vol - MF63 SubSonic

Résultats détaillés	Temps	Altitude z	Portée x	Vitesse	Accélération	Angle
	s	m	m	m/s	m/s ²	°
Décollage	0	0	0	0	-	80
Sortie de Rampe	0,21	2,25	0,40	22,2	105,1	80,0
Vit max & Acc max	-	-	-	73	129,8	-
Fin de Propulsion	1,1	52	10	72	13,0	78,3
Culmination, Apogée	7,6	271	97	13	9,8	0,0
Impact balistique	15,3	~0	188	68	7,0	-81,2
Ouverture parachute fusée	7,2	270	92	13	9,9	17,3
Impact fusée sous para.	33	~0	-37 221	10	9,8	-

Figure 4 : Suite du Trajecto définitif

Electronique embarquée

Expérience

L'expérience que nous avons choisi de mettre en œuvre à bord de la fusée est principalement composée d'un enregistreur de vol. Cet enregistreur comprend des capteurs de différents types, un micro-contrôleur Arduino Nano et un lecteur de carte MicroSD.

Les capteurs embarqués sont :

- MPU-6050, IMU 6 axes (gyroscope 3 axes + accéléromètre 3 axes)
- BMP-280, baromètre

Ensemble, ces capteurs nous permettent de déterminer l'accélération, la vitesse de rotation et l'altitude du véhicule dans l'espace. Par calcul, nous pouvons aussi déterminer, par différentes méthodes, sa vitesse, sa position, son orientation, entre autres.

Les données du vol sont enregistrées sans retraitement à bord, ce qui nous permet de l'effectuer au sol.

Prototypage

Dans un premier temps, nous avons commencé par réaliser un prototype de l'enregistreur de vol sur breadboard, afin de valider les branchements et le code.

Lors de cette phase, nous avons pu nous rendre compte de la sensibilité du bus i2c utilisé pour les capteurs à la qualité du câblage. Il en va de même avec le bus SPI permettant de communiquer avec la carte SD.

Séquenceur (Système de récupération)

Le séquenceur est l'organe responsable du déclenchement du parachute.

La trappe latérale contenant le parachute est maintenue fermée par le bras d'un servo-moteur de 9g, et est libérée par la rotation de ce dernier d'un angle de 90°.

Le système est lui aussi contrôlé par une carte Arduino Nano.

Prototypage

Afin de valider le bon fonctionnement du système, il a été, de la même façon que pour l'expérience, réalisé sur une breadboard.

C'est à ce moment que nous nous sommes rendus compte que le micro-contrôleur redémarrait lors d'un mouvement du servomoteur. La source du problème était le courant demandé par le servomoteur : il était trop important, provoquait une chute de la tension du rail 5V et faisait crasher le processeur temporairement (par opposition à un black-out, ce type de phénomène est nommé brown-out).

En effet, le servomoteur était alimenté par le régulateur 5V intégré au microcontrôleur, qui supporte sans problème le courant d'alimentation du servomoteur lorsqu'il ne bouge pas, mais pas lorsqu'il est en mouvement ou qu'il résiste à un effort externe.

La meilleure solution à ce problème est de réaliser un circuit de régulation externe au microcontrôleur. Cependant, par manque de temps, nous avons tenté de placer un grand

condensateur sur le rail 5V. A notre surprise, le problème s'est volatilisé, ce qui nous a aussi fait penser à un problème d'interférence.

Si nous avions poussé la conception de l'électronique jusqu'à la fabrication d'un circuit imprimé, nous aurions pu prévoir un système d'alimentation adapté.

Réalisation sur breadboard à pastilles ("perma-proto" ou "perfboard")

Une fois que nous étions sûrs de nos branchements et que le code était suffisamment avancé pour le permettre, nous avons réalisé une version permanente de l'expérience sur une plaque à essais à pastilles.

Les circuits ainsi réalisés sont ensuite assemblés dans la structure interne de la fusée.

Panneau de contrôle

Il est important de pouvoir interagir avec l'utilisateur afin de définir correctement les bons réglages et de lui présenter les bonnes informations.

Il s'agit d'un élément du système embarqué que nous avons réalisé très tard dans le projet, car il était plus important pour nous de terminer les fonctionnalités du code avant. Le panneau de contrôle est équipé de quatre interrupteurs et quatre LEDs, deux pour chaque carte embarquée.

Pour chacune d'entre elles, un interrupteur et une LED sont dédiés au contrôle et à l'indication de l'alimentation, et les autres sont dédiés à l'état du code.

D'un point de vue mécanique, il s'agit d'une pièce imprimée en 3D se fixant à la structure interne des systèmes embarqués, directement sur les tiges filetées.

Il est placé en face d'une découpe dans le tube.

Éléments n'ayant pas été mis en œuvre

Déclenchement du système de récupération par altimètre barométrique

Initialement, nous avons prévu de tenter de déclencher le parachute avec un signal envoyé par l'expérience au système de récupération.

Cela n'a pas été effectué par manque de temps.

Par ailleurs, la communication UART entre les deux cartes n'étant pas nécessaire, nous n'avons pas non plus implémenté l'électronique d'isolation galvanique entre elles.

Conception des pièces et du modèle 3D

Réflexion général

L'idée de départ concernant la structure était de permettre le remplacement ou le changement de pièces/modules facilement. Pour cela, nous avons opté pour une structure interne attachée à l'ogive pouvant être retirée simplement en tirant sur celle-ci. L'utilisation de 3 tiges filetées s'est donc imposée. Pour fixer les cartes, capteurs et batteries dans la fusée, nous avons décidé de réaliser un anneau universelle (basé sur les dimensions des breadboards à pastilles (cf. figure 6)) qui vient se glisser le long des tiges filetées et pouvant accueillir les modules que nous avons créé par la suite (cf. figure 5).

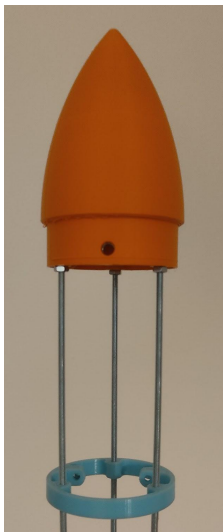


Figure 5 : Montage tiges filetées et ogive

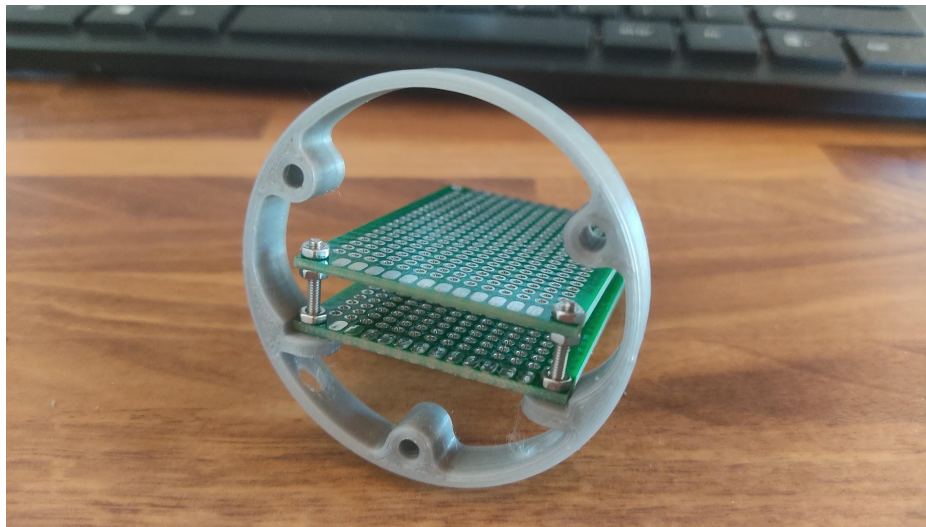


Figure 6 : Breadboards fixées sur un anneau

La fusée a été divisée en deux parties principales, ce qui nous a permis de travailler chacun de notre côté malgré la distance qui nous séparait (surtout pendant la phase de fabrication). Lors de la fabrication, Antoine en Eva se sont occupés de la partie dite “haute” qui regroupe l'ogive ainsi que tout le système interne de la fusée (parachute compris) et Rémi s'est occupé du corps de la fusée et des ailerons.

Partie “haute”

L'ogive

Cette pièce a été l'une des premières pièces réalisée et imprimée. Elle a connu deux versions. Une première (cf. figure 7) abritant des emplacements dans lesquels des écrous devaient être glissés afin d'attacher les tiges filetées et de fixer l'ogive sur la partie basse. Une deuxième (cf. figure 8) comprenant des emplacements dans lesquels des inserts sont posés pour remplacer les écrous de la version précédente. La forme de l'ogive a aussi été légèrement modifiée sur celle-ci.



Figure 7 : Ogive version 1



Figure 8 : Ogive version 2

L'anneau

Pièce conçue dans le but d'accueillir des modules (module expérience ou batterie par exemple), s'enfile simplement dans les tiges filetées en est retenue par des écrous (cf. figure 6).

Module batterie

Pièce fixée sur deux anneaux servant à accueillir les deux batteries nécessaires à l'alimentation de la fusée (cf. figure 9). Une dernière pièce sert à retenir les batteries une fois celle-ci en place (cf. figure 10).

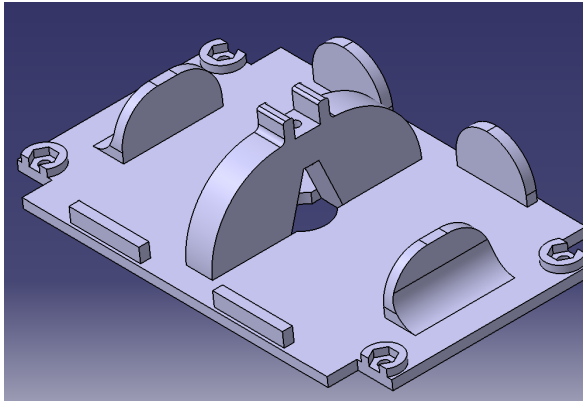


Figure 9 : Modèle 3D du support batterie

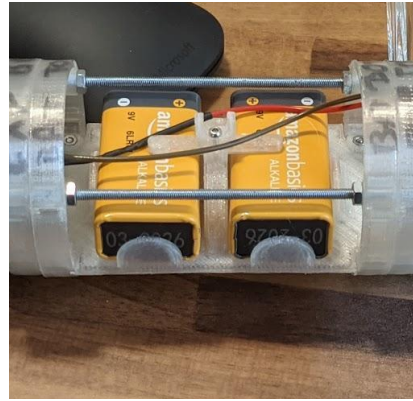


Figure 10 : Module batterie

Module expérience

Module composé de deux anneaux permettant de tenir en place deux breadboard à pastilles sur lesquelles sont soudées les cartes et capteurs (cf. figure 6 et 11).

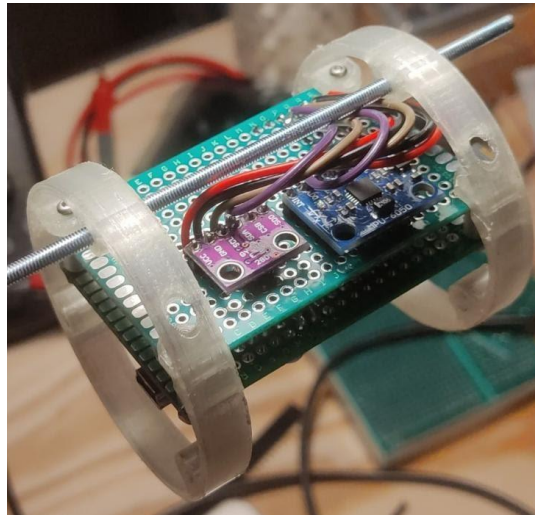


Figure 11 : Module expérience côté capteurs

Module séquenceur

Module composé de deux anneaux et d'une breadboard à pastilles contenant la carte Arduino servant principalement à contrôler l'ouverture du parachute en l'état de la fusée. La panneau de contrôle vient se fixer sur deux tiges filetées par dessus ce module et permet de contrôler l'état de la fusée ainsi que l'ouverture manuelle de la trappe à parachute.

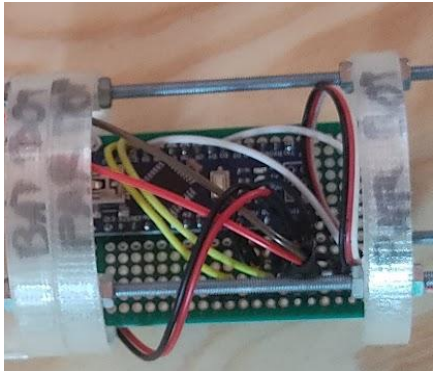


Figure 12 : Module séquenceur

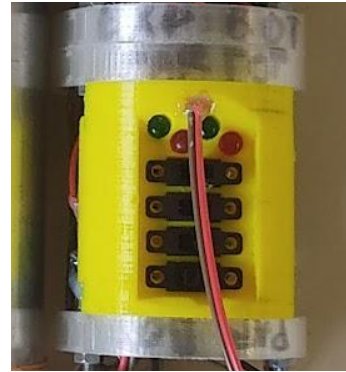


Figure 13 : Panneau de contrôle

Module servomoteur

Ce module est composé de deux anneaux spécialement conçus pour accueillir le servomoteur et pour permettre son réglage selon l'axe longitudinal de la fusée ainsi que son réglage pour fermer correctement la trappe (cf. figure 14).



Figure 14 : Module servomoteur

Partie "basse"

Corps de la fusée

Tube en aluminium de 2,5 mm d'épaisseur et de 60mm de diamètre externe composé de 3 trous d'un côté pour accueillir l'ogive, 3 autres trous de l'autre côté pour les ailerons et le compartiment moteur. 2 trappes ont été découpées à la meuleuse pour le parachute et pour le panneau de contrôle (cf. figure 15).



Figure 15 : Corps de la fusée attaché au compartiment moteur

Trappe à parachute

La trappe a dû être imprimée en deux pièces du fait de sa longueur importante. Pour solidifier l'ensemble après assemblage, il nous a été conseillé d'y ajouter 3 tiges en fibre de carbone. Celles-ci ont simplement été glissées dans des fentes prévues à cet effet et ont été collées grâce à de la colle époxy (cf. figure 16).



Figure 16 : Trappe à parachute

Ailerons et compartiment moteur

Les ailerons sont imprimés en 3D d'une seule pièce, en PLA. Il est pourvu de trois ailerons calés à 4° d'inclinaison, afin de donner à la fusée un mouvement de rotation autour de son axe. Ces ailerons sont composés à partir d'un profil NACA symétrique extrudé. L'idée originale était de nous servir de cette pièce en PLA comme support moteur. Cependant, des interrogations sur la capacité de la pièce à résister à la chaleur dégagée par le moteur. Nous avons donc isolé le moteur de la pièce avec des bagues en bois.

Après le vol, nous nous sommes rendus compte que la reprise de poussée par le bas n'était pas bien dimensionnée et que le moteur était en appui sur cette pièce. Aucun signe

de dégradation dû à la chaleur n'a été observé. Pour cette raison, nous pensons qu'il serait bon de réaliser des essais statiques au sol avec différents matériaux imprimables en 3D afin de déterminer si certains sont utilisables dans le contexte d'un projet C'Space, ce qui permettrait plus de liberté dans la forme des pièces pour les clubs.



Figure 16 : Version finale du compartiment moteur en des ailerons

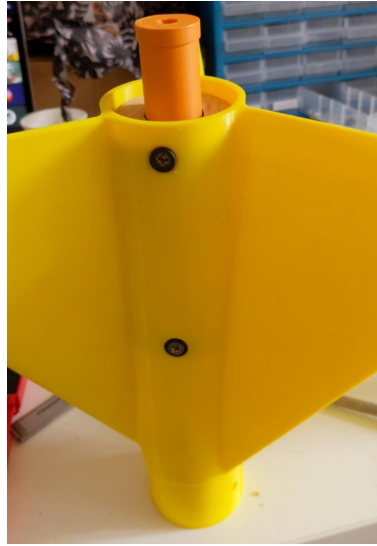


Figure 17 : Test de mise en place d'une reproduction d'un moteur



Figure 18 : Impression de la pièce - vue de profil d'un aileron



Figure 19 : Impression de la pièce - vue globale

Compte rendu du vol

Traitement de données

Malgré les tests effectués au sol, aucune donnée n'a été enregistrée sur la carte SD.

La source de ce problème proviendrait certainement du code de sauvegarde de données. En effet, par option de facilité et par manque de temps, nous avons utilisé des objets de la classe String. Cet outil, nous a permis de générer les chaînes de caractères, qui sont par la suite stockées dans un fichier format CSV. L'utilisation de cette classe pourrait causer ce que nous appelons la fragmentation de mémoire car celle-ci utilise l'allocation dynamique de la mémoire (car les chaînes de variables sont de différentes tailles).

Dans le cas où il y aurait des données sauvegardées, un filtre de Kalman aurait été implémenté à l'aide d'un code Python, celui-ci nous permettrait de fusionner les données des différents capteurs.

De plus, nous aurions souhaité vérifier les résultats obtenus avec les simulations Openrocket, en effet, nous y avons observé un lien de proportionnalité entre la vitesse de rotation et la vitesse axiale de la fusée, avec les données enregistrées par le MPU, cela aurait pu être démontré.

Déroulement du vol

Malgré une extraction tardive du parachute, le vol est nominal.

Depuis le sol, nous avons aperçu la trappe ouverte mais le parachute semble être resté bloqué dans sa case.

Comme nous n'avons pas pu récupérer les données du vol, nous ne pouvons pas le confirmer.

Nous émettons l'hypothèse que l'extraction tardive du parachute est due à la rotation de la fusée autour de son axe, qui aurait enroulé la corde du parachute autour du tube.

Si des vidéos du vol de bonne qualité sont disponibles, il devrait être possible de le confirmer.

Comme le système de récupération est un élément de sécurité, il paraît risqué de retenter l'expérience une seconde fois, mais s'il est impossible de le déterminer sur les vidéos, il peut s'agir de la seule méthode fiable pour confirmer l'origine du phénomène.

Retour sur expérience

Problèmes rencontrés lors du projet

L'écrasante majorité des problèmes rencontrés lors du projet sont dus à une mauvaise gestion du temps. Travailler dans l'urgence est certes motivant, mais nous fait faire de nombreuses erreurs.

Un exemple notable est l'accident mettant en jeu un fer à souder chaud et un parachute fragile en nylon, qui nous a forcés à le réparer avec une rustine.

Globalement, aucun problème rencontré au cours du projet ne s'est montré réellement bloquant. Nous nous sommes rendus compte, en revanche, que puisque la source de la majorité de nos difficultés venaient d'un manque de temps, travailler dans l'urgence ne fait qu'empirer la situation.

Nous nous sommes alors retrouvés dans un cercle vicieux qui nous a causé encore plus de difficultés. Fort heureusement, nous avons su les contourner ou les corriger, et ainsi mener à bien ce projet.

Amélioration à voir pour un futur projet

Vers la fin du projet, nous nous sommes posés la question suivante : "Si nous nous lançons à nouveau dans un projet de minifusée, que ferions-nous différemment ?".

Il est important pour nous d'apprendre de nos erreurs, donc nous poser cette question est primordial. Nous avons plusieurs éléments de réponse :

- Une meilleure gestion du temps. Nous avons eu beaucoup de difficultés avec ce point, majoritairement à cause de nos études qui nous demandent beaucoup de travail. Cependant, nous avons trop gaspillé du temps libre.
- Une électronique plus poussée. Nous avons émis le souhait de réaliser un circuit imprimé sur-mesure, ce qui aurait pu nous permettre d'éviter certaines difficultés lors de l'intégration finale avant le vol. Les cartes prototypes que nous avons
- Intégrer un système de télémétrie. Juste avant le vol, un système de télémétrie nous aurait permis de nous rendre compte du bug d'enregistrement et potentiellement de repousser le lancement en conséquence
- Prévoir des mises à jour du code alors que la fusée est assemblée. Nous avons dû passer un câble USB à travers toute l'électronique afin de mettre à jour l'une des cartes Arduino. Cette opération s'est révélée délicate et passer le câble à travers toutes les connexions fragiles des cartes électroniques est risqué