

Projet d'études N° 31

Noms des élèves :

ALLEON Julien
CHARREAUX Guillaume
JOUHET Guilhem
MANCY Damien
YANNE Louis

Commanditaire :

Centrale Lyon Cosmos (CLC)

Tuteur scientifique :

M.NAVARRO David

Conseiller en communication :

M.BEAUBOIS Vincent

Conseiller en gestion de projet :

Mme CONTRAIRES Élise

Département d'accueil :

EEA

Date du rapport :

09/06/2017

**Mini-Fusée : structuration interne et
communication sans fil à longue portée**



1 Résumé

Français

Ce rapport présente les activités et les résultats obtenus lors de la réalisation du PE 31 de l'année scolaire 2016/2017. Il consiste à élaborer et fabriquer une minifusée, avec son électronique embarquée, qui devra être lancée lors d'un séminaire organisé par Planète Sciences. Celle-ci devra y effectuer un vol nominal, par opposition à un vol balistique, où la fusée n'est pas récupérée intacte. Cette électronique embarquée a aussi pour but de rendre cette dernière capable de réaliser des mesures, puis de les communiquer à la base de lancement en communication sans fil. Ce projet, commandité par le CLC et encadré par M.NAVARRO du département EEA, est réalisé par 5 étudiants de l'école. Depuis plusieurs années, le CLC propose des PE dans lesquels les étudiants construisent selon les mêmes standards des minifusées, ce qui permet à chaque nouvelle équipe de ne pas partir de zéro.

Ainsi, ce projet vise à la construction d'une minifusée standardisée sur laquelle sera pour la première fois embarqué un module de communication LoRa. Le fait de pouvoir se reposer sur les acquis du CLC permet de dégager le temps nécessaire à l'implantation du module de communication LoRa. Le projet étant par ailleurs pluridisciplinaire, les différents pôles ont été répartis entre les élèves afin que chacun d'entre eux puisse se former puis se concentrer sur des domaines précis. Grâce à cette répartition par pôle et à la récupération des travaux précédents, le fuselage et la structuration interne de la fusée ont été réalisés, les cinq cartes électroniques embarquées ont été dessinées, usinées, soudées, puis programmées.

Ce rapport s'attachera à détailler l'ensemble des étapes nécessaires à la réalisation de la minifusée, aux problèmes que l'équipe a rencontrés, aux solutions qu'elle y a apportées, et ceci pour l'ensemble des pôles mécanique, électronique, et informatique.

English

This report details the activities and results achieved through the making of SP 31. The goal is the conception and the manufacturing of a small-scale rocket with its embedded electronics. That rocket should be launched during a meeting, the C'space, and would perform a nominal flight. The electronic part would also realise measures and transmit them to the launching pad through wireless communication. This CLC-sponsored project was done by 5 students from the school and was supervised by Mr.NAVARRO from the EEA department. Because the CLC has been offering SP wherein students build mini-rockets with the same standards for several years, every new team doesn't need to start from scratch.

The project aims thus to the making of a standardised mini-rocket, which will for the first time embed a Long Range communication device : LoRa. The possibility to rely on the CLC's legacy vacates the time needed to implant the LoRa-communication device. Furthermore, the project being multidisciplinary, all students were given different fields to work on so as to prepare them and make them focus on precise objectives. Thanks to this distribution and the data retrieval of past teams' works, the fuselage and intern structuring were crafted, the five embedded electronic cards were designed, processed, welded and programmed.

The report shall describe the necessary steps to the building of the mini-rocket as a whole, the issues the team met, the solutions it brought in response, and all that for the mechanical, electronics and computing fields.

2 Remerciements

Avant de détailler notre projet, nous tenons à remercier M. NAVARRO David sans qui ce projet n'aurait pas pu être mené à bien. Tuteur scientifique de plusieurs projets semblables au nôtre depuis quelques années, ses conseils et son expérience nous ont été d'un grand secours. Sans lui, notre projet n'aurait pas été possible.

M. COCHARD Julien, ancien du CLC, nous a également été d'une grande aide en ce qui concerne la gestion du projet. Nous souhaitons le remercier pour le temps qu'il nous a consacré, et pour le recul qu'il nous a permis de prendre sur l'avancement de notre projet tout au long de l'année.

Nous remercions également l'ensemble de l'équipe pédagogique qui nous a accompagnés au cours de cette année. Mme CONTRAIRES Élise, M. BEAUBOIS Vincent, respectivement nos conseillers en gestion de projet et en communication, nous ont donné les outils nécessaires d'une part à la gestion d'un projet ambitieux, qui, comme le nôtre, ne saurait être mené à bien sans ces outils, et d'autre part à présenter correctement les résultats, nécessairement complexes, d'un tel projet. Nous les remercions aussi pour les conseils et l'aide qu'ils nous ont fournis lors des rédactions des différents livrables.

Nous tenons à remercier l'équipe du Fablab de l'École Centrale de Lyon pour leur aide et la mise à notre disposition de leurs outils pendant la durée de notre projet. Nous remercions également l'équipe de Planète Sciences, organisatrice du C'space, pour s'être tenu à notre disposition toute l'année, et pour le temps qu'ils ont pu consacrer à l'organisation des RCE et du C'space, des occasions uniques pour rencontrer les étudiants d'autres écoles, également passionnées d'aéronautique et d'aérospatiale. C'est une chance pour nous de pouvoir ainsi découvrir le milieu aérospatial.

Nous remercions l'entreprise lyonnaise CIRLY, qui nous fournit gratuitement les circuits imprimés dont nous avons besoin pour réaliser notre électronique embarquée. Enfin, nous souhaitons remercier l'École Centrale de Lyon, et l'équipe du Bonus Qualité Projet (BQP) qui la représente, d'avoir financé notre projet de manière à nous permettre de participer au C'space, vitrine nationale pour l'école, et occasion unique pour nous de nous forger une première expérience professionnelle dans l'aérospatiale.

Table des matières

1	Résumé	2
2	Remerciements	3
3	Introduction	7
4	Contexte du projet	9
4.1	Encadrement du projet	9
4.1.1	Planète Sciences	9
4.1.2	C'Space	9
4.1.3	CNES	10
4.1.4	Centrale Lyon Cosmos	10
4.1.5	Équipe du projet	10
4.1.6	Encadrement au sein de Centrale	11
4.2	État de l'art	11
4.3	Objectifs du Projet d'Étude	12
4.3.1	Position du problème	12
4.3.2	Cahier des charges	13
5	Planification des tâches	15
5.1	Architecture de la fusée	15
5.2	Gestion de projet	15
5.3	Rencontres Club Espace	18
6	Réalisation du projet	19
6.1	Partie mécanique	19
6.1.1	Présentation générale de la fusée	19
6.1.2	Conception technique de la fusée	19
6.1.3	Propulsion et stabilité	23
6.2	Partie électronique	30
6.2.1	Présentation générale de la case électronique	30
6.2.2	Présentation des cartes	30
6.2.3	Réalisation des cartes	35
6.2.4	Base au sol	37
6.3	Partie programmation	37
6.3.1	La carte séquenceur	38
6.3.2	Carte capteur : module de la centrale inertielle	40
6.3.3	Carte capteur : télémétrie émettrice	44
6.3.4	Carte télémétrie réceptrice	45
7	Bilans	46
7.1	Bilans du projet par pôles	46
7.1.1	Partie mécanique : Alleon Julien	46
7.1.2	Partie électronique : CHARREAUX Guillaume	46
7.1.3	Partie programmation : JOUHET Guilhem, MANCY Damien, YANNE Louis	46
7.1.4	Partie gestion de projet : JOUHET Guilhem	47
7.2	Perspectives	48

8	Conclusion	49
9	Bibliographie	50
10	Annexe : code en C des cartes programmables	52
10.1	Carte séquenceur	52
10.2	Carte capteur : centrale inertielle	57
10.3	Carte capteur : communication sans fil à longue portée	63
10.3.1	fichier main.c	63
10.3.2	fichier get_param.c	66
10.3.3	fichier default_param.c	71
10.3.4	fichier communication.c	73
10.3.5	fichier set_param.c	75
10.3.6	Check-List de rapport de Projet d'Etude	78

Table des figures

1	Modélisation Catia de la fusée Balistey, qui sera construite pour être lancée lors du C'space	8
2	Case électronique programmable embarquée dans la fusée	8
3	Logotype de Planète Sciences	9
4	Bannière du C'Space 2017 [2]	9
5	Logotype du CNES	10
6	logo du CLC	10
7	Equipe du projet	11
8	Encadrement du projet	11
9	Différentes phases du vol de la fusée [10]	12
10	Diagramme de GANTT initial	16
11	Diagramme de GANTT final	17
12	Trésorerie du projet (en €)	18
13	Carte mixte	19
14	Modélisation Catia de la fusée en vue normal et éclatée	20
15	Corps de la fusée	20
16	Modélisation du module de propulsion	21
17	Modélisation du module parachute	22
18	État de la bague mixte du PE précédent après lancement de la fusée	23
19	Simulation de comportement à un effort de l'ancienne bague (à gauche) et de la nouvelle (à droite)	23
20	Modélisation du module cartes	24
21	Modélisation de la coiffe sous Catia	24
22	Propulseur Cariacou	25
23	Courbe de poussée du propulseur Cariacou	25
24	Dimensions des ailerons	27
25	Exemple de bagues	27
26	Résultat du calcul de centre de masse par Catia	28
27	Validation de la stabilité par Stabtraj	29
28	Anticipation de trajectoire par Trajecto	29
29	Schéma de répartition des cartes dans la fusée	31
30	Cartes électroniques dans la fusée	31
31	Carte capteur mère, entre deux cartes capteurs filles	33

32	Carte fille supportant la centrale inertielle	33
33	Schéma de la centrale inertielle avec les trois axes [1]	34
34	Carte mixte, réalisée par l'autre projet	35
35	Schéma électrique et typon de la carte centrale inertielle	36
36	Typon de la nouvelle carte mixte	36
37	Récepteur LoRa, utilisé au sol avec un ordinateur	37
38	Logotypes de Microchip (cartes électroniques) et PIC C Compiler (logiciel de programmation)	38
39	Organigramme informatique de la carte séquenceur	39
40	Schéma représentants les différents repères	41
41	Figures de changement de base	42
42	À gauche : une fusée dans sa phase de montée. À droite : fusée à l'apogée	42
43	Organigramme de la carte capteur intégrant la centrale inertielle	43
44	Organigramme de la carte télémétrie émettrice	45

Liste des tableaux

1	Constituants de la minifusée	30
2	Caractéristiques et dimensionnement de la carte alimentation, les dimensionnements sont d'après le PE 25 2016	32
3	Caractéristiques et dimensionnement de la carte séquenceur, les dimensionnements sont d'après le PE 25 2016	32

3 Introduction

Le projet 31 de l'année 2016-2017, intitulé minifusée : structuration interne et communication sans fil, s'appuie sur les retours d'expériences des précédentes minifusées réalisées au cours des PE des années antérieures commandités par le CLC. Au fil des années, celles-ci se sont progressivement complexifiées, embarquant de nouvelles fonctionnalités et améliorations. Aidé par le contenu de l'ensemble des rapports finaux de ces projets, une minifusée, qui doit participer à la campagne de lancement du C'space de Planète Sciences, est confectionnée. Pour cela, la fusée à réaliser, nommée Balistey, embarquera un ensemble de cartes électroniques qui seront programmées, lui permettant de déployer un parachute à l'apogée, et d'effectuer des mesures en vol. En effet, avant de pouvoir être lancé au C'space, toute fusée doit prouver qu'elle réalisera un vol nominal (récupération de la fusée intacte), et qu'elle embarquera des expériences d'intérêt scientifique.

Les objectifs scientifiques du projet ont été déterminés à partir des rapports et des bilans des précédents PE du CLC qui ont fabriqué des minifusées. Par exemple, une minifusée de l'année dernière a embarqué une expérience permettant d'évaluer le dégagement de chaleur créé par le propulseur. Cette dernière a permis de mettre en évidence qu'il est possible de placer le parachute juste au-dessus du propulseur, ce qui permet de simplifier l'architecture interne de la fusée, en diminuant le nombre de cartes électroniques embarquées. Aussi, les modules de communication sans fil jusqu'ici utilisés ne permettaient pas de communiquer avec la fusée durant tout le vol. Par ailleurs, une bague mixte, utilisée pour la première fois l'an dernier sur une minifusée, devait permettre d'incorporer le système d'allumage de la fusée en surface du fuselage, ce qu'elle fit avec succès. Cependant, celle-ci ne s'est pas révélée assez solide et s'est brisée pendant le vol.

Ainsi, la plupart des éléments mécaniques de la fusée étant déjà dimensionnés grâce à ces précédents PE, l'accent est ici mis sur la simplification de la structure interne de la fusée, liée au déplacement de la position du parachute dans le corps du parachute, sur la mise en place d'un module de communication sans fil à longue portée, et sur l'amélioration de la bague mixte. En ce qui concerne la communication à longue portée, c'est une solution proposée par l'internet des objets qui sera utilisée, avec un module LoRa. La technologie LoRa (pour Long Range) a été pressentie pour être intégrée dans la fusée pour son faible coût, sa faible consommation énergétique, son faible poids, sa longue portée et son débit. La résistance de la bague mixte sera alors étudiée pour être renforcée. Pour se familiariser avec les livrables du projet, le lecteur trouvera en Figure 1 et 2 une modélisation Catia de la fusée, Balistey, qui est fabriquée lors du projet, et la case électronique embarquée qui est programmée. Le groupe du projet dispose des archives (rapports de projets, quelques documents techniques) des précédents PE sur le site internet du CLC, auquel il a accès après s'y être inscrit grâce à un moteur de recherche intégré.

La suite de ce rapport présente tout d'abord en détail le contexte du PE 31 de l'année scolaire 2016/2017, puis la planification des tâches telle qu'elle a été faite en octobre, et telle qu'elle a été modifiée par la suite, et enfin la réalisation de la minifusée, en décrivant les problèmes rencontrés, les solutions qui y ont été apportées, et les résultats obtenus. Enfin, chaque pôle expose le bilan qu'il tire de cette expérience, et une perspective du projet est proposée.

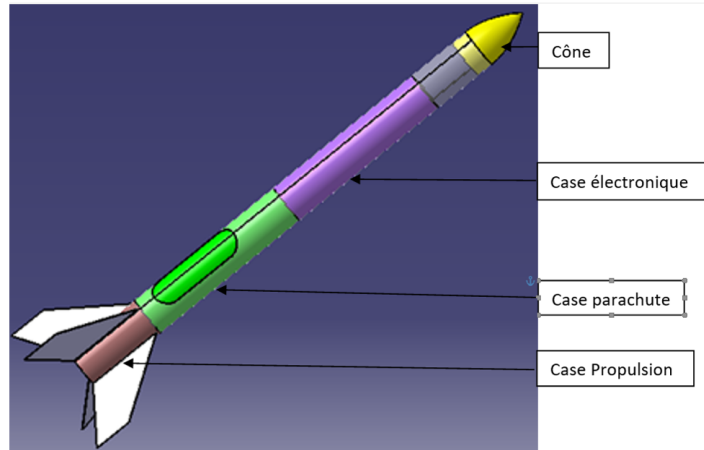


FIGURE 1: Modélisation Catia de la fusée Balistey, qui sera construite pour être lancée lors du C'space

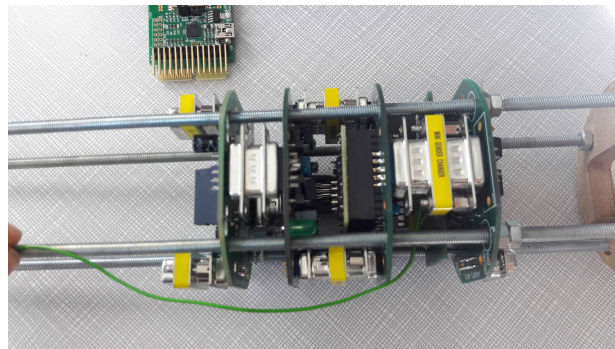


FIGURE 2: Case électronique programmable embarquée dans la fusée

4 Contexte du projet

4.1 Encadrement du projet

4.1.1 Planète Sciences

Les membres du projet sont en lien avec l'association Planète Sciences qui s'occupe chaque année du suivi des projets étudiants semblables à celui du PE 31 à travers la France. Ils sont également en charge de l'organisation du C'space, la campagne nationale de lancement de fusées de fin d'année. Il s'agit d'une association dont l'objectif est d'encourager l'intérêt et la pratique scientifique des jeunes. Le C'space n'est que l'une des nombreuses activités qu'elle propose, cette association étant également impliquée dans l'organisation des Trophées de robotique ou de la Nuit des étoiles.

Au cours de l'année, Planète Sciences a accompagné l'équipe en organisant trois Rencontres de Clubs Espace (RCE) afin de la guider et de lui fournir les informations essentielles en vue de la campagne de lancement.



FIGURE 3: Logotype de Planète Sciences

4.1.2 C'Space

Dans le but d'expérimenter les améliorations apportées aux fusées du CLC, le projet sera représenté à une campagne de lancement nationale : le C'Space. Celui-ci se déroule du 15 juillet au 22 juillet au camp de l'armée de Terre de Gers, à proximité de Tarbes (65). Organisé par Planète Sciences en partenariat avec le CNES, le C'Space constitue l'aboutissement de ce projet. Balistey y sera lancée, et devra effectuer un vol nominal, composé d'un décollage durant lequel la fusée est propulsée, suivi d'une phase balistique en attente de l'atterrissage, au cours duquel la fusée est freinée jusqu'à 5 m/s en attendant de retomber au sol.



FIGURE 4: Bannière du C'Space 2017 [2]

4.1.3 CNES

Le Centre national d'études spatiales (CNES) est coorganisateur du C'space avec Planète Sciences. Agence spatiale la plus importante de l'Union européenne par l'ampleur de ses financements, cet établissement public est en charge du programme spatial français. Dans notre cas, il fournit les propulseurs à poudre nécessaires au lancement des fusées étudiantes, qui restent très dangereux à manipuler. [3]



FIGURE 5: Logotype du CNES

4.1.4 Centrale Lyon Cosmos

Association Centralienne reconnue d'Intérêt Pédagogique (ACIP), elle fait le lien entre les institutions précédentes et les élèves de l'École, en leur proposant des PE et des PA liés à des thématiques aérospatiales. Elle commandite cette année deux projets de minifusées semblables, les PE 31 et PE 32. Le PE 31 est ici même décrit, le PE 32 doit également procéder au lancement d'une fusée au C'space et y détecter l'apogée du vol à l'aide d'une sonde Pitot.



FIGURE 6: logo du CLC

4.1.5 Équipe du projet

Le projet est mené par cinq étudiants centraliens au cours de l'année scolaire. Afin de former ce groupe, le CLC a procédé à une sélection sur la base d'une lettre de motivation. Une fois le groupe arrêté et affecté à ce projet, il a été réuni et découpé en trois domaines techniques pour que ses membres puissent se répartir le travail. Ils ont ainsi été répartis sur les parties électronique, mécanique et informatique. Après avoir nommé JOUHET Guilhem chef de projet, l'équipe a été répartie comme indiqué en figure 7 dans

les différents pôles. MANCY Damien, YANNE Louis et JOUHET Guilhem se chargent alors de la programmation de l'ensemble des cartes électroniques embarquées, qui constitue la partie la plus dense du projet. Ces cartes électroniques, et le corps de la fusée sont respectivement fabriqués par CHARREAUX Guillaume et ALLEON Julien. Les trois pôles précédents sont coordonnés tout au long de l'année par JOUHET Guilhem afin d'optimiser le temps disponible.



FIGURE 7: Equipe du projet

4.1.6 Encadrement au sein de Centrale

Régulièrement, l'équipe rencontre son tuteur scientifique, M. NAVARRO David, du département EEA, les anciens et les membres du bureau du CLC afin de partager l'avancement du projet, les problèmes rencontrés, et de discuter des solutions qui peuvent y être apportées.

En plus du tuteur scientifique, l'équipe est encadrée par M. BEAUBOIS Vincent et Mme CONTRAIRES Élise de l'École Centrale de Lyon qui lui donnent au cours de l'année les éléments nécessaire à la gestion du projet, et à la communication interne et externe.



FIGURE 8: Encadrement du projet

4.2 État de l'art

Comme il a été mentionné plus haut, les objectifs du projet ont été fixés à partir des résultats des précédents PE. L'équipe dispose des rapports finaux de ces derniers, qui lui servent à récupérer des brides de codes nécessaires à la programmation des cartes

électroniques, et également avoir accès aux fichiers de Conception Assistée par Ordinateur (CAO) des bagues en bois qui servent à rigidifier le corps de la fusée. Le pôle gestion de projet a également pu planifier les tâches dès le mois d'octobre à partir des diagrammes de GANTT de ces précédents PE.

4.3 Objectifs du Projet d'Étude

4.3.1 Position du problème

Outre sa conception, nous devons aussi faire en sorte que la minifusée effectue un vol nominal, c'est-à-dire un vol qui se termine par la récupération intacte de la fusée, par opposition à un vol balistique¹. Le vol de la fusée présente différentes phases (également résumés sur la Figure 9) :

La propulsion : elle débute à partir de la mise à feu du propulseur lorsque la fusée se trouve encore sur la rampe de lancement jusqu'au moment où tout le carburant a été consommé.

La phase balistique : au cours de cette étape, qui succède immédiatement la phase précédente, la fusée s'apparente à un corps libre soumis uniquement à son poids et aux forces de frottement.

La récupération : après avoir atteint l'apogée, le parachute se déploie pour faire atterrir en douceur la minifusée. Sa vitesse diminue alors fortement.

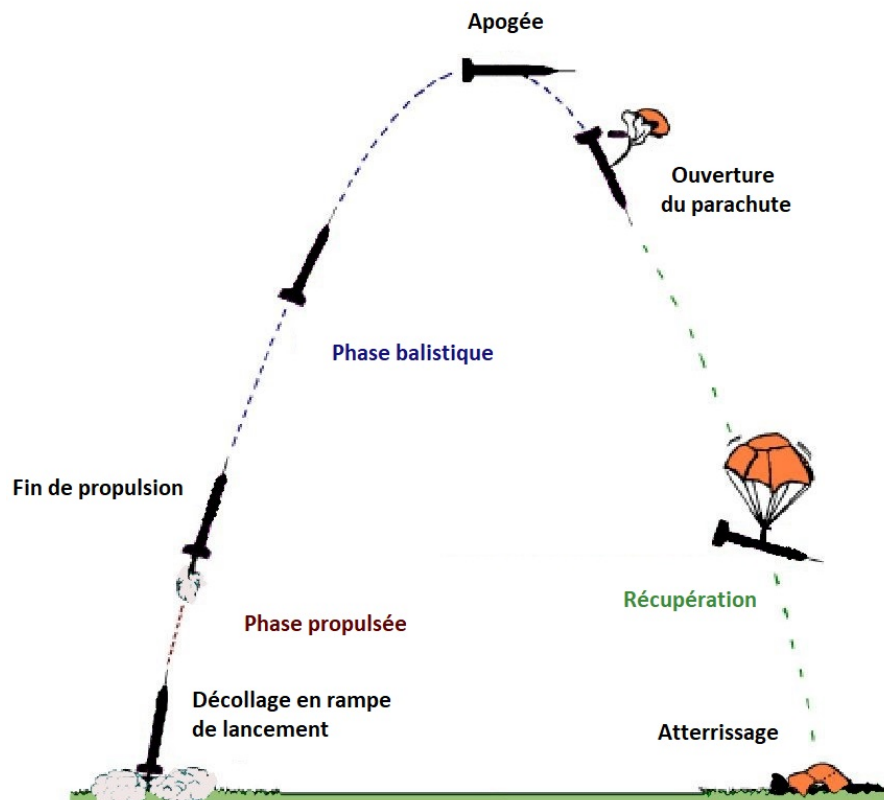


FIGURE 9: Différentes phases du vol de la fusée [10]

1. Retombée de la fusée due au non-fonctionnement du système de récupération (ici un parachute).

À cet objectif primaire de réussir un vol nominal, nécessaire à tout bon déroulement d'un vol, un certain nombre d'exigences sont propres à l'intitulé du PE.

- La communication des données doit se faire sans fil à longue portée par un module LoRa.
- Une case supplémentaire appelée bague mixte doit comporter les afficheurs, tels que les LED, et les actionneurs, c'est-à-dire les interrupteurs.
- Le parachute peut être rapproché de la case propulseur. Les tests menés l'an passé ont mis en évidence que la température dégagée lors de la propulsion n'est pas suffisante pour endommager le parachute.

4.3.2 Cahier des charges

Afin de participer à la campagne de lancement du C'space, la minifusée doit satisfaire le cahier des charges fourni par Planète Sciences qui se rapporte essentiellement à des consignes de sécurité. En effet, le site de lancement étant un camp militaire et les propulseurs utilisés dangereux à manipuler, tout projet doit respecter certaines normes très strictes. De plus, le projet se doit d'atteindre les objectifs fixés par le CLC, le cahier des charges précédent a donc été complété afin d'y faire figurer ces objectifs.

Le cahier des charges retenu est alors le suivant :

Fonctions principales

- FP1** : La minifusée doit réussir un vol complet, du lancement à l'atterrissage.
- FP2** : Mesurer l'accélération de la minifusée et détecter l'apogée pour déployer le parachute.
- FP3** : Transmission à longue portée (LoRa) des données et visualisation de la trajectoire au sol.
- FP4** : Participer à la campagne de lancement à Tarbes.
- FP5** : La fusée mesure 1m et contient une bague mixte.
- FP6** : Le parachute doit se situer en bas de fusée, au-dessus de la case propulseur.

Généralités

- GN1** : La fusée ne doit présenter aucun danger pour les personnes ou l'environnement. Tout élément éjecté doit être solidement relié à la fusée ou ralenti lors de sa chute dans les mêmes conditions que pour la fusée.
- GN2** : Seule l'utilisation des inflammateurs fournis par Planète Sciences est possible. Aucune modification ou ajout de matière inflammable n'est autorisé.
- GN3** : La fusée doit-être compatible avec la rampe de lancement Idefix lors de la campagne de lancement
- GN4** : L'autonomie de l'alimentation doit-être d'au moins 15 minutes due aux procédures d'avant-vol. Il est nécessaire d'avoir un interrupteur marche/arrêt de l'alimentation.
- GN5** : La fusée doit disposer d'indicateurs clairs pour permettre de savoir à tout moment dans quel état elle se trouve (marche/ arrêt, position sécurité, position vol, etc.).

Conditions de vol

- VL1** : Le groupe de PE doit fournir une chronologie au responsable du lancement.
- VL2** : Le propulseur doit entraîner la fusée. Il doit pouvoir être monté ou démonté rapidement, être centré sur le diamètre, être maintenu dans l'axe de la fusée et s'appuyer sur une plaque de poussée plane.
- VL3** : L'axe longitudinal de chaque aileron doit être parallèle à l'axe longitudinal de la fusée. Les ailerons doivent supporter les fortes contraintes aérodynamiques du vol.
- VL4** : La fusée doit avoir un vol stable. La vitesse nominale de sortie de rampe doit-être d'au moins $18m.s^{-1}$.
- VL5** : La Flèche inférieure à 1% (entre la plaque de poussée propulseuse et le haut de l'ogive).
- VL6** : Tous les éléments de la fusée doivent rester fixés durant la durée du vol.

Récupération

- RC1** : La fusée doit être munie d'un système de récupération lui permettant de rejoindre le sol à une vitesse comprise entre 5 et $15m.s^{-1}$.
- RC2** : La fusée doit être munie d'un système réglable en temps permettant la mise en œuvre du ralentisseur à l'apogée de la trajectoire.
- RC3** : Le ralentisseur et ses fixations doivent résister au choc lors de l'ouverture du parachute.
- RC4** : La trappe latérale ne doit pas s'ouvrir sans être commandée, mais doit s'ouvrir malgré les contraintes du vol.
- RC5** : La fusée doit retomber dans le gabarit de lancements défini.

5 Planification des tâches

5.1 Architecture de la fusée

5.2 Gestion de projet

La gestion de projet doit être un effort soutenu et continu sur toute la durée de l'étude. En effet, la répartition de notre équipe en différents pôles a pour conséquence directe une perte de visibilité sur l'avancement réel du projet. De plus, les pôles sont interdépendants, ce qui implique que tout retard sur un d'entre eux entraîne un retard cumulable sur les autres. Par exemple, il est impossible de coder les cartes électroniques tant que les composants n'ont pas été commandés, et soudés entre eux. Ainsi, dès le début de notre projet, il a été soigneusement identifié quelles tâches devaient être réalisées prioritairement. En se basant sur le rapport du PE 25 de 2016 [PE25], le déroulement du projet a été programmé, et les objectifs à court et long termes fixés. Cette planification est décrite dans un premier diagramme de GANTT, donné en Figure 10.

Comme la gestion de projet est menée tout au long de l'année, le diagramme de GANTT a été modifié au cours de la réalisation des tâches. Il est visible sur le diagramme final (voir Figure 11) que nombre de tâches de la partie programmation ont été décalées vers la fin de l'année, ce qui a alerté dès le mois d'avril l'équipe du projet. Cependant, nombre de problèmes ont empêché de rattraper ce retard, qui a tout juste été stabilisé. En effet, l'équipe a attendu deux mois la carte mixte qui était alors en test chez le PE 32 pour finir le programme de la carte séquenceur, le PE 68 a utilisé pendant le mois de mai un des deux ordinateurs dont l'équipe avait alors besoin pour coder en parallèle la carte capteur et la carte de communication sans fil, et un module LoRa ayant grillé en février n'a pas encore pu être remplacé ce qui a posé de gros problèmes pour coder correctement notre système de communication à longue portée. Bien que tous les programmes ne soient pas encore opérationnels, les programmes vitaux le sont parfaitement, et l'équipe pourra continuer à travailler en juin pour présenter au C'space une fusée totalement fonctionnelle.

Tous les livrables sont également préparés puis rédigés avec l'aide de tous les membres du projet. Il faut, pour chaque livrable, qu'il soit pour l'École Centrale de Lyon ou Planète Sciences, analyser les attentes, préparer les sections à rédiger, rédiger les parties communes, pour que chaque pôle puisse ensuite rédiger sa partie dans de bonnes conditions. Pour ce faire, les membres du projet ont tous appris à utiliser un utilitaire L^AT_EX collaboratif.

Un autre domaine de la gestion de projet est la comptabilité. Chaque PE a un budget initial de 300€. D'après les précédents rapports de PE de minifusée, c'est approximativement le coût correspondant à la réalisation d'une minifusée. Pour participer au C'space, il est alors nécessaire de monter un dossier de candidature au BQP pour financer le déplacement et les frais d'inscription. Le dossier a été accepté malgré des réticences du jury à financer deux PE de minifusées, et le projet est financé par le BQP à hauteur de 700€. Le projet est également soutenu par l'entreprise CIRLY, qui fabrique les cartes électroniques embarquées. Cette entreprise fournit alors gratuitement les cartes au projet.

Ensuite, chaque commande, chaque ressource est comptabilisée pour présenter un budget valide en fin de projet. La trésorerie est alors donnée en Figure 12 et montre un résultat positif, à 148€. Cette trésorerie est prévisionnelle, l'inscription au C'space a été budgétisée, mais son montant définitif n'est pas encore connu.



FIGURE 10: Diagramme de GANTT initial



FIGURE 11: Diagramme de GANTT final

	Crédit	Débit
Dépenses réalisées		
Budget PE	300	
BQP	700	
Partenariat avec CIRLY	250	
Cartes électroniques standart CIRLY		350
Batterie LiPo		7,32
Composants électroniques		100
Carton (fuselage fusée)		2,5
Bois Medium (structuration interne)		12
Cartes filles + modules LoRa		70
Divers éléments structurels		30
Propulseur		Planète Sciences (0)
Parachute et servomoteur		50
RCE		Planète Sciences (0)
Bilan provisoire		
Crédit	1250	621,82
Participation au C'space		
Décoration fusée (peinture, sponsors)		30
Inscription au C'space		270
Déplacement au C'space		180
Total du Projet		
Totaux	1250	1101,82
Budget final	148,18	

FIGURE 12: Trésorerie du projet (en €)

5.3 Rencontres Club Espace

Le projet a participé aux deux dernières des trois RCE qui ont été organisées par Planète Sciences, et qui se déroulaient à Paris en week-end. Elles se sont bien passées : la première a permis une prise de contact entre le projet et les bénévoles de Planète Sciences, tandis que la seconde a approuvé la participation de Balistey à la campagne de lancement du C'space. Les frais de déplacement à Paris ont entièrement été pris en charge par Planète Sciences.

6 Réalisation du projet

6.1 Partie mécanique

6.1.1 Présentation générale de la fusée

En s'inscrivant dans la lignée des PE minifusée des années précédentes et notamment du PE 25 de l'année dernière, le premier objectif du projet est de simplifier et d'alléger au maximum la structure de la fusée. Comme l'an dernier, cette décision mène à la conception d'une bague dite mixte (car une carte électronique y est prise en sandwich entre deux demies bagues de maintien en bois). Cette dernière a un objectif bien précis. Pour pouvoir communiquer avec la fusée sur le pas de tir, il faut placer des composants (diodes, interrupteurs et prise Jack) à la surface du corps de la fusée. Le but étant de ne pas percer le corps de la fusée, la bague doit donc pouvoir intégrer cette carte mixte, permettre aux composants qui le nécessitent d'affleurer à la surface, et de connecter la carte au reste des cartes électroniques.

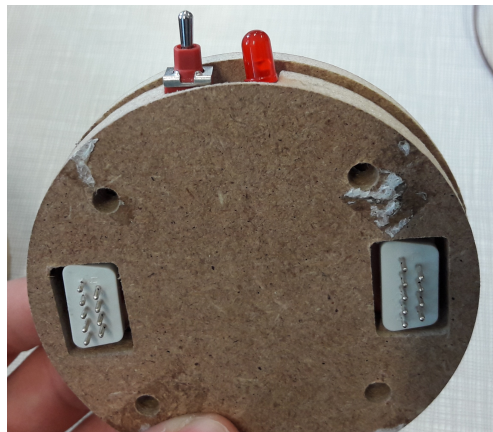


FIGURE 13: Carte mixte

6.1.2 Conception technique de la fusée

Structure générale

La fusée est divisée en 5 sous modules :

- Le corps de la fusée (qui correspond à la peau en carton)
- Le module propulseur (au niveau des ailerons)
- Le module parachute
- Le module cartes
- La coiffe

Le corps de la fusée :

La peau, avec la bague mixte, les ailerons et la coiffe, constitue l'ensemble de l'extérieur de la fusée. Le corps de la fusée est formé d'un tube de carton de 80 mm de diamètre intérieur et de 1 mm d'épaisseur. Ce tube est devenu depuis quelques années et quelques PE un standard pour les minifusées du CLC. Il permet de donner une bonne robustesse aux minifusées tout en garantissant leur légèreté, tout cela à moindre coût. Enfin, son diamètre de 80 mm permet d'y insérer les cartes électroniques standardisées. Les tubes

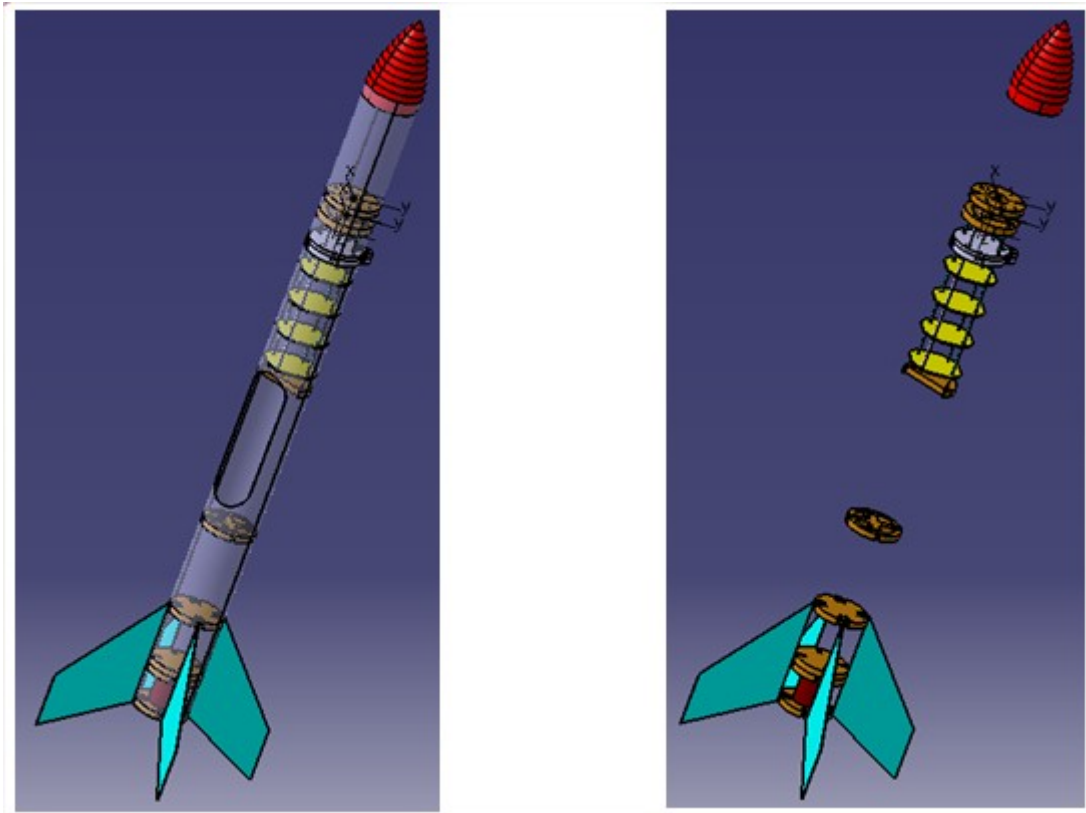


FIGURE 14: Modélisation Catia de la fusée en vue normal et éclatée

commandés sont ensuite partagés entre les deux groupes des PE 31 et 32 avant d'être découpés et percés de manière à y boulonner les bagues, la bague mixte, les ailerons et les différents modules.



FIGURE 15: Corps de la fusée

Le module propulseur :

Ce module possède les mêmes caractéristiques que les minifusées précédentes (notamment les PEs minifusée de 2015 et 2016). Cette similarité s'explique grâce aux objectifs remplis par ce module qui sont les mêmes depuis quelques années, notamment pour la partie propulsion et la partie stabilité de la fusée (qui correspond aux ailerons). Les bagues constituant le module ont plusieurs utilités :

- Contenir le propulseur dont les dimensions sont fixées par le CNES
- Transmettre la force de propulsion du propulseur à la fusée en y étant attaché par l'intermédiaire de boulons
- Fixer les ailerons à la peau de la fusée grâce à un système de vis et d'équerres

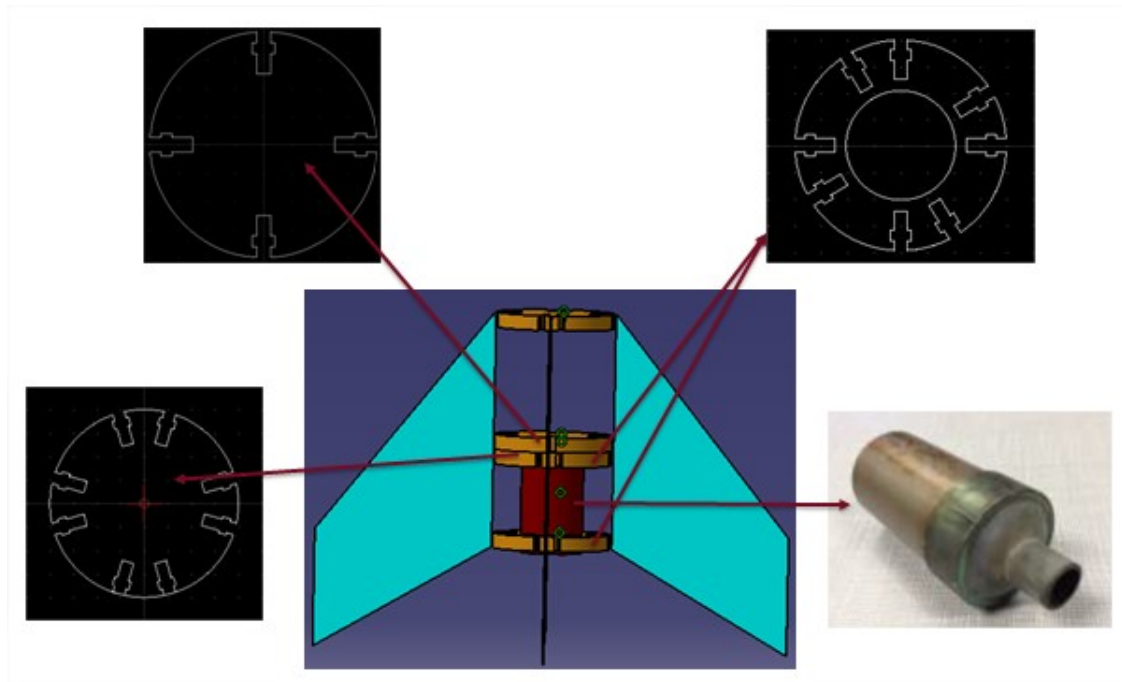


FIGURE 16: Modélisation du module de propulsion

Le module parachute :

Une des innovations de cette année est de déplacer le parachute vers le bas de la fusée de manière à ne plus devoir utiliser deux cases électroniques, qui étaient séparées par le parachute et nécessitaient de communiquer sans fil à l'intérieur de la fusée pour le PE précédent. Le parachute est donc placé vers le milieu de la fusée et non pas juste au-dessus du module de propulsion pour des raisons de stabilité en vol, qui est liée à la position du centre de masse.

L'objectif du module de parachute est simple : permettre l'éjection du parachute pour pouvoir faire atterrir la fusée. Le parachute de l'an dernier qui est suffisant pour ralentir la chute de la minifusée a été logiquement conservé. Le servomoteur est commandé directement par le module carte décrit ci-dessous. Il bloque la trappe de parachute lors du vol, puis tourne d'un quart de tour à l'apogée, permettant à la trappe d'être poussée vers l'extérieur par les ressorts introduits dans la bague servomoteur et le vent relatif créé par le déplacement de la fusée.

La trappe arrachée de la peau de la fusée emporte avec elle le parachute qui ralentira la fusée jusqu'à la vitesse imposée par le cahier des charges (entre 5 m/s et 15 m/s)

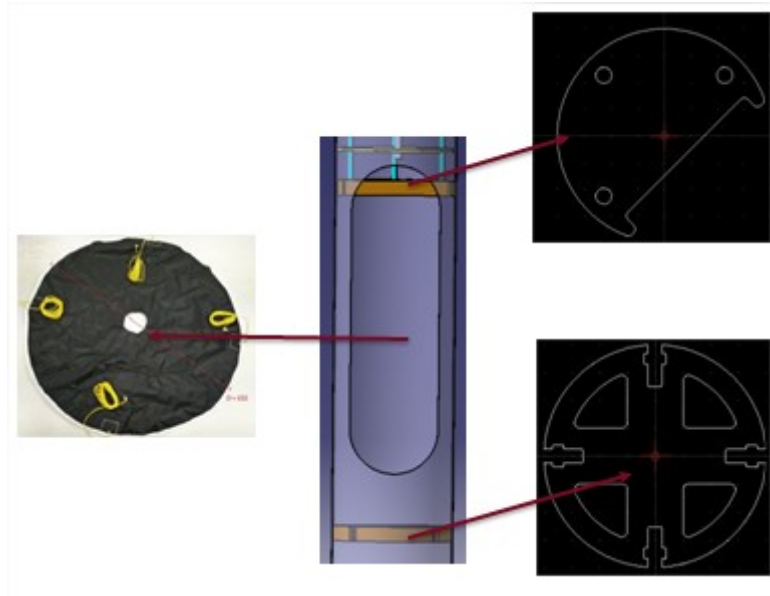


FIGURE 17: Modélisation du module parachute

Le module cartes :

Grâce au déplacement du module parachute vers le bas de la fusée, les cartes se trouvent désormais toutes au même endroit de la fusée, ce qui permet, entre autres, de simplifier la structure interne de la fusée, mais également de favoriser la communication entre les cartes par CAN bus et non plus par communication sans fil. Ce module est en quelque sorte le cerveau de la fusée. C'est là que sont centralisées toutes les mesures réalisées par la partie électronique du projet. Sa composition est simplifiée par rapport à l'an dernier. Il est maintenant composé de 5 cartes électroniques (en comptant la carte mixte introduite dans la bague éponyme), deux diodes, un interrupteur, une batterie, quatre tiges filetées cinq bagues, seize boulons et huit vis.

La bague de fermeture permet d'attacher le module à la peau de la fusée. Elle tient les cartes grâce aux tiges filetées. En dessous de cette dernière se trouvent la batterie, dont la bague a été repensée cette année de manière à améliorer la tenue de la batterie et la connexion de ses fils aux cartes électroniques. Le point important de ce module est la bague mixte. Elle est entièrement repensée par rapport à l'an dernier. Son but est de permettre à la fusée de communiquer avec l'extérieur. Une carte électronique y est bloquée entre deux bagues en bois de 10 mm et 6 mm d'épaisseur et de largeur le diamètre extérieur de la peau de la fusée.

Des encoches permettent aux diodes, à l'interrupteur et à la prise Jack d'être à fleur de peau de la fusée. L'interrupteur permet de mettre en marche la fusée, les diodes permettent de montrer le bon fonctionnement des cartes et des connexions à l'intérieur de la fusée, et enfin, la prise Jack permet de relier la fusée au sol, pour détecter le décollage lorsqu'elle est arrachée. La bague mixte de cette année se plus robuste que celle de l'an dernier et doit permettre d'éviter une rupture en vol comme cela s'est passé lors du lancé du PE 25.

Pour redessiner cette bague mixte, les zones les plus sensibles de celle de l'an dernier ont été renforcées, puis les efforts qui s'exercent sur cette bague lors du vol de la fusée ont été simulés sous Catia. Cette étude montre que l'ancienne bague subit de fortes dé-

formations, tandis que la nouvelle résiste sans déformations à cette contrainte.

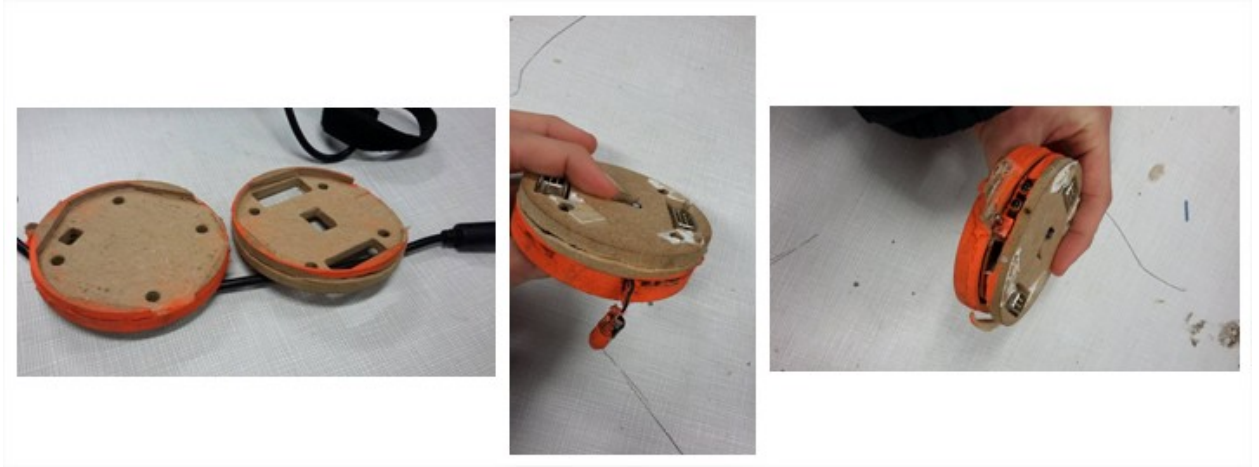


FIGURE 18: État de la bague mixte du PE précédent après lancement de la fusée

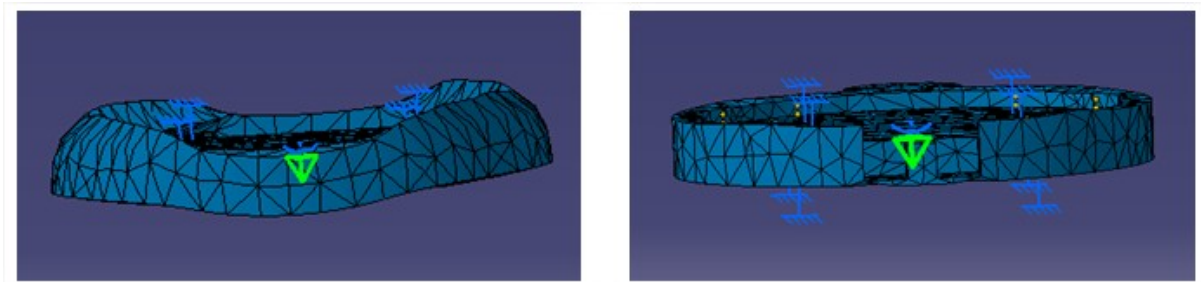


FIGURE 19: Simulation de comportement à un effort de l'ancienne bague (à gauche) et de la nouvelle (à droite)

En dessous de cette bague mixte se trouve l'ensemble des cartes programmables dont le fonctionnement sera expliqué dans la partie électronique. Ce sont ces cartes qui commanderont le servomoteur qui actionnent la trappe du parachute décrite plus haut, et lui sont cette année directement reliées.

La coiffe :

Le dernier module correspond à la coiffe de la fusée. Son apport concernant la stabilité sera expliqué plus loin à la page 26. Son objectif est de refermer la peau de la fusée et lui assurer de bonnes propriétés aérodynamiques. Elle a été calculée par le PE Fuseyyy de 2015 et réutilisée par le PE 25 de l'an dernier [PE25]. Nous nous inscrivons donc dans la continuité de ces projets en conservant le même profil.

6.1.3 Propulsion et stabilité

Maintenant que l'ensemble de la structure générale de la fusée a été décrite, il est possible de s'intéresser aux problèmes mécaniques que sont la propulsion de la fusée et sa stabilité.

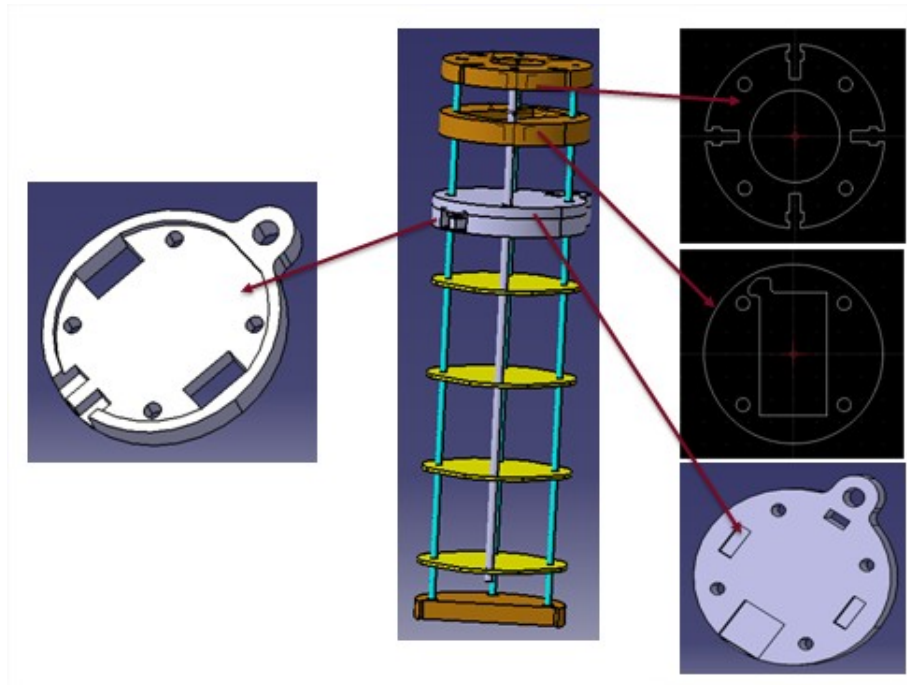


FIGURE 20: Modélisation du module cartes

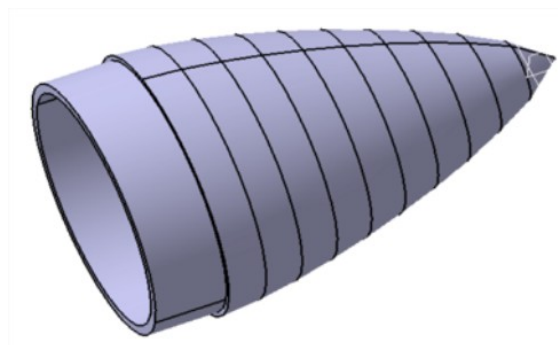


FIGURE 21: Modélisation de la coiffe sous Catia

La propulsion :

La propulsion de la fusée est assurée par le propulseur à poudre de type Cariacou fournit par le CNES pour la campagne du C'space. La poussée y est créée par combustion d'ergols solides lors de la phase de décollage. Ses dimensions sont normalisées par le CNES, d'où la similarité du module de propulsion avec les PE des années précédentes.

On retrouve les dimensions suivantes :

- 40 mm de diamètre
- 57 mm de longueur insérée dans la fusée
- 98.5 mm de longueur totale



FIGURE 22: Propulseur Cariacou

En matière de performance, on peut retrouver la courbe de poussée sur le site du CNES [6] :

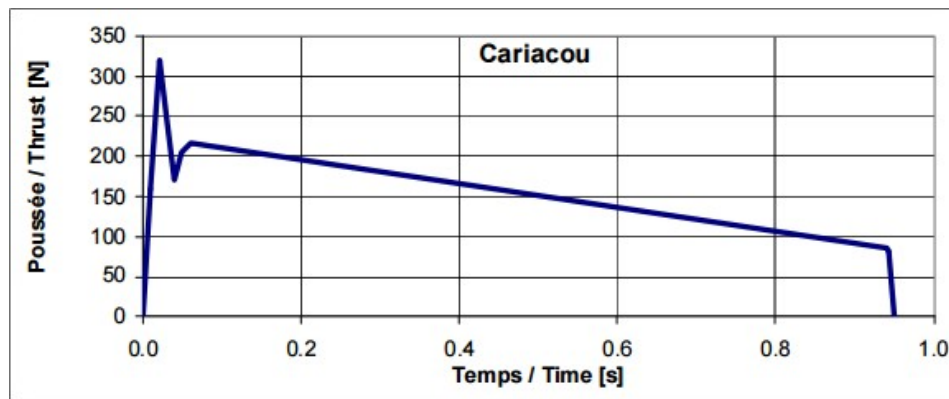


FIGURE 23: Courbe de poussée du propulseur Cariacou

On remarque une poussée maximum de 320 N et une poussée moyenne d'environ 150 N pendant presque une seconde, ce qui est un bon compromis pour un propulseur de cette taille.

La stabilité :

La stabilité de l'appareil lors de la phase de vol et notamment du décollage est calculée grâce à une page Excel, nommée Stabtraj, fournie par le CNES [11]. En entrant dans la page l'ensemble des dimensions de la fusée et les coordonnées de son centre de masse,

la feuille de calcul retrouve si la fusée est stable ou non. La stabilité de la fusée peut être contrôlée par les ailerons, la coiffe et le centre de masse, et est ajustée en jouant sur les dimensions des quatre ailerons. Des captures du StabTraj sont présents à la Figure 27.

- La coiffe :

La coiffe a entièrement été calculée par le PE Fuseyyy de 2015 pour améliorer l'aérodynamisme de la fusée [PE25]. Elle suit un profil de LD-Haack, profil qui maximise les performances de l'engin en régime subsonique (vitesse inférieure à la vitesse du son). Le profil est défini par l'équation (1) qui s'adapte à tout rayon intérieur R et toute longueur L de minifusée.

$$y = \frac{R\sqrt{\theta - \frac{\sin(\theta)}{2}}}{\sqrt{\pi}} = \arccos\left(1 - \frac{2x}{L}\right) \quad (1)$$

(où x et y sont la position selon les axes cartésiens de la fusée).

Cette coiffe est réalisée en CAO sous Catia, et a été imprimée grâce aux imprimantes 3D du Fablab de l'École Centrale de Lyon.

- Les ailerons :

Les ailerons sont un des éléments fondamentaux pour la stabilité de la fusée. Leur rôle est d'éviter que la fusée ne vrille et ainsi assurer une bonne éjection du parachute. Planète Sciences imposant d'en placer 4 pour convenir à la géométrie de la plate-forme de lancement, un matériau léger et rigide a été choisi de manière à éviter toute détérioration durant la phase de vol. Le matériau choisi est donc une plaque d'aluminium de 2 mm d'épaisseur.

Les dimensions de chacune des plaques sont déduites du StabTraj (page Excel mentionnée précédemment) qui vérifie la stabilité de la fusée en fonction de ses dimensions et de la position de son centre de masse (dont le calcul est précisé dans la prochaine partie). Les dimensions retenues sont présentées sur la Figure 24.

- Le centre de masse :

Plusieurs éléments sont à l'origine de la position du centre de masse. Les éléments les plus importants dans le calcul sont les ailerons dont la masse est relativement élevée par rapport au reste de la fusée, et l'ensemble des bagues et des cartes embarquées dans la fusée.

- Les bagues :

Les bagues sont conçues grâce aux logiciels QCAD et Galaad qui permettent pour le premier de concevoir la géométrie 2D des bagues et pour le deuxième, de communiquer la géométrie souhaitée à la fraiseuse automatisée Technodril du bâtiment F7 de l'École Centrale de Lyon. Cette fraiseuse taille les bagues dans des plaques de bois médium de 3mm, 6mm ou 10 mm d'épaisseur, ce bois alliant robustesse et faible coût (en matière de prix et de coût d'usinage). Le diamètre de ces bagues correspond exactement au diamètre intérieur de la fusée, ce qui leur permet d'être totalement solidaires avec le corps de la fusée.

- Le calcul :

Le centre de masse de la fusée a été calculé de manière précise grâce au logiciel

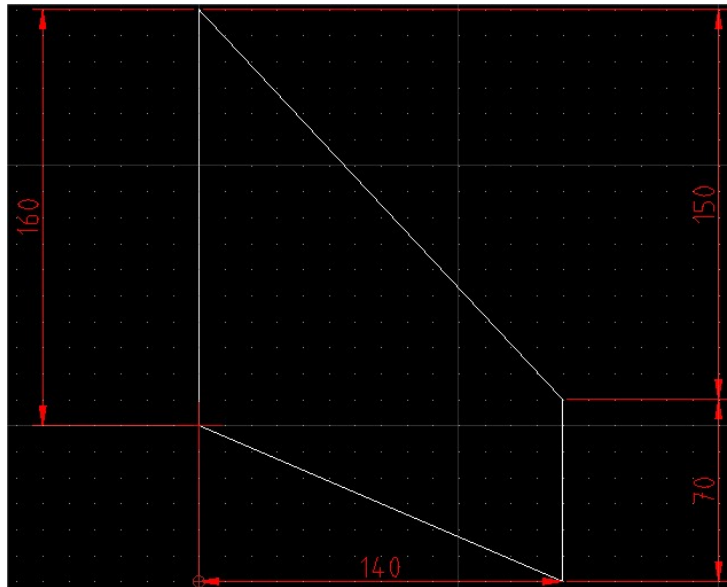


FIGURE 24: Dimensions des ailerons

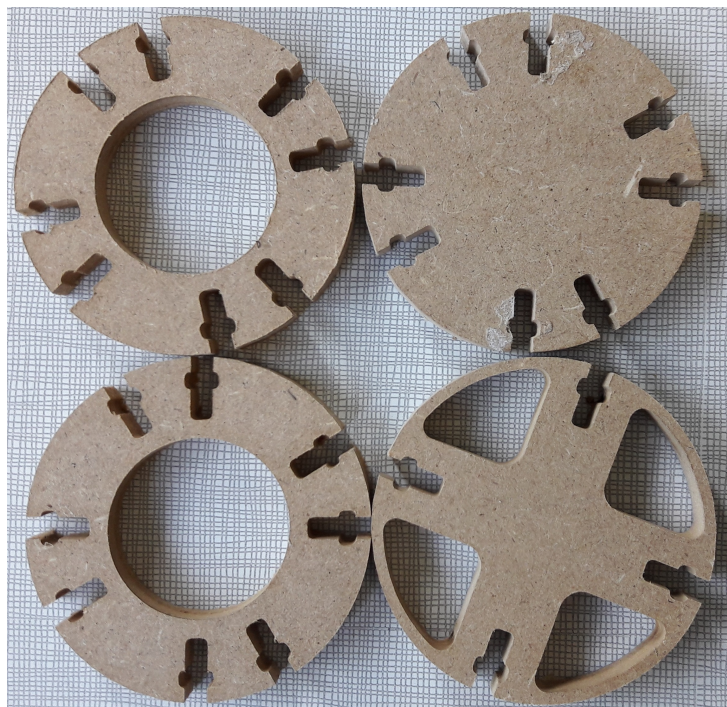


FIGURE 25: Exemple de bagues

Catia. Après la conception de l'ensemble des composants de la fusée, les masses volumiques de chaque matériau sont introduites et les masses des composants sont retrouvées grâce à l'ensemble des calculs de volume effectués par Catia. Une fois tout cela réalisé, une étude mécanique est lancée et Catia calcule facilement la position du centre de masse. Les résultats sont affichés sur la Figure 26.

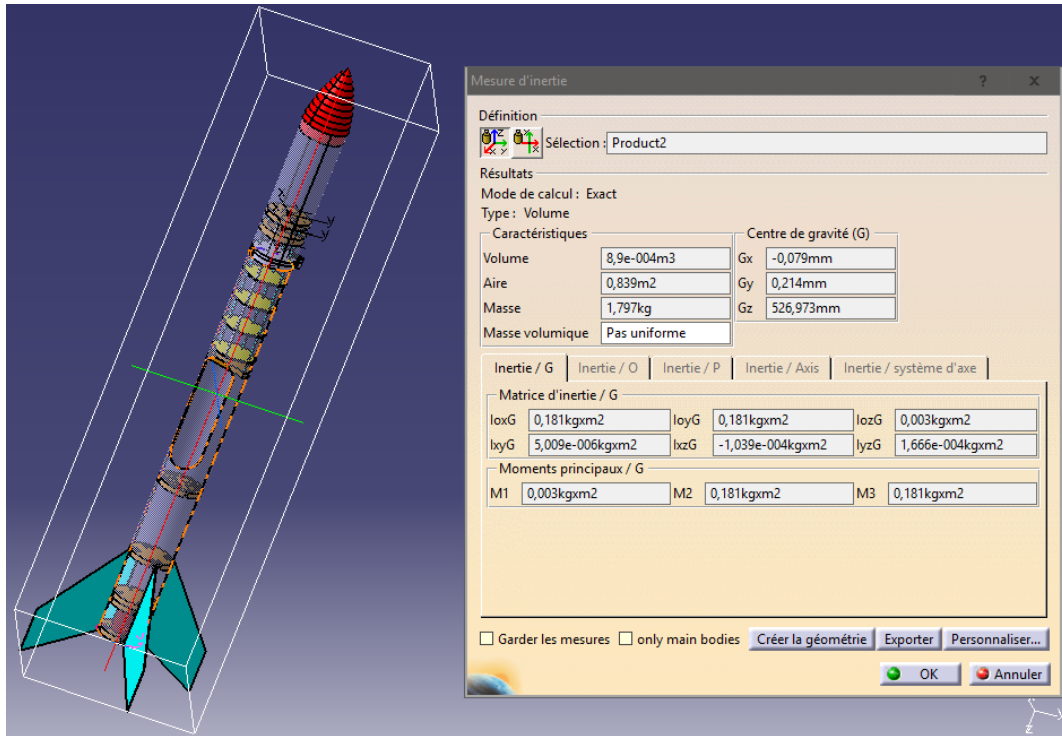


FIGURE 26: Résultat du calcul de centre de masse par Catia

Ce centre de masse est ensuite entré dans la feuille de calcul StabTraj qui permet de vérifier la bonne stabilité de la fusée, et par la même occasion, d'obtenir un aperçu de sa trajectoire et des données de vols qui seront récupérées par les parties informatique et électronique.

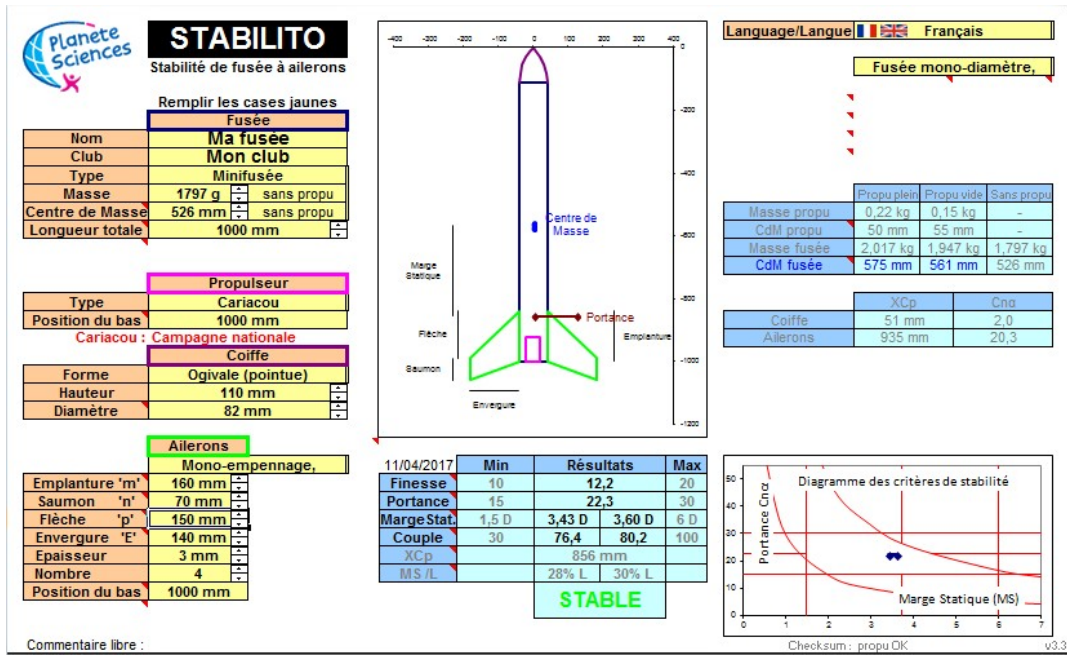


FIGURE 27: Validation de la stabilité par Stabtraj

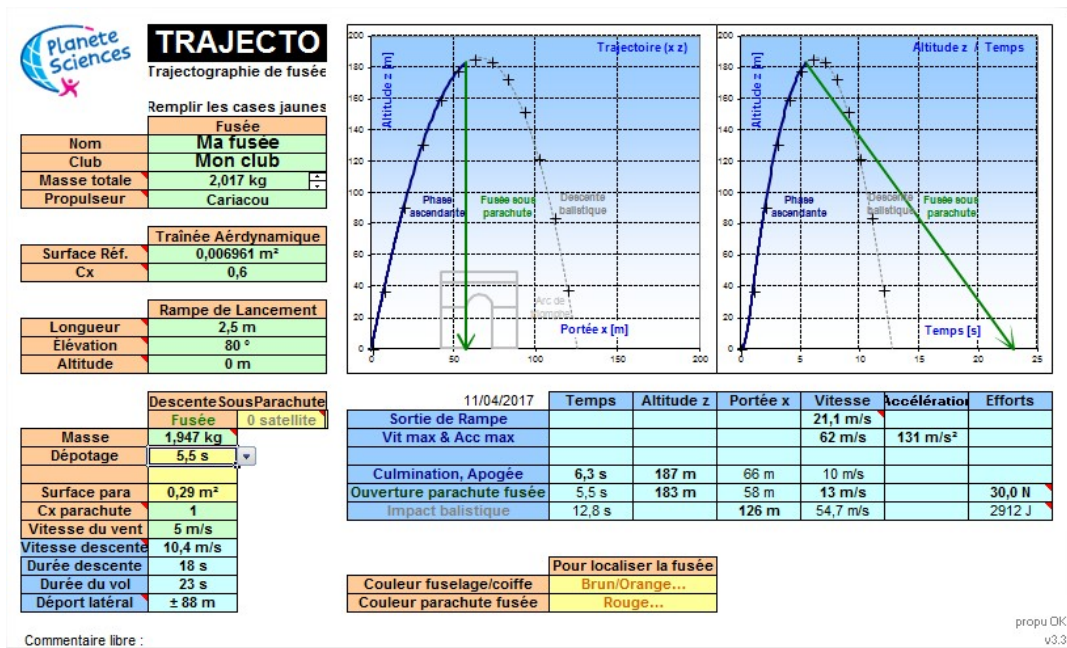


FIGURE 28: Anticipation de trajectoire par Trajecto

Éléments constitutifs de la fusée

La fusée incorpore un certain nombre de cartes, bagues et composants répertoriés dans la table ci-dessous :

	Dimensions	Masse
Fusée (sans propulseur ¹)	1 000 mm de hauteur 80 mm de diamètre	1,797 kg
Peau	1 000 mm de hauteur 80 mm de diamètre 1 mm d'épaisseur	217 g
Bagues	80 mm de diamètre 6 mm ou 10 mm d'épaisseur	
Parachute	650 mm de diamètre	43 g
Ailerons	Emplanture : 160 mm Envergure : 140 mm Flèche : 150 mm Saumon : 70 mm	
Propulseur	40 mm de diamètre 57 mm de profondeur dans la fusée	150 g à vide 220 g chargé
Coiffe	100 mm de hauteur 82 mm de diamètre	
Cartes	80 mm de diamètre	
Batteries LiPo	55 × 30 × 9mm	35 g

TABLE 1: Constituants de la minifusée

6.2 Partie électronique

6.2.1 Présentation générale de la case électronique

Afin de déclencher le lancement du parachute et d'assurer les diverses fonctions de la fusée, à savoir communiquer avec le sol et effectuer des mesures en vol, il est nécessaire de concevoir et réaliser des cartes électroniques, qui sont intégrées à la fusée. Le travail consiste notamment à reprendre et à valider les travaux effectués durant les précédents PE de minifusées.

6.2.2 Présentation des cartes

Structure générale

La case électronique est composée de 5 cartes :

- une carte alimentation
- une carte séquenceur

1. Le propulseur est fourni par le CNES au C'Space, il faut juste réaliser la case dans laquelle il sera inséré.

- deux cartes capteur
- une carte fille centrale inertielle
- une carte fille télémétrie
- une carte mixte

Elles sont disposées à l'intérieur de la fusée comme présenté sur la Figure 29. La Figure 30 est une photographie de ces cartes.

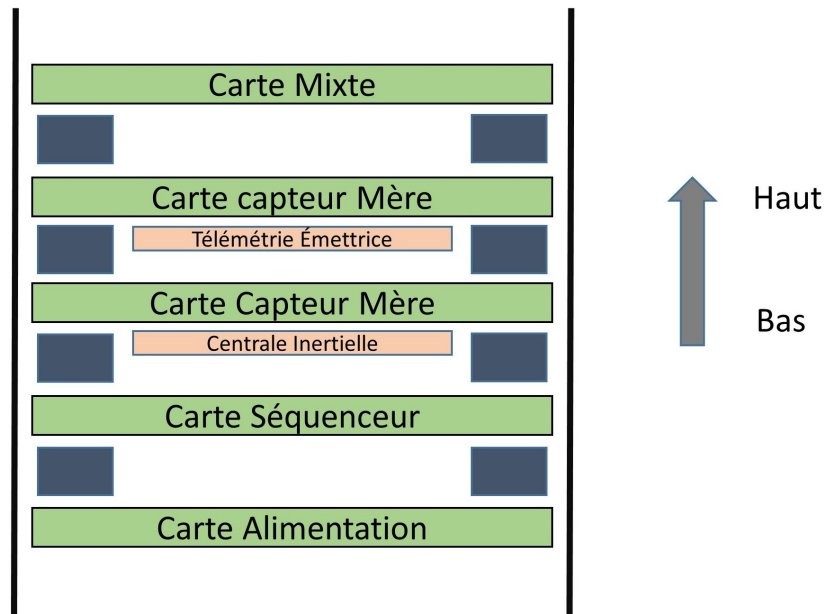


FIGURE 29: Schéma de répartition des cartes dans la fusée

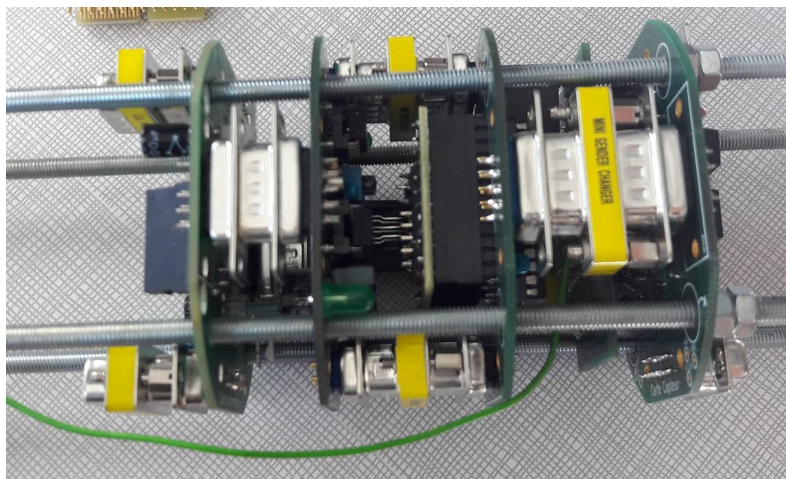


FIGURE 30: Cartes électroniques dans la fusée

La carte alimentation :

La carte alimentation est la carte qui sert à connecter la batterie et à fournir en tension les autres cartes. Elle comprend notamment deux convertisseurs, des convertisseurs

linéaires, qui fournissent en sortie une tension 3,3V pour l'un et 5V pour l'autre. L'énergie de la carte alimentation provient d'une batterie de type Lithium Polymère à 2 cellules avec une tension nominale de 7,4V.

Les dimensionnements souhaités, déterminés par les PE de l'année dernière, et les caractéristiques physiques de la carte alimentation sont résumés dans la Table 2.

Carte alimentation	Caractéristiques physiques	Dimensionnement
Masse	<100g	<100g
Potentiels	+3,3V,+5V	+3,3V et 5V
Tension nominale	7,4V	7,4V

TABLE 2: Caractéristiques et dimensionnement de la carte alimentation, les dimensionnements sont d'après le PE 25 2016

La carte séquenceur :

La carte séquenceur a un rôle clé dans l'électronique de la fusée, elle est la carte maîtresse du dispositif électronique, elle doit gérer la procédure de vol de la fusée de la rampe de lancement à l'atterrissage.

Dans un premier temps, la carte, par l'activation de diodes et de vibreurs, indique que la fusée est prête à être lancée. C'est également elle qui commande le déploiement du parachute embarqué si l'apogée est détecté par la centrale inertielle ou si les instruments sont défaillants.

Cette carte fonctionne à une fréquence de 20 MHz et présente les dimensions résumées dans la Table 3.

Carte séquenceur	Caractéristiques physiques	Dimensionnement
Masse	<100g	<100g
Technologie	PIC 18F2685	Sécurisante
Potentiel	+5V	+5V
Capacité de calcul	20MHz	Suffisante

TABLE 3: Caractéristiques et dimensionnement de la carte séquenceur, les dimensionnements sont d'après le PE 25 2016

Les cartes capteur mères :

Les cartes capteur sont en fait un assemblage de deux cartes, la carte mère et la carte fille. Les cartes mères servent de support aux cartes filles. Elles permettent de contrôler les cartes filles, d'acquérir les données, de les stocker et de les transmettre à la carte séquenceur.

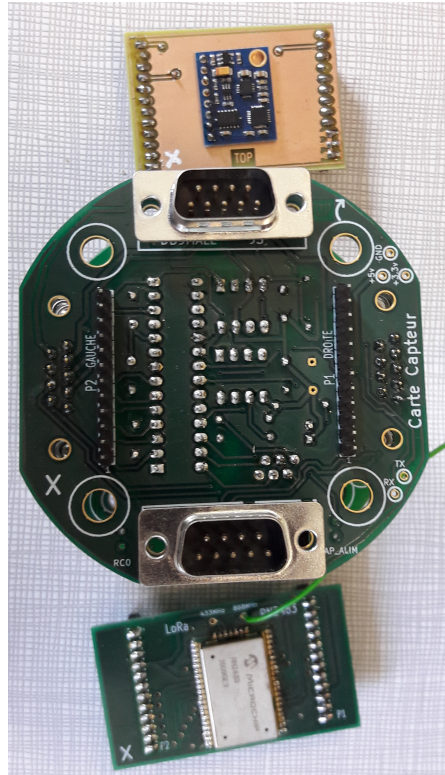


FIGURE 31: Carte capteur mère, entre deux cartes capteurs filles

La carte fille de la centrale inertielle :

La carte fille de la centrale inertielle est une nouvelle carte, sur laquelle est soudée la centrale inertielle. Contrairement à la plupart des projets minifusées qui embarquaient généralement un accéléromètre ou une sonde Pitot pour détecter l'apogée, il a été décidé, en plus du thème sur la communication à longue portée, d'ajouter une centrale inertielle à la fusée pour cette détection d'apogée.

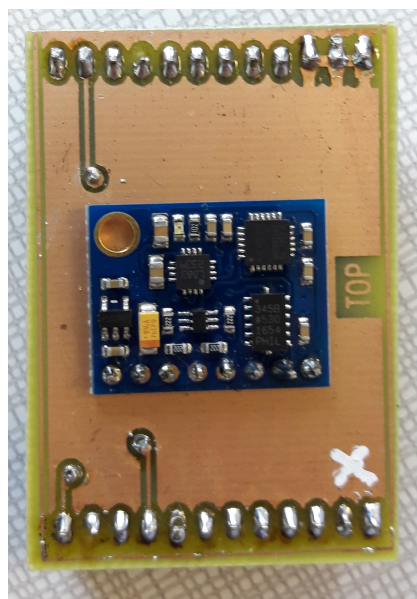


FIGURE 32: Carte fille supportant la centrale inertielle

Une centrale inertielle est un module qui incorpore un accéléromètre, qui mesure les accélérations linéaires, un gyroscope, mesurant les vitesses de rotation, et un capteur de champ magnétique. Les trois capteurs présents sur la centrale inertielle effectuent leurs mesures selon les trois axes de l'espace comme représenté sur la Figure 33.

Le module de centrale inertielle utilisé est un MPU-3000 de la société californienne INVENSENSE. Il embarque un accéléromètre numérique trois axes ADXL345 fabriqué par ANALOG DEVICE, un gyroscope trois axes ITG-3200 d'INVENSENSE et un magnétomètre numérique trois axes d'HONEYWELL. La centrale permet donc de mesurer des mouvements linéaires et rotatifs. Les fiches techniques des capteurs permettent d'avoir l'ensemble de leurs caractéristiques [5].

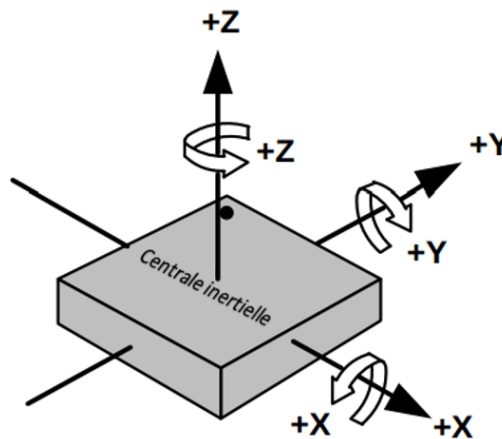


FIGURE 33: Schéma de la centrale inertielle avec les trois axes [1]

Dimensionnement Tout appareil de mesure présente un certain domaine de fonctionnement lui permettant d'être utilisé pour des applications spécifiques. La sensibilité du gyroscope est de $0,2^\circ/s$ et présente une échelle de fonctionnement de $\pm 2000^\circ/s$ ce qui est suffisant, la fusée ne devrait pas tourner sur elle-même de plus de 5,5 tours par seconde.

L'accéléromètre a une gamme de fonctionnement de $\pm 16g$. Les simulations faites sur Stabtraj ont montré que la fusée ne devrait pas subir une accélération de plus de $14g$. cet accéléromètre ne devrait donc pas saturer et convient à cette utilisation.

La carte fille télémetrie :

Cette carte sert de support au module LoRa, qui est un module de communication longue portée (voir la carte du bas de la Figure 31).

Contrairement à l'année précédente, il n'est pas nécessaire de communiquer par télémetrie entre deux étages de cartes. Ce module a pour vocation d'assurer la communication avec le sol et le recueil de données du vol au sol, pour ne pas avoir à les stocker dans la fusée.

La carte mixte :

Cette carte, basée sur les travaux des années précédentes fait partie de ce que l'on appelle la bague mixte. Sur cette carte sont fixés :

- un interrupteur, qui permet d'allumer la fusée
- une diode, qui permet d'indiquer visuellement l'état de la fusée
- une prise Jack, qui permet à la carte séquenceur de détecter le décollage

Cette carte, une fois insérée dans la bague mixte, permet de présenter ces éléments à l'extérieur de la fusée sans en compromettre la structure et l'esthétique. Son rôle est également d'éviter l'utilisation de fils, qui présentent un haut risque pour la fusée en cas d'arrachement, de court-circuit, ou de mauvais arrangement à déboguer, raison pour laquelle la communication par bus est privilégiée.

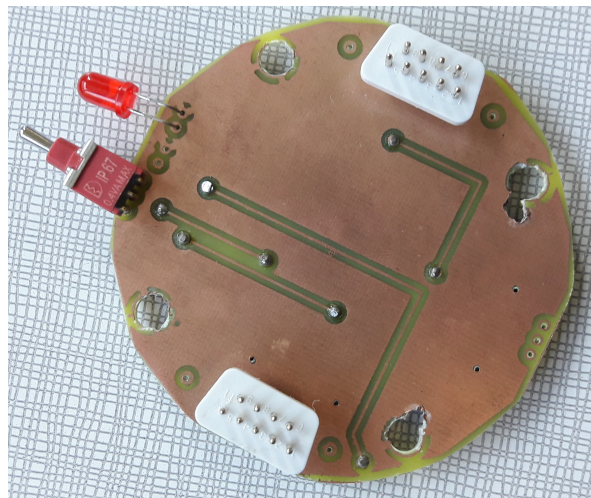


FIGURE 34: Carte mixte, réalisée par l'autre projet

6.2.3 Réalisation des cartes

Les cartes alimentation, séquenceur et capteurs, sont standardisées et documentées, grâce à des travaux précédents [4]. De fait, il faut uniquement souder les composants sur ces cartes, ce qui permet de se concentrer plus amplement sur le reste du projet. Cependant, ce standard ayant été revu l'année dernière suite à un partenariat entre le CLC et CIRLY, qui produit les nouvelles cartes, un problème de court-circuit sur la carte alimentation, et des problèmes de communication carte à carte, qui concernent la carte séquenceur et les cartes capteur sont apparus. Ces problèmes ont été résolus par notre tuteur, pour le premier en supprimant la connexion problématique, pour le second en faisant de nouvelles cartes, cela ayant bien entendu un impact non négligeable sur l'avancée du projet.

Cette année, il faut construire la carte fille avec la centrale inertielle. Le logiciel KiCad, qui est un logiciel de conception pour l'électronique, est utilisé pour réaliser cette carte. En se basant sur un tutoriel fourni par le tuteur et sur la documentation des composants, le schéma électrique de la carte puis son typon sont tracés, comme montré ci-dessous.

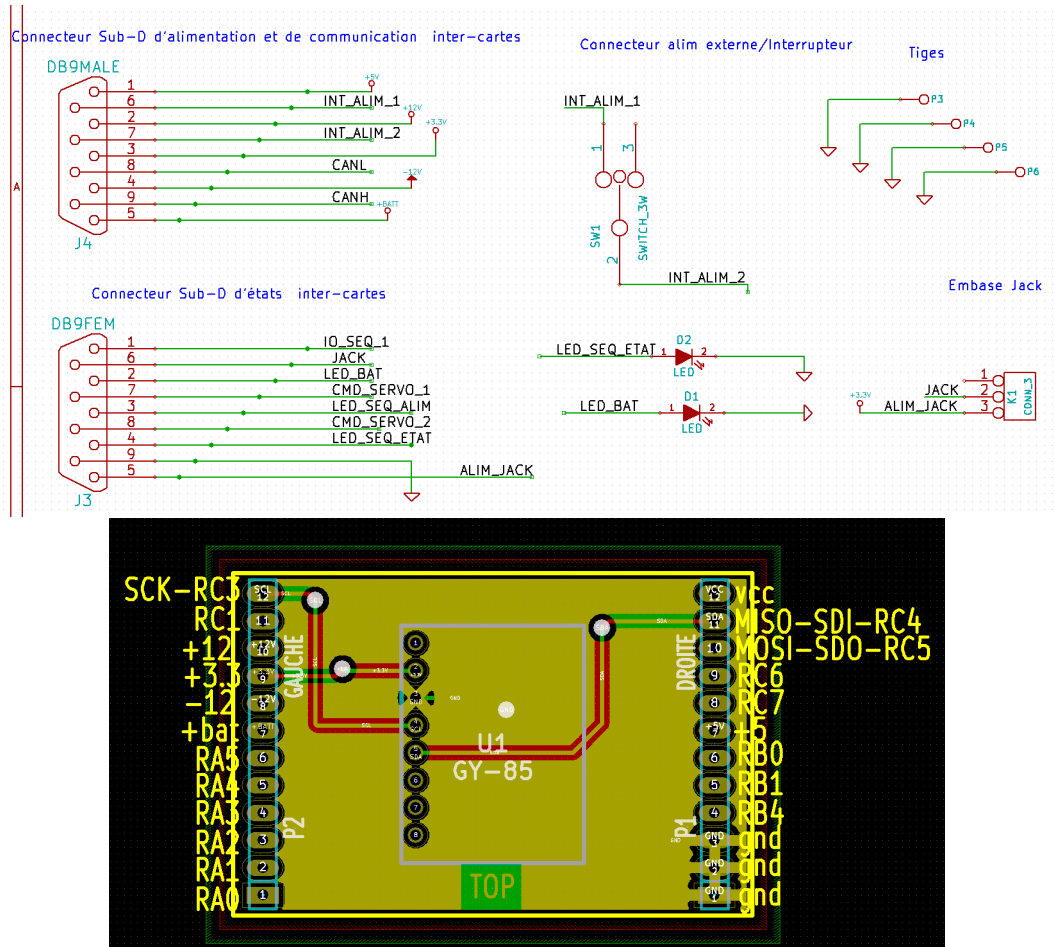


FIGURE 35: Schéma électrique et typon de la carte centrale inertielle

Il est aussi nécessaire de reprendre la conception de la carte mixte, en modifiant les contours de la carte, afin de répondre aux besoins de la partie mécanique. Le redimensionnement de la carte est fait en diminuant la taille des bords, ce qui permet d'augmenter l'épaisseur de la paroi de la bague.

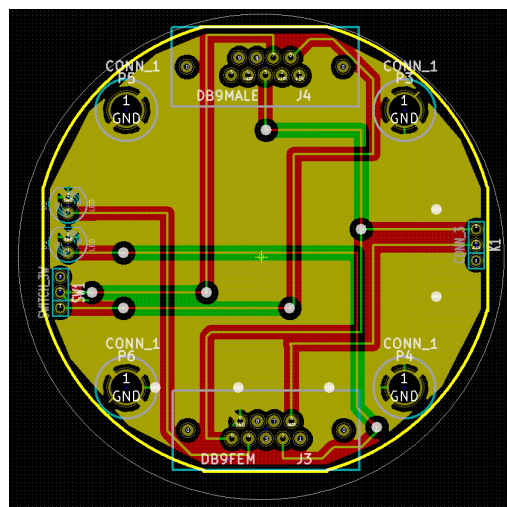


FIGURE 36: Typon de la nouvelle carte mixte

Lorsque la conception sur KiCad est achevée, le tuteur fabrique les cartes, puis les composants sont soudés dessus. Afin de vérifier le bon fonctionnement de cette carte, celle-ci est produite en un seul exemplaire, qui doit être testé par l'autre projet avant de réaliser celle pour ce projet.

6.2.4 Base au sol

L'un des objectifs de ce PE est de communiquer avec le sol pendant toute la durée du vol. Afin de répondre à cet objectif, il faut une carte qui restera au sol et sera reliée à un ordinateur, sur laquelle est présent un module LoRa pour récupérer les données émises par la fusée (voir Figure 37).

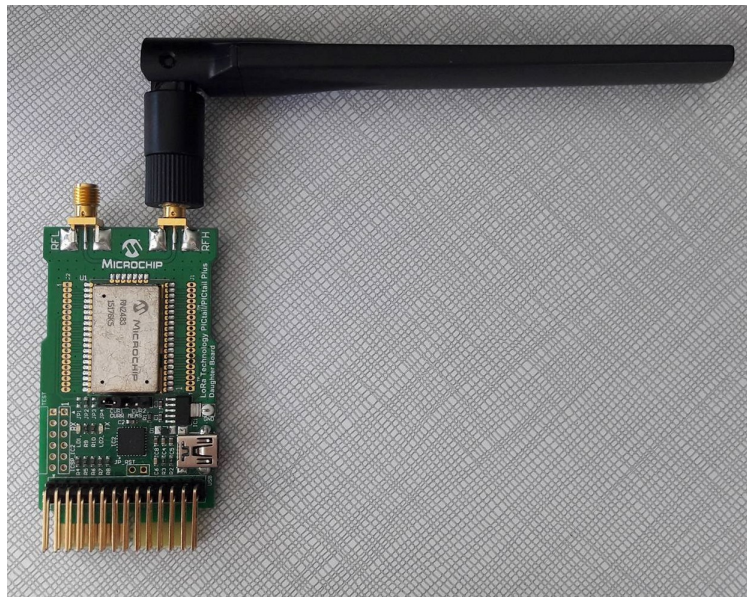


FIGURE 37: Récepteur LoRa, utilisé au sol avec un ordinateur

6.3 Partie programmation

Chaque carte électronique doit remplir un certain nombre de fonctions précises : la programmation leur permet d'assurer ces objectifs. Pour cela des microcontrôleurs PIC de la société Microchip Technology² sont utilisés. Ils sont programmés grâce au logiciel PIC C Compiler en langage C (Figure 38).

Afin de pouvoir programmer sans accumuler les erreurs, le logiciel dispose d'un mode exécutant le programme pas à pas, ce qui permet de repérer facilement les anomalies dans le code. Il n'est toutefois pas possible de toujours profiter de ce mode débogger, notamment lorsqu'il faut tester plusieurs cartes simultanément. En effet, chaque carte est programmée indépendamment des autres. Ainsi, pour tester la communication entre deux cartes à travers les CAN BUS, il faut programmer chaque carte l'une après l'autre sans mode débogage et tout tester d'un coup. Cela peut devenir problématique dès lors qu'il y a une erreur quelque part, car il est très difficile de localiser précisément la source d'erreur.

2. Microchip Technology, ou plus simplement Microchip, est un fabricant de semi-conducteurs américain fondé en 1989.[12]



FIGURE 38: Logotypes de Microchip (cartes électroniques) et PIC C Compiler (logiciel de programmation)

Dans la suite de cette partie ne sont détaillés que les principes et les fonctionnalités des programmes implémentés dans chacune des cartes programmables. Les codes en C qui y ont été implémentés sont eux fournis en annexe.

6.3.1 La carte séquenceur

Le programme utilisé sur la carte séquenceur est très largement inspiré de celui des années précédentes. Son but est de gérer l'enchaînement de toutes les phases du vol, notamment du point de vue des normes de sécurité imposées par le cahier des charges de Planètes Sciences.

Cette carte est essentielle au bon déroulement du vol et de sa préparation, elle doit indiquer par des diodes l'état de la fusée sur le pas de tir. Ces diodes sont disposées à la surface de la fusée de sorte à être visibles lorsque la fusée est installée sur la rampe.

Le programme de la carte séquenceur doit servir à prendre en compte et enclencher chacune des différentes étapes du vol. On y retrouve notamment la commande de vérification de la fermeture du servomoteur puis de son ouverture permettant le déploiement du parachute. L'organigramme informatique du code de la carte séquenceur est représenté sur la Figure 39.

Ce programme a un fonctionnement globalement très simple car il est très linéaire. Après avoir détecté et annoncé au reste des cartes que la fusée a décollé, le séquenceur lance un compte à rebours. Celui-ci est programmé pour se terminer 5 secondes environ après le départ, ce qui représente le temps théorique calculé avec Stabtraj que la fusée met à atteindre l'apogée. Lorsque le compte à rebours se termine, le parachute s'ouvre. Toutefois, une boucle vérifie régulièrement que l'apogée n'a pas déjà été détecté par la carte capteur, auquel cas elle libère immédiatement le parachute.

Ce compte à rebours sert de sécurité - imposée par le cahier des charges - au cas où un dysfonctionnement surviendrait dans la détection de l'apogée. Son réglage est très incertain car il dépend directement du temps d'apogée de la fusée, dont il est impossible d'avoir une idée réellement précise. Celui-ci dépend énormément des conditions environnementales propres au moment du lancement et dont le Stabtraj ne peut pas prendre compte.

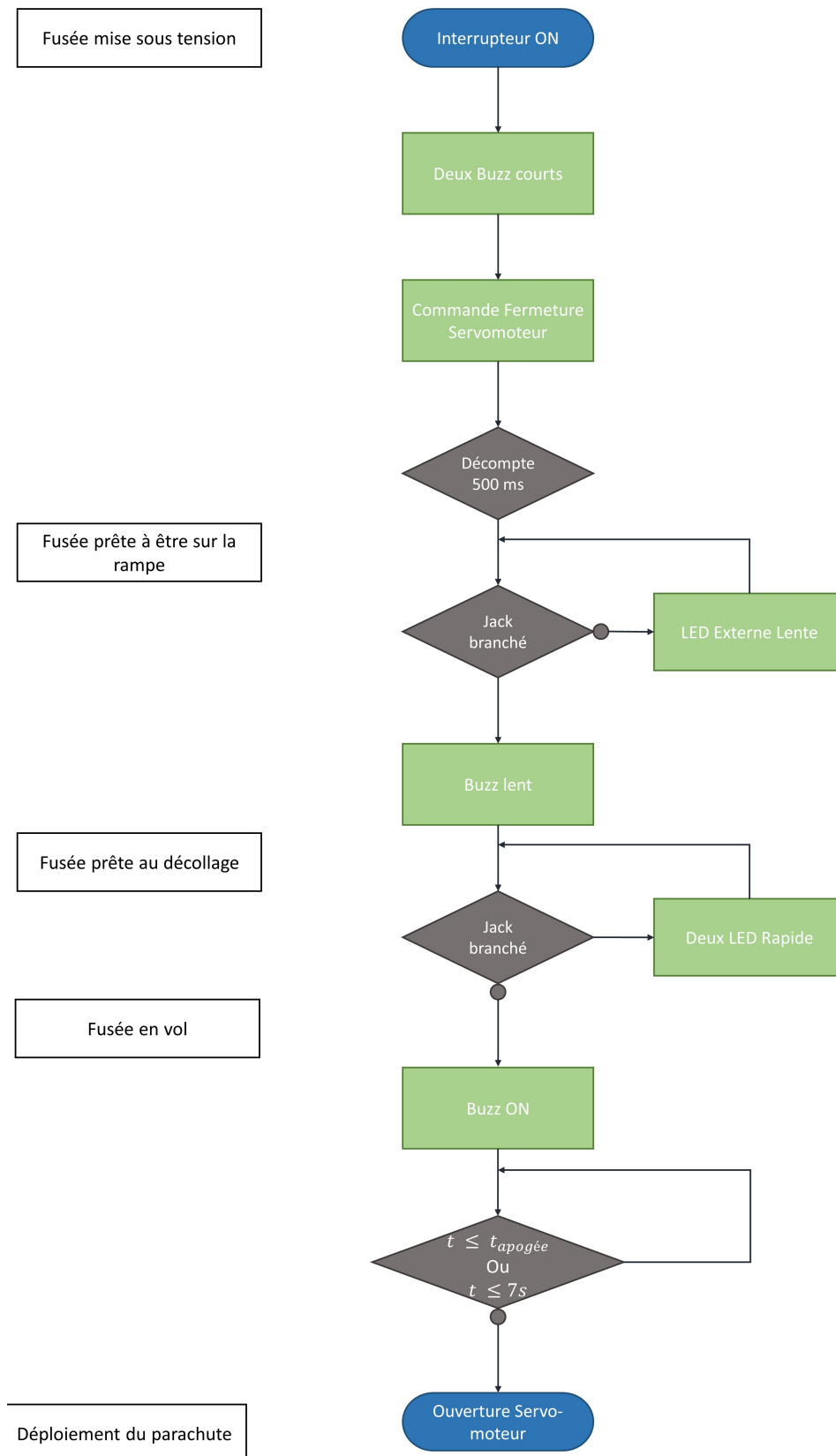


FIGURE 39: Organigramme informatique de la carte séquenceur

Or son réglage est décisif puisque s'il est mal fait, les conséquences sont graves. S'il est trop court, il risque de se terminer avant que l'apogée ne soit atteint par la fusée, ce qui avorte le vol en amont en plus d'empêcher de savoir si le programme de détection d'apogée fonctionne.

Au contraire, s'il est trop long, il risque de déclencher le parachute trop tard après l'apogée, c'est-à-dire à un moment où la fusée a déjà repris beaucoup de vitesse. Dans cette situation, le parachute ne sert plus à rien et la fusée est en chute libre vers la terre. À la hauteur d'où elle tombe, cela signifie la destruction totale de la fusée au sol.

On peut donc seulement espérer régler le compte à rebours pour qu'il se termine très peu de temps après l'apogée, c'est-à-dire suffisamment tard pour laisser l'algorithme de la carte capteur la détecter et suffisamment tôt pour empêcher la fusée de reprendre de la vitesse.

6.3.2 Carte capteur : module de la centrale inertielle

La centrale inertielle est capable de faire trois types de relevés : des relevés de champ magnétique, d'accélération et de vitesse angulaire.

Seules les mesures du gyroscope sont utilisées et traitées par un ordinateur pour détecter l'apogée. Les mesures de l'accéléromètre sont prises uniquement pour être transmises au sol afin de reconstruire la trajectoire de la fusée *a posteriori* avec un véritable ordinateur, elles ne subissent aucun traitement direct dans la fusée. La mesure du champ magnétique étant totalement disjointe de ces objectifs, il a été décidé de négliger purement et simplement cet aspect de la centrale inertielle.

En effet, la qualité de la reconstruction de la trajectoire dépend directement du nombre de points de mesure des données utiles réalisés durant le vol. Or ce nombre est limité car il faut du temps au microcontrôleur pour aller chercher la donnée dans la centrale, la récupérer et l'envoyer à la télémétrie. Si en plus de l'accélération et de la vitesse angulaire, le champ magnétique était pris en compte, cela augmenterait de 50% tous les temps de prise de mesure, et donc diminuerait d'au moins 30% le nombre de points réalisés pour la reconstruction de trajectoire. Ce luxe ne pouvant être pris, le champ magnétique n'est pas mesuré durant le vol.

Le facteur limitant en terme de temps pour l'ensemble de la fusée est précisément la carte capteur. En effet, il lui faut du temps à chaque fois qu'elle récupère une donnée dans la centrale, et elle doit le faire 12 fois par cycle (6 variables obtenables par 2 voies). Puis, elle doit réaliser le moyennage temporel des valeurs, traiter les mesures pour les rendre utilisables, vérifier si l'apogée a été atteint et enfin envoyer sur le BUS les données. Il est prouvé plus loin dans le rapport qu'aucune autre boucle n'excède en termes de temps celle de la carte capteur.

Le problème est qu'il est difficile de déterminer avec précision le temps que met le microprocesseur à réaliser chacune de ces opérations. D'autant que ces temps peuvent varier légèrement à chaque cycle de l'algorithme. Il a donc été décidé d'imposer régulièrement des délais de 5ms après toute opération complexe. Pendant que les 5 ms s'écoulent, le microprocesseur a le temps de terminer la tâche qu'il a reçu juste avant. Le temps de 5ms est surestimé afin de s'assurer que l'ordre est accompli avant de passer à la suite.

Calculer la somme des délais créés tout au cours d'une boucle du programme de la carte permet d'obtenir le pas temporel du système. Toutes les grandes boucles des cartes de la fusée ayant un rôle à l'échelle intercartes ont comme consigne temporelle ce pas de temps.

Algorithme de détection de l'apogée

Le programme du PE 25 de l'année dernière exploitait les mesures d'un accéléromètre et devait détecter l'apogée suite à un algorithme empirique qui s'est montré très peu efficace³. En effet, le parachute s'est déployé trop tôt, avant l'apogée, ce qui aurait pu être fatal, bien que la fusée ait pu atterrir en douceur par la suite. Ainsi, voulant éviter un déploiement assez approximatif, nous avons voulu repenser le processus de détection du sommet de la trajectoire.

C'est l'inclinaison de la fusée qui est étudiée pour commander l'ouverture du servomoteur au lieu de la trajectoire du vol. Pour repérer la fusée dans l'espace, nous utilisons les vitesses angulaires de *précession* ψ et de *nutation* θ d'Euler. On prend les notations de la Figure 40.

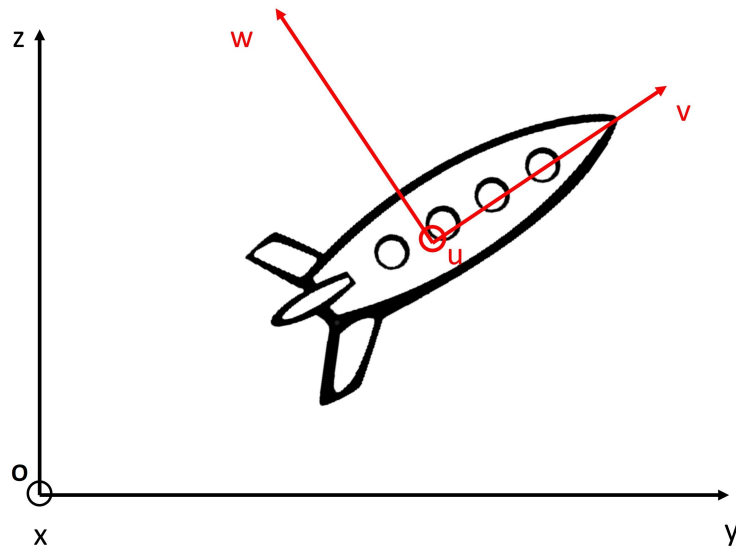


FIGURE 40: Schéma représentant les différents repères

On définit les trois repères suivants :

- le repère galiléen $R_T(O, \vec{x}, \vec{y}, \vec{z})$ lié à la Terre.
- le repère de transition $R_1(O, \vec{x}, \vec{y}', \vec{z}')$.
- le repère $R_f(O, \vec{u}, \vec{y}' = \vec{v}, \vec{w})$ lié à la fusée.

Les figures de changements de bases sont représentées sur la Figure 41.

Par transitivité, la vitesse de rotation de la fusée par rapport au sol s'écrit :

3. Cet algorithme faisait la moyenne des cinq dernières valeurs mesurées par l'accéléromètre et comparait cette somme à un seuil déterminé empiriquement.

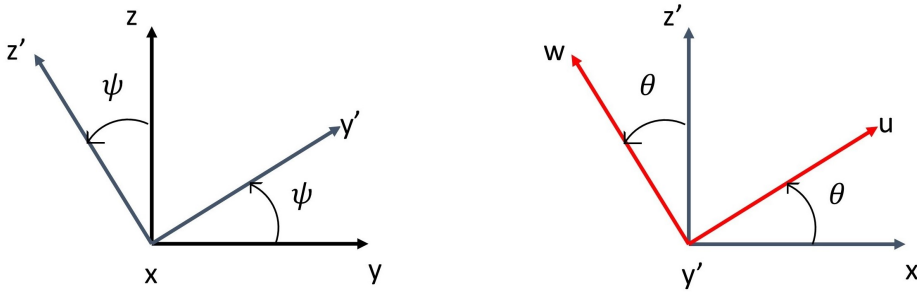


FIGURE 41: Figures de changement de base

$$\begin{aligned}\vec{\Omega}_{R_f/R_t} &= \vec{\Omega}_{R_f/R_1} + \vec{\Omega}_{R_1/R_t} \\ \vec{\Omega}_{R_f/R_t} &= \theta' \vec{y}' + \psi' \vec{x}\end{aligned}$$

Or, par décomposition on obtient $\vec{x} = \cos(\theta)\vec{u} - \sin(\theta)\vec{w}$

$$\vec{\Omega}_{R_f/R_t} = \psi' \cos(\theta)\vec{u} - \psi' \sin(\theta)\vec{w} + \theta' \vec{y}' \quad (2)$$

À partir de ces données, il faut déterminer l'inclinaison ψ de la fusée par rapport à la Terre. La condition qui permettra de déployer le parachute est alors $\psi \leq 0$. Celle-ci est illustrée sur la Figure 42, au cours de la phase de montée, l'angle ψ est positif, ce qui empêche tout déploiement d'un parachute. Au contraire une fois que la fusée atteint l'apogée, l'angle ψ devient négatif et décroît.

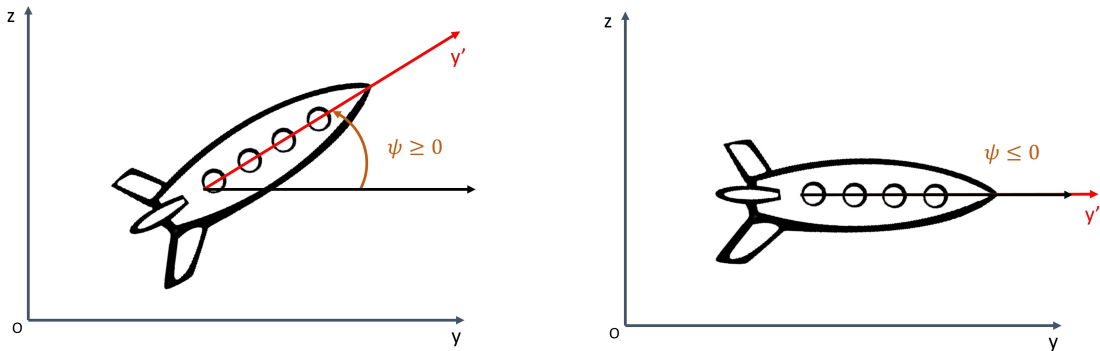


FIGURE 42: À gauche : une fusée dans sa phase de montée. À droite : fusée à l'apogée

L'algorithme de la détection d'apogée est décrit par la suite. On note Δt le pas de temps dans l'algorithme. De plus, l'indice n affecté aux mesures signifie qu'il s'agit de la mesure réalisée à la $n^{\text{ième}}$ itération.

- Avant le lancement, l'angle ψ est initialisé à 80° ($\psi_0 = 80^\circ$). Comme indiqué dans le cahier des charges, c'est l'angle entre la rampe de lancement et le sol. D'autre part, l'angle θ est initialisé à 0° .
- À chaque itération n de l'algorithme, les mesures de la vitesse angulaire θ'_n , et de $\psi'_n \cos(\theta_n)$ et $-\psi'_n \sin(\theta_n)$ sont obtenues sur les axes respectifs \vec{y}' , \vec{u} et \vec{v} du

gyroscope d'après l'équation (2).

- θ_n est calculé par l'intégration : $\theta_n = \theta_{n-1} + \Delta t \cdot \theta'_n$.
 - $\cos(\theta_n)$ et $\sin(\theta_n)$ sont calculés afin de retenir le plus grand par la suite⁴. Si $\cos(\theta_n)$ est supérieur, alors $\psi'_n \cos(\theta_n)$ est divisé par $\cos(\theta_n)$, sinon c'est $-\psi'_n \sin(\theta_n)$ qui est divisé par $-\sin(\theta_n)$. Ceci permet d'obtenir la vitesse angulaire ψ'_n .
- ψ_n , la valeur de l'angle ψ au cours de la n ième itération, est ensuite déduit par le calcul $\psi_n = \psi_{n-1} + \Delta t \cdot \psi'_n$.
- On compare l'angle ψ à 0.

L'organigramme de la Figure 43 relève les différentes étapes effectuées par la carte capteur sur laquelle est disposée la centrale inertielle :

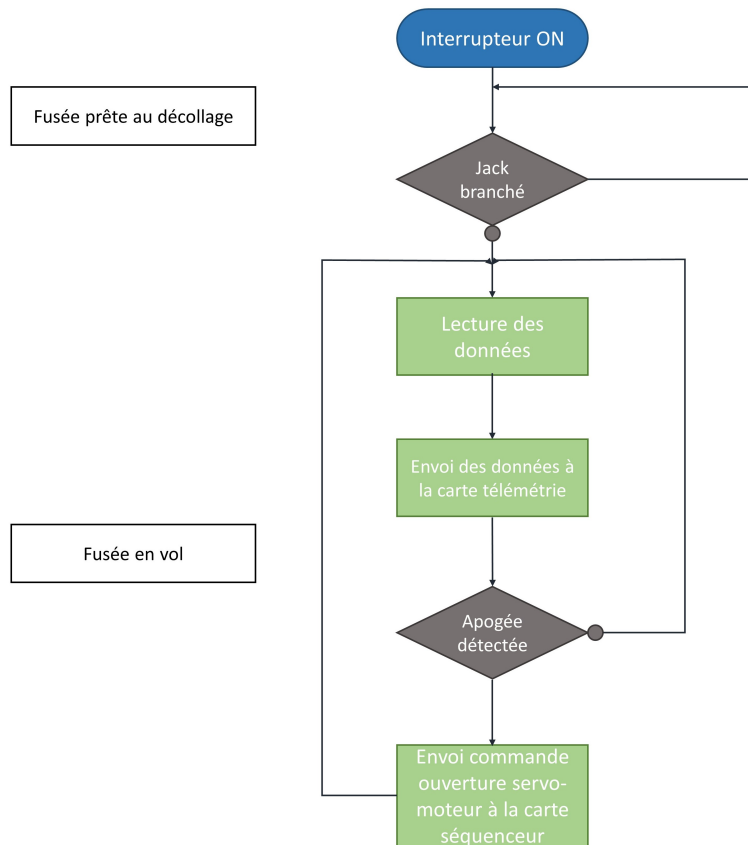


FIGURE 43: Organigramme de la carte capteur intégrant la centrale inertielle

4. Ceci est effectué dans le but de diminuer les erreurs dues aux incertitudes.

6.3.3 Carte capteur : télémétrie émettrice

Pour gérer la communication à longue portée, une carte émettrice, dont l'objectif est d'envoyer toutes les données mesurées par la centrale inertielle, et une carte réceptrice, positionnée au sol, doivent toutes les deux être programmées.

La maîtrise de la communication longue portée est d'autant plus importante qu'aucune donnée mesurée n'est stockée dans une mémoire, contrairement à l'année précédente. Autrement dit, aucune reconstitution de la trajectoire de la fusée n'est possible si cette partie de la programmation n'est pas fonctionnelle.

Aucune sauvegarde sur mémoire morte n'est réalisée afin de réaliser des gains de temps. En fait, il s'agit du même raisonnement que pour le choix de négliger la mesure du champ magnétique. Comme la télémétrie s'occupe déjà de stocker les données hors de la fusée, c'est un luxe de les mémoriser encore autre part. Hors ce luxe coûte du temps de travail au microprocesseur, ce qui diminue en bout de course la qualité de la reconstitution.

La communication des mesures se fait grâce à la technologie LoRa. Cela est réalisé à l'aide d'un module de transmission MicroChip RN2483 à 868 MHz incorporé sur une carte capteur fille. La puissance et la fréquence d'émission de ce module sont normalisées : la portée n'est pas grande (15km dans une base militaire en pleine campagne) et la bande de fréquence est inutilisée.

La carte dispose d'une longue antenne enfermée dans le corps de la fusée pour pouvoir émettre correctement. L'antenne a été réalisée avec un simple fil électrique légèrement dénudé à chaque extrémité.

Le module est capable d'envoyer environ 1367 octets/s. 6 variables sont envoyées au sol, chacune codée sur 16 bits, ce qui représente donc 96 bits. Sachant que des bits sont aussi utilisés pour le protocole de reconnaissance, on majore largement en considérant que chaque émission est de 200 bits. On peut donc réaliser 55 émissions par seconde, soit un écart de 18 ms entre chaque mesure. Le vol jusqu'à l'apogée durant environ 5 seconds, cela représente plus de 250 points pour la reconstitution de la trajectoire préapogée, ce qui est largement suffisant. Le fait que le poids numérique des émissions ait été en plus fortement surestimé permet de considérer que le facteur limitant temporellement est la vitesse de mesure de la centrale inertielle. Le pas de temps qui sépare chaque émission découle donc directement du temps mis par la carte capteur à faire toutes ses mesures et à les envoyer sur le BUS.

Le fonctionnement de la carte se résume à une unique boucle infinie dans laquelle elle rentre après avoir reçu le signal de décollage de la carte séquenceur. Cette boucle consiste exclusivement à aller chercher dans le BUS les données émises par la carte capteur, pour les envoyer immédiatement au sol par le module LoRa.

L'organigramme de l'algorithme de la carte télémétrie émettrice est présenté Figure 44.

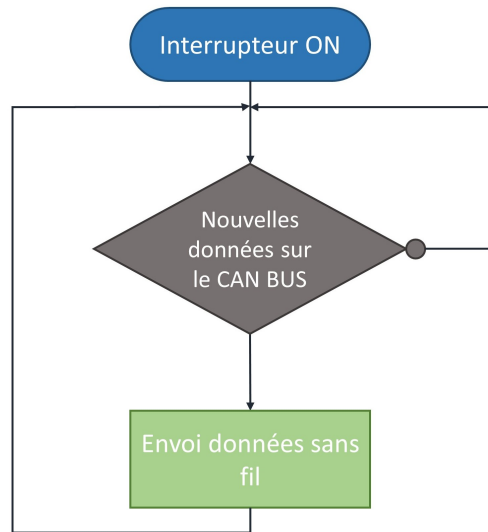


FIGURE 44: Organigramme de la carte télémétrie émettrice

6.3.4 Carte télémétrie réceptrice

La carte réceptrice est très particulière. Elle reste au sol, attachée à un ordinateur portable et couplé à un logiciel adéquat : Visual Studio. Son objectif est de capter toutes les données envoyées par la carte de télémétrie émettrice et de les communiquer à l'ordinateur. Visual Studio permet ensuite de traiter les données reçues pour les enregistrer sous forme de tableur (typiquement au format CSV), ce qui permet n'importe quel traitement ultérieur, comme représenter la trajectoire de la fusée en 3D.

Cette carte représente le test ultime de fonctionnement des programmes de la fusée. En simulant le vol, de la pose de la fusée sur la rampe à son atterrissage, il est facile de constater sur l'ordinateur un grand nombre d'anomalies de programmation. En effet, le fichier que génère l'ordinateur une fois l'acquisition de données terminée est le résultat de tous les programmes réalisés en amont, de la carte séquenceur à la télémétrie réceptrice en passant par la carte capteur.

De même que pour la télémétrie émettrice, l'architecture globale de cette carte est très simple : il s'agit d'une boucle qui à chaque cycle récupère les données envoyées depuis la fusée et les transmet à l'ordinateur, qui les stocke dans des tableurs. La durée de la boucle est modulable afin de s'adapter facilement à l'expérience en cours, selon qu'il s'agit d'un test de quelques secondes ou du lancement véritable.

7 Bilans

7.1 Bilans du projet par pôles

7.1.1 Partie mécanique : Alleon Julien

Le fait de s'inscrire dans la lignée des PE minifusées des années précédentes fut un avantage non négligeable pour le bon avancement du projet. Cependant, le retard pris sur certaines parties du projet s'est en partie répercuté sur la partie mécanique (notamment le retard de fabrication de la carte mixte qui a engendré un retard sur la bague mixte). De plus, la prise en main de certains outils comme Catia devrait être un peu plus encadrée par les deuxièmes années de manière à transmettre certaines connaissances et éviter une perte de temps.

Cependant, le bilan de la partie mécanique reste positif malgré les problèmes rencontrés. Une meilleure passation des connaissances et une meilleure gestion des conflits entre les pôles du projet permettraient à la partie mécanique de se terminer plus rapidement et ainsi permettre d'aider la partie programmation.

Pour ma part, j'ai trouvé le projet très intéressant et instructif. Le fait de travailler sur un objet concret tout au long de l'année permet de motiver le groupe et les échéances permettent de mieux gérer les retards et le projet en général. Ce fut donc une bonne expérience malgré les quelques points noirs cités précédemment.

7.1.2 Partie électronique : CHARREAUX Guillaume

Pour la partie électronique, il a fallu travailler en tandem avec un autre projet minifusée, ce qui, malgré les retards que cela a pu engendrer pour la carte mixte, a permis de construire plus rapidement une base solide sur laquelle pouvait s'appuyer le pôle de programmation. Les travaux laissés par les PE précédents ont aussi contribué à cette dynamique ; la validation, ou non, de leurs travaux permettant de léguer aux prochains des conceptions sûres et fonctionnelles.

Le bilan de cette partie est positif, les délais prévus dans le diagramme de GANTT ayant été globalement respectés. Il est à noter cela dit que la technicité des cartes rend le projet extrêmement dépendant de leur bonne conception, mais rend également très complexe toute correction à apporter, ce qui s'est notamment soldé par un remplacement des cartes séquenceur et capteur. De fait, il est primordial d'avoir un retour du pôle de programmation.

Le fait de travailler sur un aspect aussi concret de la fusée que la réalisation de son électronique a été une réelle motivation, mais aussi un apport de savoir-faire non négligeable. Cela fut aussi l'occasion de découvrir le travail en équipe et en mode projet dans ses différents aspects. Ce fut donc une bonne expérience malgré les embûches rencontrées.

7.1.3 Partie programmation : JOUHET Guilhem, MANCY Damien, YANNE Louis

Malgré la mise à disposition des codes de l'année précédente, et l'assurance que ceux-ci fonctionnaient (puisqu'à part la détection de l'apogée, tout s'était impeccablement déroulé pour le groupe de 2016), la partie programmation a été particulièrement laborieuse. En effet, la passation de connaissance d'une année à l'autre est utile si les receveurs sont en mesure d'utiliser ces connaissances, ce qui n'a pas été le cas cette année. Pour d'anciens préparateurs exclusivement familiers du Python, l'apprentissage du C "sur le tas", presque sans tutoriels et sans aucun cours théorique s'est révélé délicat. D'autant que les programmes utilisaient pour beaucoup des fonctions spéciales intimement liées à

l'électronique plus qu'à l'informatique à proprement parler, ce qui les rendaient particulièrement peu adaptés pour une prise en main par des débutants. Sans même parler de modifier les programmes pour qu'ils conviennent aux nouveaux objectifs, il était difficile de seulement les comprendre.

Cette difficulté de prise en main du projet n'a pas été facilitée par le matériel disponible. Il a en effet fallu se familiariser avec le logiciel PIC C Compiler, qui est très complet, donc long à maîtriser. Des courts circuits et des problèmes de synchronisation entre les ordinateurs et nos cartes programmables étaient récurrents en début de projet, car nous avions du mal à initialiser correctement le système. Par ailleurs, les ordinateurs, très anciens, n'aidaient pas à accélérer les opérations, et leurs écrans cathodiques ne sont pas adaptés à de longs travaux. Cependant, il a été récupéré en milieu de projet un ordinateur plus récent qui a facilité le travail de la partie programmation. Toujours du point de vue du matériel, les délais de réception des cartes terminées ont fortement handicapé la progression de la programmation, puisque sans cartes, il n'y a rien à programmer. Un gâchis d'autant plus important que cela pouvait rendre d'un coup 3 membres du groupe chômeurs.

Un manque de soutien s'est aussi fait sentir lors du premier rendez-vous avec l'association Planètes Sciences en février. Alors que cela était prévu depuis septembre, l'utilisation du module LoRa, qui constitue l'essence du PE, a reçu un accueil des plus glacials. Lors d'une correspondance y faisant suite, elle a même clairement exprimé qu'elle demandait l'utilisation de modules radicalement différents. Cela est plus dû à un manque de communication regrettable entre le projet et Planète Sciences, dont nous tenons encore une fois à remercier les bénévoles qui prennent sur leur temps pour nous faire découvrir le milieu aérospatial. En effet, il est plus simple d'utiliser les modules de communication KIWI fournis par l'association Planète Sciences, mais l'utilisation du module LoRa répond à des objectifs du CLC à long terme, les performances du module KIWI étant limitées et son poids élevé.

L'avantage de la partie programmation a été de nous préparer à toutes les éventualités dans l'accomplissement d'un projet. Nous avons à présent une vision claire de la difficulté que peut représenter une tâche collective sur le long terme et de tout ce qui peut nuire à son achèvement. Enfin surtout, grâce aux nombreuses restitutions jalonnant l'année, nous avons appris à défendre efficacement ce type de projet.

7.1.4 Partie gestion de projet : JOUHET Guilhem

Bien que notre travail repose sur celui des PE précédents, l'équipe a eu du mal à récupérer les données complètes du PE 25 de l'an dernier. C'est pourquoi il serait profitable pour l'association comme le CLC de mettre en place un Wiki pour assurer une meilleure transmission des savoirs, le site du CLC n'étant pas suffisant pour trouver des réponses à des questions précises, ou trouver les codes en C des anciens PE. La gestion de projet a permis de ne pas perdre de temps pendant l'année. Cependant, bien qu'il ait été identifié que le projet prenait du retard avant le RVP2, il n'a pas été possible de trouver des solutions qui auraient permis de finir le projet dans de bonnes conditions. La nécessité du PE 68 d'utiliser un ordinateur pendant trois semaines au moment où l'équipe de programmation avait besoin de deux ordinateurs pour coder en parallèle le module de communication sans fil et la centrale inertielle. Un temps non négligeable a également été perdu en attendant pendant deux mois la carte électronique qui s'insère

dans la bague mixte, pendant qu'elle était en test dans le PE 32. En effet, cette carte est nécessaire à l'édition du programme définitif de la carte séquenceur. C'est pourquoi une meilleure cohésion entre les PE du CLC ne pourrait qu'être bénéfique pour tout le monde, en permettant un meilleur partage de connaissances et en limitant le temps perdu. Tout au long de l'année, des problèmes matériels (problème de CAO sur les cartes électroniques, module LoRa grillé en février, remplacé début juin) ont ralenti le projet. Cependant, ces problèmes ont pu être minimisés en modifiant l'ordre des tâches réalisées par rapport au diagramme de GANTT initial.

Nous avons cette année été soutenus par le BQP, mais le rapport fourni par le jury du BQP a mis en évidence que l'existence de deux PE fabriquant chacun une minifusée n'est pas viable sur le long terme [9]. Peut-être est-il possible de continuer à travailler sur des problématiques de communication sans fil à longue portée, mais sur des projets différents, tels que des CanSat⁵, des CubeSat⁶. Il serait également intéressant de réfléchir sur la transmission des savoirs dans le cadre de ce PE.

7.2 Perspectives

La communication sans fil à longue portée est bien avancée, mais il reste du travail à faire pour optimiser les transferts de données entre la fusée et la base au sol.

Comme c'est le cas depuis plusieurs années, le CLC entend encore proposer des PE pour les années à venir. Leurs objectifs seront de constamment améliorer les minifusées du CLC et de proposer de nouvelles expériences à incorporer aux minifusées. C'est pourquoi la transmission de données d'une année sur l'autre est primordiale, c'est elle qui permet d'envisager des perspectives d'optimisation et de perfectionnement.

Ainsi, l'une des ambitions de ce projet a été également d'apporter des améliorations qui pourront être utiles aux prochains projets se basant sur la même structure de minifusée. L'idéal serait de confirmer que le programme de détection d'apogée fonctionne. Ainsi, les projets futurs pourraient s'affranchir de tous les aspects concernant strictement le "vol" de la fusée pour se concentrer sur ses accessoires et les mesures à faire à bord. Cela permettrait de gagner un certain temps tout au long du projet pour ne s'attacher qu'à l'un des aspects de la fusée.

Un soin particulier sera apporté à l'accompagnement des futurs membres des PE fusée, car il s'agit certainement des PE nécessitant un encadrement notamment technique des plus poussés.

5. Dispositif autonome de très petite taille (celle d'une canette de 33 cL) qui effectue des expériences scientifiques à très haute altitude.

6. Nanosatellite standardisé en 1999.

8 Conclusion

Bien que le projet ne sera terminé qu'une fois le vol nominal de la fusée effectué, il est déjà possible de dresser un bilan du projet. La minifusée, Balistey, qui doit être produite, assemblée et sera lancée à Tarbes en juillet. Bien que les cartes électroniques ne soient pas toutes fonctionnelles, en particulier celle liée à la réception des données au sol, l'ensemble des cartes électroniques est capable d'assurer un vol nominal à la fusée, de réaliser des mesures pendant le vol, et de les émettre en communication sans fil. Ce projet, pluridisciplinaire, a permis à l'ensemble de l'équipe de découvrir de nouveaux outils, notamment la programmation de PIC en C, qui n'est enseignée à l'École Centrale que dans des électifs de troisième année. Bien que l'équipe n'eût pas les compétences pour parfaitement conduire ce projet, son investissement dans le projet a permis de concevoir une fusée dont les caractéristiques par rapport au cahier ont été revues à la baisse, mais dont les spécifications sont suffisantes pour participer au C'space.

Ce PE, très exigeant, prépare excellemment bien au métier d'ingénieur, aussi bien du point de vue du management d'équipe, de la nécessité de se former rapidement à l'utilisation de nouveaux outils, et de la rigueur scientifique qu'il requiert. Il a donné à ses membres des atouts pour la suite de leur parcours, en leur permettant de maîtriser des outils que peu de centraliens rencontreront dans leur parcours, et les a armés pour une poursuite de projets aérospatiaux.

9 Bibliographie

Références

- [1] ADMIN. *MPU-9250 et Arduino*. <http://www.lucidarme.me/?p=5070>. Jan. 2015.
- [2] CNES. *C'SPACE 2017 : Le rendez-vous espace étudiants*. <https://jeunes.cnes.fr/fr/cspace-2017-le-rendez-vous-espace-etudiants>. avril 2017.
- [3] *cnes, le 2ème budget au monde*. <https://cnes.fr/fr/web/CNES-fr/11507-le-2eme-budget-au-monde.php>. juin 2017.
- [4] *documentation des cartes électroniques standardisées*. Centrale Lyon Cosmos. juin 2017.
- [5] INVENSENSE. *MPU-3000/MPU-3050 Motion Processing Unit Product Specification Rev 2.9*. <https://store.invensense.com/datasheets/invensense/MPU-3000A.pdf>. Nov. 2011.
- [6] *Les propulseurs des clubs aérospatiaux*. <https://www.planete-sciences.org/espace/IMG/pdf/propulseur>. juin 2017.
- [7] *LoRa Alliance*. <https://www.lora-alliance.org/What-Is-LoRa/Technology>. mai 2017.
- [8] bénévoles et permanents PLANÈTE SCIENCES. *cahier des charges Minifusées*. https://www.planete-sciences.org/espace/IMG/pdf/cahier_des_charges_minif_pro2x.pdf. avril 2017.
- [9] *RésultatsBQP2016-2017*. <https://projets.ec-lyon.fr/resource/open/file/5152>. mai 2017.
- [10] Planète SCIENCES. *Le Vol de la Fusée, Stabilité et Trajectographie*. juillet 2008.
- [11] *Trajectoire-Secteur espace*. <https://www.planete-sciences.org/espace/Ressources/Trajectoire/>. juin 2017.
- [12] WIKIPÉDIA. *Microchip Technology*. https://fr.wikipedia.org/wiki/Microchip_Technology. juin 2017.

Glossaire

- BQP** Bonus Qualité Projet. Financement supplémentaire pour les PE auxquels les 300€ de départ ne seraient pas suffisant. Les sommes allouées dépendent des projets, de leur solidité, et de leur motivation démontrés au travers d'un oral et d'un rapport écrit . 3, 15
- C'space** Un des rassemblements organisés par Planète Sciences. Tous les projets étudiants aérospatiaux y sont conviés. On y lance des fusées, largue des ballon-sondes, parachute des CanSat.. 3, 7, 13, 15
- CAO** Conception Assistée par Ordinateur . Désigne un ensemble d'outils informatiques tels que des logiciels, permettant de modéliser des pièces dans le but de réaliser des simulations ou de les fabriquer. . 12, 48
- CLC** Centrale Lyon Cosmos. Association centralienne, commanditaire de nombreux projets pour les étudiants de l'école. 2, 3, 7, 9, 11, 48
- CNES** Centre national d'études spatiales. Établissement public français en charge du programme spatial de la France. . 30
- LoRa** Long Range (communication sans fil). Nouveau standart de communication sans fil, permettant une transmission de données à longue distance, à faible coût énergétique et financier. À vocation à occuper une place grandissante dans le cadre de l'internet des objets. [7]. 2, 7, 13, 15, 44
- minifusée** petite fusée, dont les dimensions répondent au cahier des charges fournit par planète-sciences. [8]. 2, 12
- PE** Projet d'Etude. 2, 7, 11–13, 41
- Planète Sciences** Association à but non lucratif qui encadre de nombreux projets étudiants dans de nombreux domaines et organise des rencontres autour de ces projets . 2, 3, 13, 15, 18, 47
- RCE** Rencontre des Clubs Espace. Journées de rencontre organisées par Planète Sciences ayant pour but de faire le point sur l'avancement du projet avec des bénévoles de l'association. Au nombre de trois dans l'année, elles se déroulent en week-end à Paris.. 3, 9

10 Annexe : code en C des cartes programmables

10.1 Carte séquenceur

```

#include <18F2685.h>
#include "can-18xxx8.c"
#include delay(clock=20M)
#include HS,NOPROTECT,NOLVP,NOWDT
#include rs232(baud=57600,xmit=PIN_C6,rcv=PIN_C7,PARITY=N,TIMEOUT=250)
//#define CARTE_ID 100
//#define CARTE_destinataire 200 //à vérifier
#define BUZZER (PIN_A5)
#define decollage 128
#define JACK (PIN_C0)
#define LED_EXTERNE (PIN_C1)
//#define LED1 (PIN_A2)
#define LENT 700
#define RAPIDE 100
#define SERVO (PIN_B1)
#define apogee 99
void blink(unsigned int pin, unsigned int32 periode)
{
output_high(pin);
delay_ms(periode);
output_low(pin);
delay_ms(periode);
}
void buzz()
{
output_high(BUZZER);
delay_ms(1000);
output_low(BUZZER);
delay_ms(1000);
}

```

```

void ouverture_servo()
{
    output_high(SERVO);
    delay_us(1000);
    output_low(SERVO);
    delay_ms(20);
}

void fermeture_servo()
{
    output_high(SERVO);
    delay_us(1950);
    output_low(SERVO);
    delay_ms(20);
}

void main()
{
    struct rx_stat ma_stat;

    int32 seq=200;    // adresse où l'apogée est signalée
    int32 imu=100;    // adresse où l'arrachage du jack est signalée
    int8 rx_len = 2,i;
    int8 in_data[2]={0,0};
    int8 out_data[2];

    setup_spi(SPI_MASTER|SPI_L_TO_H|SPI_XMIT_L_TO_H|SPI_CLK_DIV_4); //On utilise le microproc
    setup_timer_0(RTCC_INTERNAL|RTCC_DIV_2);
    can_init(); // initialisation du CANBUS
    can_set_baud(); //La fusée est alimenté, on buzz 2 fois court
    for(i=0;i<2;i++)
    {
        buzz();

        //Initialisation de la trappe : on envoie un message pour signaler à la partie haute de la fusée
        //de fermer le servomoteur
    }
}

```

```

}
for(i=0;i<50;i++)
{
    fermeture_servo();
}
delay_ms(500);
//On attend le Jack
while(!input(JACK)) // on attend le jack
{
    blink(LED_EXTERNE,LENT);
}
//Une fois le Jack mis, on buzz une fois pour signaler que la fusée est
//prête au lancement
output_high(BUZZER);
delay_ms(500);
output_low(BUZZER);
//Tant qu'il y a le Jack, les 2 LED clignotent rapidement pour signaler
//que la fusée est en attente
while(input(JACK))
{
    blink(LED_EXTERNE,RAPIDE);
}
    buzz());
//Décollage : le buzzer reste allumée pour récupérer au mieux la fusée
//on coupe les led : servent à rien. à décaler à après l'ouverture para
output_high(LED_EXTERNE);
output_high(BUZZER);
out_data[0] = decollage;
for (i=0;i<10;i++)
{
    if can_tbe()

```

```

{
    can_putd(imu,&out_data[0],rx_len,1,1,0); // on signale le décollage de la fusée
    delay_ms(10);
}
can_abort();
}
//On attend 7 secondes avant de déclencher l'ouverture de la trappe par
//sécurité : le séquenceur prend le relai sur l'accéléromètre (en cas de
//disfonctionnement)
for (i=1;i<700;i++)
{
    if can_kbhit()
    {
        can_getd(seq , &in_data[0], rx_len, ma_stat);
        delay_ms(5);
        //read_canbus = true;
        if (in_data[0] == apogee )
        {
            goto exit_loop1;
        }
    }
    delay_ms(5); // 700*10 ms = temps interruptible de 7s
}
exit_loop1:
for(i=0;i<50;i++)
{
    ouverture_servo();
}
}

```


10.2 Carte capteur : centrale inertielle

```

#include <18F2685.h>

#define ICD=TRUE

#include "can-18xxx8.c"

#define delay(clock=20M)

#define i2c(master,SCL=PIN_C3,SDA=PIN_C4)

#define fuses HS,NOPROTECT,NOLVP,NOWDT

#define rs232(baud=57600,xmit=PIN_C6,rcv=PIN_C7,PARITY=N,TIMEOUT=250)

#define decollage 128

#define GYROR 0xD1

#define GYROW 0xD0

#define G_PWR_MGM 0x3E

#define G_SMPLRT_DIV 0x15

#define G_DLPF_FS 0x16

#define G_INT_CFG 0x17

#define apogee 99

#define i 80

#define theta 0

#define pas_temps 0.001

#define condition 0

//int data;

void write(ad, val)//ad : le registre; val : à mettre dans le registre
{
    i2c_start();
    i2c_write(0xD0); //adresse de l'accéléromètre en écriture = ad+0
    delay_us(20);
    i2c_write(ad); //adresse de la donnée
    delay_us(20);
}

```

```

i2c_write(val);
delay_us(20);
i2c_stop();
//delay_ms(1);
}

```

```

int read(ad)
{
int data;
data = 0;
i2c_start();
i2c_write(0xD0); //adresse de l'accelero en ecriture = ad+0
delay_us(20);
i2c_write(ad); //adresse de la donnee
delay_us(20);
i2c_stop();
i2c_start();
i2c_write(0xD1); //ad de l'accelero en lecture = ad+1
delay_us(20);
data = i2c_read(); //Read Next
delay_us(5);
i2c_stop();
//delay_ms(1);
return data;
}

```

```

void initGyro()
{
int va;
write(0x3E,00); //mise sur l'oscillateur interne
write(0x15, 0x07); //sample rate 125Hz

```

```

write(0x16,0x19); // +/- 2000 dgrs/sec

va = read(0x15);

// writeTo(GYROW, G_INT_CFG, 0x00); //pas d'interruptions

va = read(0x16);

}

void main()

{

struct rx_stat ma_stat;

int32 imu = 100;

int32 seq = 200;

int32 lora = 300;

int8 rx_len = 2,i;

int8 out_data[2];

int8 in_data[2]={0,0};

while(1) //attente decollage

{

if can_kbhit()

{

can_getd(imu , &in_data[0], rx_len, ma_stat);

delay_ms(5);

//read_canbus = true;

if (in_data[0] == decollage )

{

goto exit_loop1;

}

}

}

delay_ms(5); // 200*10 ms = temps interruptible de 2s

```

```

}

exit_loop1:
{

    int x, y, z;
    int xh , xl, yh, yl, zh, zl;
    int i;

    delay_ms(1000);

    xh = read(0x00);

    initGyro();

    while(1) //lecture données
    {
        xh = read(0x1D ); //read the gyro data from the ITG3200
        xl = read(0x1E );
        x = ((xh << 8) | xl); //+ g_offx;
        yh = read(0x1F );
        yl = read(0x20 );
        y = ((yh << 8) | yl); //+ g_offy;
        zh = read(0x21 );
        zl = read(0x22 );
        z = ((zh << 8) | zl); //+ g_offz;

        theta = theta + pas_temps * z
        if cos(theta) > sin(theta)
    {
psi2 = x/cos(theta)

```

```

    }
if cos(theta) <= sin(theta)
{
psi2 = -y/sin(theta)
    }

i = i + pas_temps * psi2

if i <= 0
{
condition = 1
    }

    out_data[0] = z;
    for (i=0;i<10;i++) //on envoie a la telem les donnees
{
if can_tbe()
{
    can_putd(lora,&out_data[0],rx_len,1,1,0);
    delay_ms(10);
}
can_abort();
}

if (condition==1)// detection apogee
{
    out_data[0] = apogee;

```

10.3 Carte capteur : communication sans fil à longue portée

Les programmes suivants n'ont pas entièrement été implémentés par le groupe du projet. Ils ont été créés dans le cadre d'un stage proposé par le laboratoire EEA de l'École Centrale de Lyon, puis ils ont été modifiés pour satisfaire aux besoins du projet.

10.3.1 fichier main.c

```

#DEVICE ICD=true

#use delay(clock=20M)

#fuses HS,NOPROTECT,NOLVP,NOWDT

#use rs232(baud=57600, xmit=PIN_C6, rcv=PIN_C7, PARITY=N, TIMEOUT=250)

#define LED1 PIN_C0

#define RN2483_RST PIN_B4

#include "global_vars.h"

#include "communication.c"

#include "get_param.c"

#include "set_param.c"

#include "default_param.c"

#int_rda

void rcv_uart(){

process_int_uart();

}

void main(){

//setup_oscillator(OSC_8MHZ);

disable_interrupts(INT_AD);

enable_interrupts(INT_RDA);

enable_interrupts(global);

reset_RN2483();

zero_in_data();

get_ver() ;

set_default_param();

delay_ms(200);

mac_pause();

printf("radio set pwr 14\r\n");

delay_ms(200);

while(TRUE) {

mac_pause();

delay_ms(200);

```



```
output_high(LED1);  
printf("radio tx 0123456789\r\n");  
//delay_ms(200);  
output_low(LED1);  
}  
}
```

10.3.2 fichier `get_param.c`

```

void get_ver(){
delay_ms(200);
printf("sys get ver\r\n");
    if(uart_line_received==true) //if new line received{
    uart_line_received=false;
    //}
}

int8 get_mod(){
delay_ms(200);
printf("radio get mod\r\n");
    if(uart_line_received==true) //if new line received{
    uart_line_received=false;
        if((in_data[0]==108)&&(in_data[1]==111)&&(in_data[2]==114)&&(in_data[3]==97)){
// if answer= lora
        return 1;}
        return 0;//}
}

int8 get_freq(){
delay_ms(200);
printf("radio get freq\r\n");
    if(uart_line_received==true) //if new line received{
    uart_line_received=false;
        if((in_data[0]==56)&&(in_data[1]==54)&&(in_data[2]==56)) // if answer= 868
        {
        return 1;}
        return 0;
}

int8 get_sf(){

```

```

delay_ms(200);
printf("radio get sf\r\n");
    if(uart_line_received==true) //if new line received
    {
        uart_line_received=false;
        if((in_data[0]==115)&&(in_data[1]==102)&&(in_data[2]==49)&&(in_data[3]==50)){
// if answer= sf12
return 1;}
        return 0;}
    }

```

```

int8 get_bw(){
delay_ms(200);
printf("radio get bw\r\n");
    if(uart_line_received==true) //if new line received{
        uart_line_received=false;
        if((in_data[0]==49)&&(in_data[1]==50)&&(in_data[2]==53)) // if answer= 125
        {
            return 1;}
        return 0;
    }

```

```

int8 get_cr()
{
delay_ms(200);
printf("radio get cr\r\n");
    if(uart_line_received==true) //if new line received{
        uart_line_received=false;
        if((in_data[0]==52)&&(in_data[1]==47)&&(in_data[2]==53)) // if answer= 4/5
        {
            return 1;}
    }

```

```

    return 0;
}

int8 get_prlen() {
delay_ms(200);
printf("radio get prlen\r\n");
    if(uart_line_received==true) //if new line received
    {
    uart_line_received=false;
    if(in_data[0]==56) // if answer= 8
    {
    return 1;}
    return 0;}
}

```

```

int8 get_pwr(){
delay_ms(200);
printf("radio get pwr\r\n");
    if(uart_line_received==true) //if new line received{
    uart_line_received=false;
    if(in_data[0]==52) // if answer= 1
    {
    return 1;}
    return 0;
}

```

```

int8 get_bitrate(){
delay_ms(200);
printf("radio get bitrate\r\n");
    if(uart_line_received==true) //if new line received

```

```

{
uart_line_received=false;
    if((in_data[0]==53)&&(in_data[1]==48)&&(in_data[2]==48)&&(in_data[3]==48)){
// if answer= 5000
    return 1;}
    return 0;}
}

int8 get_crc(){
delay_ms(200);
printf("radio get crc\r\n");
    if(uart_line_received==true) //if new line received
    {
uart_line_received=false;
        if((in_data[0]==111)&&(in_data[1]==110)) // if answer= on
        {
            return 1;}
            return 0;}
    }
}

```

10.3.3 fichier `default_param.c`

```
void set_default_param()
{
set_mod('l');    //communication en mode lora

set_freq(8681); //fréquence d'émission 868.1 MHz

set_sf(12);     //facteur de propagation de 12

set_bw(125);    //bande passante 125 kHz

set_pwr(1);     //puissance du signal 1 dB

set_crc(1);     //en-tête CRC active

set_prlen(8);   //en-tête CRC de 8 bits

set_bitrate(5000); //débit binaire 5000 bps
}
```


10.3.4 fichier communication.c

```
void process_int_uart(){
in_data[uart_cptr]=getc();

if (in_data[uart_cptr]==10) //if LF (EOL)
{
uart_cptr=255;
uart_line_cptr++;
uart_line_received=true;
}
uart_cptr++;
}
```

```
void reset_RN2483(){
output_low(RN2483_RST);
delay_ms(1);
output_high(RN2483_RST);
delay_ms(1);
}
```

```
void zero_in_data(){
int8 i;
for (i=0;i<100;i++){
in_data[i]=0;
}
}
```

```
void mac_pause(){
delay_ms(200);
printf("mac pause\r\n");
}
```

10.3.5 fichier `set_param.c`

aire (bit rate) varie de "0" à "65535"

et ne peut être réglé qu'en mode fsk

Par défaut bitrate: "5000"

*/

```
int8 val_mod;
```

```
val_mod = get_mod();
```

```
delay_ms(200);
```

```
printf("radio set bitrate 65535\r\n");
```

```
delay_ms(50);
```

```
get_bitrate();
```

```
}
```

```
void set_crc(int8 val_crc)
```

```
{
```

```
/* val_crc doit être de type int8
```

```
L'en-tête crc peut être dans deux état "on" ou "off";
```

```
Pour le mode "on" val_crc ="1" et pour le mode "off" val_crc = "0"
```

```
Par défaut crc: "on"
```

```
*/
```

```
if(val_crc==1){
```

```
delay_ms(200);
```

```
printf("radio set crc on\r\n");
```

```
delay_ms(50);
```

```
}
```

```
else{
```

```
if(val_crc==0){
```

```
delay_ms(200);
```

```
printf("radio set crc off\r\n");
```

```
delay_ms(50);
```

```
}
```

```

}
get_crc();
}

void set_prlen(unsigned short val_prlen)
{
/* val_prlen doit être de type unsigned short
   La taille de l'en-tête crc (crc header) varie de "0" à "65535"
   Par défaut crc: "8"
*/
delay_ms(200);
printf("radio set prlen %u\r\n",val_prlen);
delay_ms(50);
get_prlen();
}

```

10.3.6 Check-List de rapport de Projet d'Etude

Check-list de rapport de Projet d'Etudes
A remplir par les rédacteurs (élèves)
et à insérer en dernière page du rapport

A développer

Renseigner la case par le nom du responsable, ou la date ou une simple croix lorsque la vérification a été faite.

Vérification présence	Vérification qualité
-----------------------	----------------------

Contenu

Résumé en français	X	
Résumé en anglais	X	
Table des matières	X	
Table des figures	X	
Introduction	X	
Conclusion générale	X	
Bibliographie	X	
Citation des références dans le texte	X	

Forme

Vérification orthographe	X	
Pagination	X	
Homogénéité de la mise en page	X	
Lisibilité des figures	X	