

Supélec

# Rapport de projet

Projet long



*Station sol*

# *Sommaire*

<b>Introduction .....</b>	<b>3</b>
<b>I. Structure et mécanique .....</b>	<b>4</b>
A. Présentation.....	5
B. Identification et résolution des anomalies .....	6
<b>II. Contrôle du suivi de la station sol .....</b>	<b>7</b>
C. Matériel .....	8
1. Chaîne d'information .....	8
2. Chaîne d'énergie .....	10
3. Synthèse .....	11
D. Logiciel .....	12
1. TLE .....	12
2. Date et heure .....	13
3. WiFi .....	13
4. Motorisation .....	14
5. Affichage.....	14
6. Configuration .....	16
7. Général .....	17
E. Réalisation .....	18
F. Essai .....	21
G. Résumés.....	23
H. Améliorations.....	24
<b>III. Boitier .....</b>	<b>25</b>
A. Partie basse.....	26
B. Partie haute.....	29
C. Réalisation .....	31
<b>IV. Réception et émission .....</b>	<b>34</b>
A. Bilans de liaison.....	35
1. Lien montant.....	35
2. Lien descendant .....	41

<b>B. Design de l'antenne .....</b>	<b>42</b>
1. Cahier des charges .....	42
2. Choix.....	43
3. Alimentation du réseau.....	45
4. Essai.....	50
5. Prototype final de l'antenne.....	51
<b>C. Circuit radiofréquence .....</b>	<b>53</b>
1. LNA.....	53
2. Filtre de réception.....	53
3. Amplificateur de puissance .....	54
4. Switch.....	54
5. Tranceiver.....	55
6. PCB .....	56
<b>D. Bilan.....</b>	<b>65</b>
<b>Conclusion .....</b>	<b>66</b>
<b>Remerciements .....</b>	<b>67</b>
<b>Annexes .....</b>	<b>68</b>
A. Guide d'utilisation de l>IDF de l'ESP32 .....	68
B. Bilan de liaison avec la feuille de calcul AMSAT-UK .....	69
C. Principe Télécommande et Télémétrie : .....	70
D. Antenne hélice axiale .....	71
E. Antenne Crossed-Yagi.....	79
F. Modulation .....	81
1. La modulation 2-FSK.....	81
2. La modulation GMSK.....	82
3. Performance .....	84
G. Protocole de communication .....	85
<b>Sources .....</b>	<b>87</b>

# *Introduction*

Supsat est une association fondée en 2012 par trois étudiants de Supélec à Rennes dont le but est, à terme, d'envoyer un nanosatellite (10\*10\*10cm, environ 1W, environ 1kg) dans l'espace, celui-ci étant entièrement conçu par des étudiants de Supélec.

L'objectif de Supsat est triple :

- *Pédagogique* : la conception d'un système aussi complexe qu'un satellite ne peut être que motivant et excitant pour un élève ingénieur qui a envie de s'investir dans un projet concret. Soutenu par des personnes issues de la recherche et de l'industrie, les étudiants conçoivent le satellite de A à Z. Ainsi, de la conception à l'assemblage des différentes parties, de l'assemblage à la vérification du design et enfin de la phase de tests au lancement réel ainsi qu'à son contrôle, les étudiants ont l'occasion de mettre en pratique et de confronter ce qu'ils ont appris sur un projet concret à la pointe de la technologie.
- *Scientifique* : bien que Supsat soit un nanosatellite, il embarque un micro-laboratoire qui est destiné à effectuer des tests relatifs à la reconfiguration partielle de FPGAs :
  - des tests de durcissement logiciel contre les SEEs (Single Event Effects)
  - des tests de reconfiguration sans interruption de service
- *Technologique* : des applications quant à l'utilisation de cette technologie peuvent apparaître si les expériences portent leurs fruits.

Dans le cadre de la réalisation du projet de Supsat, nous sommes chargés de la conception et de la construction de la station sol afin de communiquer avec le futur cubesat.

La station doit pouvoir le suivre pour garantir une meilleure transmission. Elle doit également être capable de recevoir les données envoyées par le satellite lors du survol de la station. Finalement, elle doit aussi pouvoir émettre pour communiquer des instructions au cubesat.

# I. Structure et mécanique

La structure doit supporter l'antenne de communication et disposer d'une motorisation permettant de l'orienter dans une direction donnée.

Nous reprenons la structure construite durant le projet dédié de l'an dernier. Elle s'inspire du projet libre *SatNOGS*. Elle est constituée d'une structure en aluminium, de deux moteurs pas-à-pas et d'un système de transmission.



Figure 1. *Structure de la station sol héritée du projet de l'an dernier*

Cependant, des dysfonctionnements ont été décelés lors de nos premiers essais de prise en main. Nos premiers objectifs furent donc de rendre cette structure fonctionnelle.

## A. Présentation

Chacun de moteur pas-à-pas contrôle un axe : azimut ou élévation.

L'azimut permet d'orienter l'antenne sur le plan horizontal. Il s'agit d'un angle compris entre  $0^\circ$  et  $360^\circ$  avec le Nord géographique comme référence.

L'élévation est l'orientation de l'antenne dans le plan vertical. Il mesure l'angle entre l'antenne et l'horizontal. Il est compris entre  $0^\circ$  et  $90^\circ$  avec l'horizon comme référence.

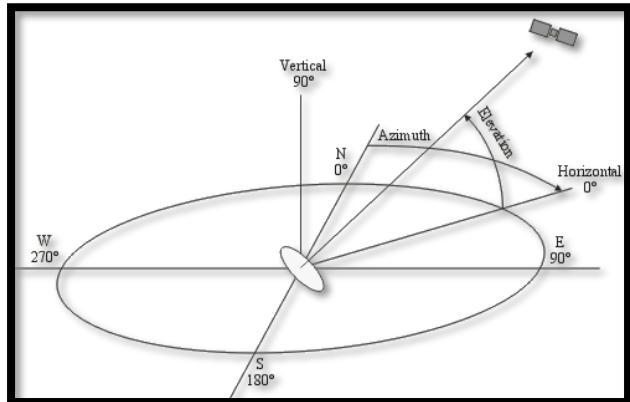


Figure 2. Angles d'azimut et d'élévation

La plupart des pièces sont imprimés en 3D en PLA.

Les moteurs pas-à-pas effectuent une rotation en 200 pas. Le système de transmission est composé d'une poulie-courroie de rapport  $\frac{20}{36}$  et d'une roue de 30 dents avec une vis à un filet ce qui nous donne un rapport de transmission total de  $\frac{1}{54}$ .



Figure 3. Chaine de transmission pour chaque axe

La précision en sortie du moteur est de  $1,8^\circ$ . Ainsi, en sortie de la chaîne de transmission, la précision est de  $\frac{1.8}{54} = 0,03^\circ$ .

Cependant, les jeux mécaniques dégradent grandement cette précision. Nous allons essayer de les minimiser.

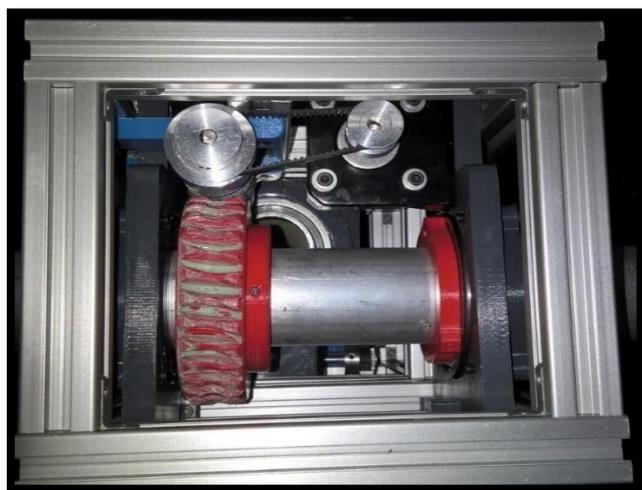


Figure 4. Aperçu du mécanisme (haut de la station et côté de la partie inférieure)

## B. Identification et résolution des anomalies

La structure présentait plusieurs défauts. En effet, les courroies étaient trop tendues et la roue-vis était bloquée, empêchant les axes de tourner. Nous avons aussi remarqué la présence de jeux dans la structure et une certaine instabilité. Ceci est problématique étant donnée la volonté de diriger l'antenne de manière assez précise.

Pour déterminer l'origine des problèmes, nous avons dû démonter la structure. Cela nous a permis de bien cibler les pièces qui posaient problème à la mise en marche de la station.

La solution à ses différents problèmes fut simplement de réimprimer les pièces et d'ajuster certaines pièces avec des outils prêtées par le service technique de l'école. Nous avons donc redimensionné certaines pièces avec le logiciel *SolidWorks*.

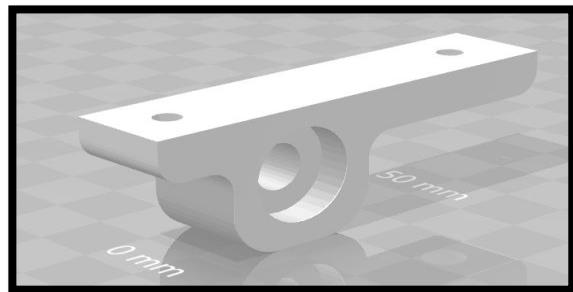


Figure 5. Pièce modifiée sous SolidWorks

Nous avons ensuite réaligné certaines pièces et ajouter des rondelles de calage pour réduire les jeux.

Nous avons pu mesurer les jeux après nos modifications.

Axes	Jeu
Azimut	1,3°
Élévation	1,6°

Figure 6. Mesures des jeux mécaniques

Nous nous sommes aussi posé la question des matériaux utilisés concernant les pièces provenant de l'imprimante 3D. Leur usure plus rapide que des pièces mécaniques pourrait poser des problèmes à l'avenir. Nous constaterons la durabilité des matériaux sur le long terme.

Finalement, seule l'instabilité du pied reste non résolue. Cependant, la station est destinée à être ancrée sur le toit de Supélec. Ce problème sera donc résolu lors de l'installation finale de la station.

## II. Contrôle du suivi de la station sol

Nous allons à présent détailler le matériel utilisé pour contrôler la structure décrite précédemment ainsi que l'implémentation des différentes fonctionnalités nécessaires au suivi d'un satellite.

L'objectif de cette partie est de concevoir un système autonome et fiable capable de pointer l'antenne vers un satellite défini en contrôlant les moteurs orientant la structure en azimut et en élévation.

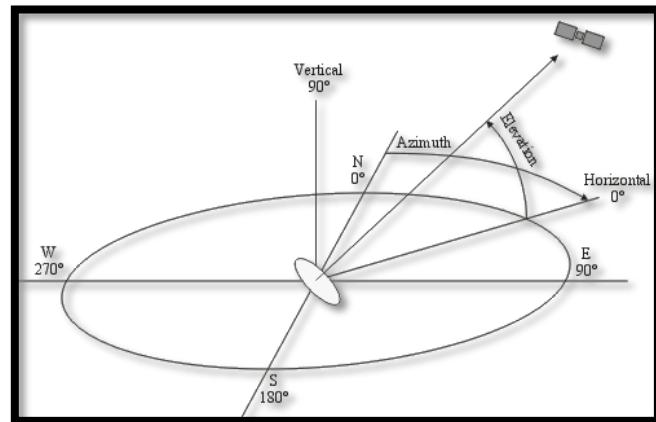


Figure 7. Angles d'azimut et d'élévation

## C. Matériel

Nous allons voir les différents composants retenus à la fin de ce projet.

### 1. Chaine d'information

#### Microcontrôleur : ESP32

Notre système s'articulera autour de l'ESP32, un microcontrôleur de faible coût fabriqué par Espressif. Sa puissance de calcul dépasse ce dont nous nécessitons a priori, mais cela laisse des possibilités d'évolutivité sans changer de matériel.



Figure 8. *ESP32, kit de développement*

Nous avons programmé ce microcontrôleur en C++ à l'aide de l'IDE Eclipse et du SDK officiel ESP-IDF. Nous utilisons l'OS FreeRTOS permettant de paralléliser des tâches.

Nous avons favorisé cet environnement à Arduino afin d'avoir plus de libertés dans l'implémentation du suivi de satellite et pour assurer la compatibilité d'autres bibliothèques ne supportant pas certaines conventions d'Arduino ou nécessitant la STL (Standard Template Library) du langage C++ indisponible sous Arduino.

De plus, l'ESP32 dispose d'un RTC (Real-Time Clock) interne permettant de déterminer la date et l'heure avec une précision importante. Cela évite l'utilisation d'un module supplémentaire de type DS1307.

Dernier avantage, il peut se connecter par WiFi sans l'ajout de module externe.

#### Contrôle des moteurs : Adafruit Motor Shield

Les moteurs pas-à-pas de la structure permettent d'orienter l'antenne en azimut et en élévation afin de pointer le satellite. Nous utilisons le module Arduino utilisé l'an dernier, en l'occurrence l'Adafruit Motor Shield v2.3.

Dans la configuration utilisée, il permet de contrôler 2 moteurs pas-à-pas via I2C.

Il peut fonctionner en 3,3V via l'utilisation d'un jumper sur la carte. Son adresse sur le bus I2C est 0x60.

Il peut délivrer jusqu'à 1,2A à chaque moteur.

Les moteurs sont alors contrôlés par 4 fils distingués par leur couleur : rouge (R), gris (G), vert (V) et jaune (J).



Figure 9. *Adafruit Motor Shield*

### Position initiales des axes : TLE4905L

Etant donné que nous utilisons des moteurs pas-à-pas, nous devons déterminer une position initiale pour chacun des axes afin de pouvoir contrôler leur orientation par la suite.

En plaçant un capteur à effet Hall et un aimant par axe, nous pouvons déterminer ces positions de référence.

Ils fonctionnent en 5V, et donnent un signal binaire selon la présence d'un champ magnétique. Si l'aimant est petit et bien orienté, le capteur signale au microcontrôleur que l'axe est aligné.

On utilise des capacités de 47nF pour lisser le signal de sortie et un pont diviseur de tension pour ramener ce signal à 3,3V afin de ne pas endommager le microcontrôleur.

Chacun des capteurs est lié à une LED indicative de son état.

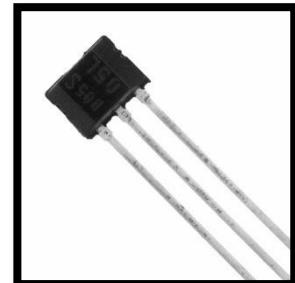


Figure 10. Capteur à effet Hall



Figure 11. Ecran OLED

Afin de contrôler le fonctionnement de la station, nous utilisons un petit écran OLED de 128x64 pixels avec une diagonale de 0,96".

Il permet l'affichage de 8 lignes de 16 caractères. Les deux premières sont jaunes, les suivantes sont bleues. Cela est suffisant pour permettre un diagnostic rapide de l'état de la station.

Il communique en I2C avec l'adresse 0x3D.

### Circuit radiofréquence

Le circuit RF est chargé de la réception du signal de l'antenne et de son traitement. Il est également en charge de l'émission. Il sera abordé en détail par la suite.

Cependant, le module d'émission et réception présent sur ce circuit communique via SPI pour transmettre les informations reçus du satellite, recevoir les données à émettre et être commandé. De plus, d'autres composants du circuit RF (LNA, HPA et commutateur) sont également commandables.

Nous avons donc décidé que l'ESP32 chargé du contrôle du suivi du satellite communiquerait avec la carte RF. Ses performances le permettant, notamment la possibilité de parallélisation, il sera en charge, lors de la phase de suivi, de contrôler le circuit RF via un bus SPI et une sortie numérique appelée RF\_CONTROL par la suite.

Nomenclature	Description
+3.3V	Alimentations
+5V	
RF_CONT	Sortie numérique de contrôle
MOSI	SPI entre ESP32 et SI4463 (carte d'émission/réception)
MISO	
SCK	
SS	
GND	Masse

Figure 12. Description du connecteur entre circuit de contrôle et circuit RF

## 2. Chaine d'énergie

Nous n'avons que peu modifié l'aspect énergétique de la station. En effet, nous utilisons toujours la batterie et le convertisseur DC/DC de l'an dernier.

Nous avons cependant refait les soudures de la plaquette du convertisseur qui nous semblaient hasardeuses, et avons ajouté un fusible et un interrupteur pour imiter le schéma d'utilisation fourni par le constructeur.

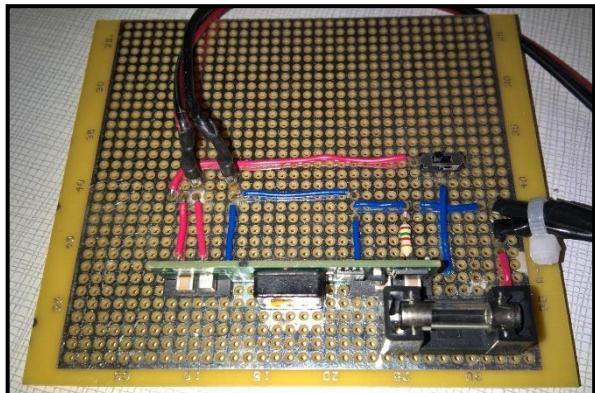


Figure 13. Plaquette avec convertisseur DC/DC

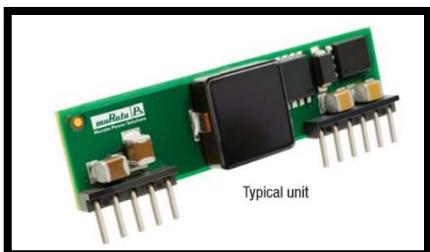


Figure 14. Convertisseur DC/DC

La régulation de 5V à 3,3V est assurée par le régulateur présent sur la carte de développement de l'ESP32. Il permet de débiter un maximum de 1A avec un bon rendement selon la datasheet de l'AMS1117 utilisé sur le devkit.

Etant donnée l'altitude du satellite de 400km, on peut déduire sa vitesse et donc sa durée de passage maximale. Nous obtenons théoriquement une durée de visibilité maximale de 10 minutes et quelques secondes. Nous considérerons donc un temps de fonctionnement théorique de 15 minutes car la station doit être initialisée quelques minutes avant le passage du satellite.

La capacité de la batterie étant d'environ 20Wh. En supposant une consommation moyenne supérieure à 1A, le rendement de la conversion est au minimum de 80% ce qui permet d'obtenir 16Wh d'énergie disponible. Soit, pour la durée de fonctionnement, une puissance moyenne de 60W.

Cette limite est approximative mais donne un ordre d'idée du dimensionnement des différents éléments de la station.

Nous disposons de 2 batteries, ce qui permet de les utiliser en alternance et d'être toujours opérationnels.

Tension	Courant instantané maximal	Courant moyen maximal
3,3V	1A	1A
5V	10A	10A

Figure 15. Résumé de la puissance approximative disponible

### 3. Synthèse

Ainsi, les différentes parties interagissent de la façon suivante :

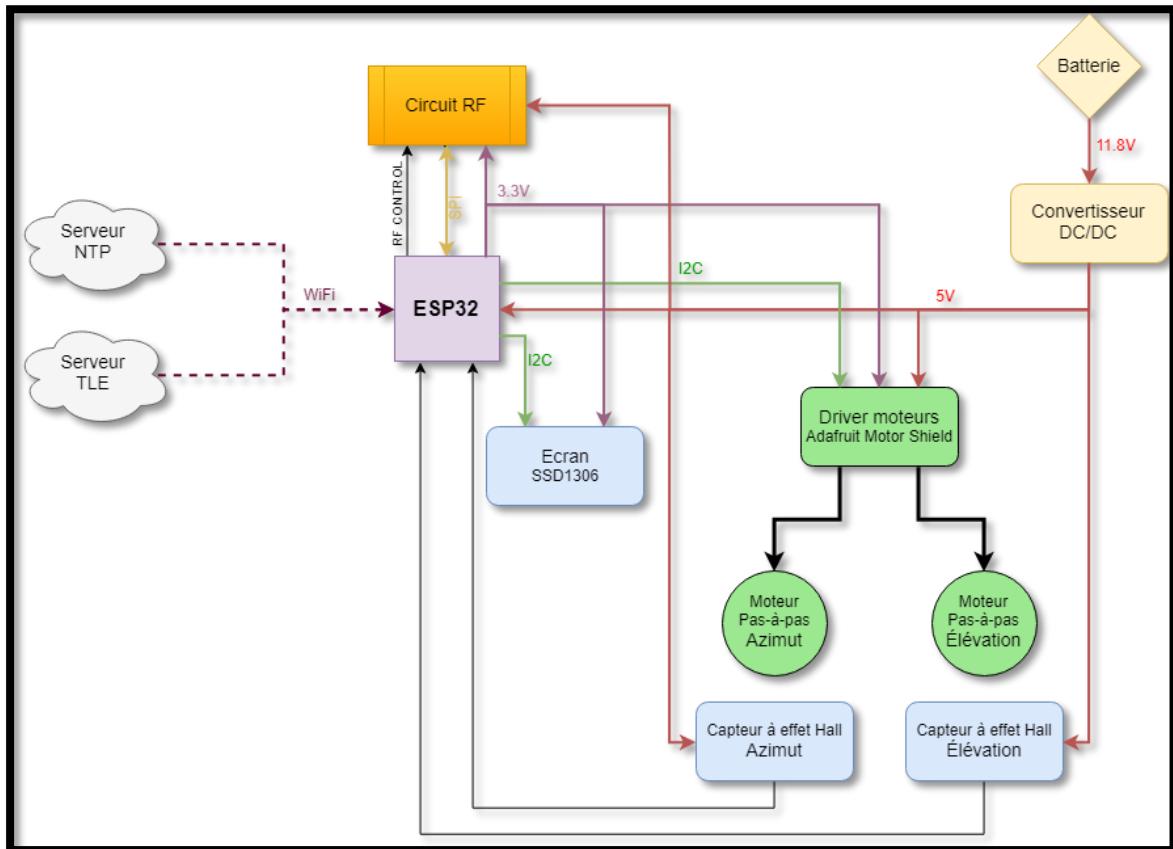


Figure 16. Résumé de l'architecture matérielle

## D. Logiciel

### 1. TLE

Afin de déterminer la position du satellite dans le ciel, nous utilisons les paramètres orbitaux. Ils peuvent aussi être appelés paramètres célestes ou TLE pour Two Lines Element. Ils décrivent la trajectoire de nombreux objets spatiaux tels que satellites, nano satellites, débris, etc...

Ils sont régulièrement mis à jour par les agences spatiales suite à leurs observations. Pour les cubesats, la NORAD (North American Aerospace Defense Command) les actualise une fois par jour. Ces informations sont disponibles à cette adresse : <http://celestrak.com/NORAD/elements/>

```
SWISSCUBE
1 35932U 09051B    18043.92525997 .00000102 00000-0 33982-4 0   9993
2 35932   98.5251 192.2353 0006949 224.9949 135.0686 14.56119658445318
```

Figure 17. TLE du SwissCube de l'EPFL

La première ligne permet d'identifier le satellite, notre programme isole les TLE du satellite recherché.

Ensuite, des algorithmes appelés *Simplified perturbations models* permettent de les décoder. Ainsi, à partir des TLE, de notre position géographique sur Terre (latitude, longitude et altitude), de la date et de l'heure, nous pouvons en déduire les angles d'azimut et d'élévation afin de pointer dans la direction du satellite. Différents algorithmes sont disponibles permettant la prise en compte de différents paramètres. En l'occurrence, nous utiliserons le plus répandu, le SGP4.

Cet algorithme présente une erreur d'environ 1km sur la position du satellite dans les 24h qui suit l'actualisation des TLE. Pour l'altitude de 400km de notre cubesat, cela nous donne une erreur angulaire de 0,008° environ.

Nous utiliserons l'implémentation de David Vallado de 2007 et porté sur ESP8266 par Hopperpop en 2015. Peu de modifications ont été nécessaires pour adapter cette librairie sur ESP32. Il est basé sur la description de l'algorithme par la NORAD en 1980. Ce code est très complet et détaillé. Nous n'utiliserons qu'une partie des fonctionnalités proposées. Il est sous licence MIT, donc libre et open-source.

Nous faisons 10 tentatives de collecte des TLE, si elles échouent toutes, le programme s'arrête et signale une erreur.

Les fichiers relatifs au traitement des TLE sont :

- *tle\_web.cpp* et *tle\_web.h* pour la récupération des TLE sur internet
- *tle.cpp* et *tle.h* pour l'utilisation de la librairie
- le dossier *sgp4* contenant la librairie

## 2. Date et heure

Le positionnement nécessite de connaître l'instant présent avec précision. Le RTC de l'ESP32 doit être actualisé après chaque mise hors tension.

Nous avons donc décidé d'actualiser le RTC à chaque démarrage de la station. Pour cela, nous allons utiliser le protocole NTP (Network Time Protocol) permettant d'interroger un serveur NTP via le port 123.

L'implémentation se base sur l'exemple donné par l'IDF sur github. Nous utilisons le serveur [pool.ntp.org](http://pool.ntp.org). Il est fiable car distribué sur un cluster de plusieurs centaines de serveurs en France.

Nous faisons 10 tentatives de synchronisation, si elles échouent toutes, le programme s'arrête et signale une erreur. Les fichiers relatifs au traitement du temps sont *rtc.cpp* et *rtc.h*.

Etant donnée l'altitude de 400km de notre futur satellite, il se déplacera à une vitesse de 7,7km/s. Si l'heure du RTC n'est pas précise, nous obtenons une erreur d'environ 0,06° par seconde d'écart.

## 3. WiFi

La connexion sans fil à Internet est requise pour la récupération de la date et de l'heure par NTP et la collecte des TLE.

Il est nécessaire d'avoir une connexion depuis l'école fiable et disposant des ports 123 et 80 ouverts. La connexion via *eduroam* n'ayant pas été fructueuse, la solution retenue est donc le réseau *supelec-invite*.

Nous devons donc nous connecter via le portail captif. Pour cela, nous avons observé les paquets avec *Fiddler 4* et imité cette connexion en reproduisant les requêtes HTTP. Ainsi, la requête permettant l'authentification est :

```
POST /reg.php HTTP/1.1
Host: 1.1.103.1
Connection: keep-alive
User-Agent: esp-idf/1.0 esp32
Content-Type: application/x-www-form-urlencoded
Content-Length: 32

username=LOGIN&password=PASSWORD
```

Figure 18. Requête POST permettant l'authentification sur supelec-invite

Le service logistique a pu nous fournir des identifiants pour nous connecter sans utiliser nos codes personnels. L'implémentation se base sur l'exemple donné par l'IDF sur github.

Nous attendons la réussite de la connexion au réseau pendant 12 secondes. Puis, nous faisons 10 tentatives d'authentification via le portail captif, si la connexion n'a pas réussi à la fin du temps imparti ou que toutes les tentatives d'identification échouent, le programme s'arrête et signale une erreur.

Les fichiers relatifs au WiFi sont :

- *wifi.cpp* et *wifi.h* pour la connexion au réseau
- *captive\_portal.cpp* et *captive\_portal.h* pour l'authentification au portail captif

## 4. Motorisation

Nous avons dû trouver une adaptation de la bibliothèque d'Adafruit pour l'IDF de l'ESP32. Nous utilisons celle de *ShellAddicted* trouvée sur github disponible sous licence BSD, donc libre.

Nous commandons chaque moteur avec des angles absolu. Le programme choisi le sens de rotation le plus court pour atteindre cette position. Il est également possible de le contrôler avec des angles relatifs ou des nombres de pas.

Pour détecter la position grâce aux capteurs à effet Hall, chaque axe avance d'un pas toutes les 5ms jusqu'à recevoir un signal du capteur associé.

Si le capteur n'a rien signalé au bout d'un tour de l'axe pour l'azimut ou d'un demi-tour pour l'élévation, le programme s'arrête et signale une erreur.

Si le satellite n'est pas visible, c'est-à-dire lorsque l'élévation est négative, les angles sont recalculés toutes les 5 secondes et seul le moteur d'azimut suit cette consigne. L'élévation reste nulle jusqu'à ce que le satellite devienne visible.

Lors du suivi, les angles sont recalculés toutes les 100ms et les deux axes sont commandés jusqu'à ce que le satellite passe sous l'horizon.

Les fichiers relatifs à la motorisation sont :

- *capteur\_hall.cpp* et *capteur\_hall.h* pour la gestion des capteurs à effet Hall
- *motor.cpp* et *motor.h* pour l'utilisation de la librairie
- le dossier *motor* contenant la librairie

## 5. Affichage

Pour la gestion de l'écran, nous utilisons la librairie et la police de caractères de yanbe trouvée sur github disponible sous licence Apache, donc libre.

Cette bibliothèque est assez rudimentaire et ne permet que d'afficher du texte. Il a fallu ajouter le contrôle du reset de l'écran pour qu'il fonctionne correctement.

Nous avons modifié certains caractères non utilisées pour afficher des symboles plus plaisants :

- « ' » est devenu « ° » pour afficher les angles en degrés
- « \* » est devenu « ✓ » pour indiquer un fonctionnement réussi

L'affichage à tout moment est de la forme :

S	t	a	t	i	o	n	S	o	I	S	u	p	s	a	t
0	1	/	0	5	/	1	8		1	7	h	2	0		
W	i	F	i	:	✓		s	u	p	e	l	e	c	-	i
R	T	C	:	✓						T	L	E	:		✓
S	a	t	:	S	W	I	S	S	C	U	B	E			
V	i	s	i	b	l	e	:	x							
A	z	:	1	3	2	°		E	I	:	-	2	5	°	
A	t	t	e	n	t	e		V	i	s	i	b	.		

ou											
S	t	a	t	i	o	n	S	o	l	S	u
0	1	/	0	5	/	1	8	1	7	h	2
W	i	F	i	:	✓		s	u	p	e	l
R	T	C	:	✓			T	L	E	:	✓
S	a	t	:		S	W	I	S	S	C	U
V	i	s	i	b	l	e	:	?	B	E	
A	z	:	x				E	I	:	?	
!	P	o	s		I	n	i	t	A	z	E
							A	z	E	r	!

Figure 19. Exemples d'affichage

Texte statique

Date

Heure

Signale l'état d'un élément : x -> Erreur ;

? -> Inconnu ; ✓ -> Fonctionne ou Valide

Azimut actuel

Elévation actuelle

Informations sur l'état ou indique l'erreur

Nom du réseau WiFi (tronqué si trop long)

Nom du satellite suivi (tronqué si trop long)

Figure 20. Description des différents éléments de l'affichage



Figure 21. Photo de l'écran en fonctionnement

La dernière ligne permet d'afficher l'état actuel de la station. En cas d'erreur, elle indique également la nature de l'anomalie.

Voici les différentes informations affichables :

! WiFi Error !	Echec de connexion au WiFi
! Port Capt Er !	Echec d'authentification au portail captif
! RTC Echec !	Echec de synchronisation du RTC
! TLE Echec !	Echec de collecte des TLE
!Pos Init Az Er!	Echec de détection de la position initiale de l'axe d'azimut
!Pos Init El Er!	Echec de détection de la position initiale de l'axe d'élévation

Connex Auth	Connexion et authentification au réseau WiFi
Synchroni. RTC	Connexion par NTP et synchronisation du RTC interne
Collecte TLE	Connexion à celestrak.com et collecte des TLE
Pos. Init. Az.	Recherche de la position initiale de l'axe d'azimut
Pos. Init. El.	Recherche de la position initiale de l'axe d'élévation
Axes Initialises	Fin de la phase d'initialisation
Attente Visib.	Attend que le satellite soit visible (élévation < 0)
Suivi...	Suivi en cours
Suivi Termine	Fin du programme

Figure 22. Description des différents éléments de l'affichage

Les fichiers relatifs à l'affichage sont :

- *screen.cpp* et *screen.h* pour l'utilisation de la librairie
- le dossier *screen* contenant la librairie et la police

## 6. Configuration

Le fichier *conf.h* permet de modifier les points essentiels du fonctionnement du programme. Après modification, il est nécessaire de connecter l'ESP32 par USB. Ensuite il faut compiler le programme et le téléverser dans l'ESP32 depuis Eclipse. Le guide en annexe illustre cette manipulation.

Variable définie	Description	Valeur par défaut
USE_PORTAL	Si cette ligne est en commentaire, l'ESP32 se connectera à un réseau WPA2 personnel. Sinon, l'ESP32 se connecte à supelec-invite via le portail captif.	Pas commenté
WIFI_SSID	Nom du réseau WPA2 personnel	
WIFI_PASS	Mot de passe du réseau WPA2 personnel	
WIFI_USERNAME	Identifiant du portail captif de supelec-invite	sessio8
WIFI_PASSWORD	Mot de passe du portail captif de supelec-invite	t4a3s2.1
SERVEUR_NTP	Adresse du serveur NTP pour synchronisation du RTC	pool.ntp.org
SAT_NAME	Nom du satellite à suivre tel qu'il est écrit sur la page de collecte des TLE	
SITE_LATITUDE	Latitude du site d'observation de la station sol	48.125 (pour Supélec Rennes)
SITE_LONGITUDE	Longitude du site d'observation de la station sol	-1.623 (pour Supélec Rennes)
SITE_ALTITUDE	Altitude du site d'observation de la station sol par rapport au niveau de la mer (en m)	65 (pour Supélec Rennes)
WEB_SERVER	Nom de domaine du site où collecter les TLE	celesttrak.com
WEB_PORT	Port pour la collecte des TLE	80
WEB_URL	URL du fichier texte à télécharger contenant les TLE	/NORAD/elements/cubesat.txt

Figure 23. *Eléments configurables depuis le fichier de configuration*

## 7. Général

Nous allons à présent voir comment les différentes parties logicielles décrites précédemment interagissent ensemble.

Le déroulement du programme peut être présenté sous forme d'organigramme :

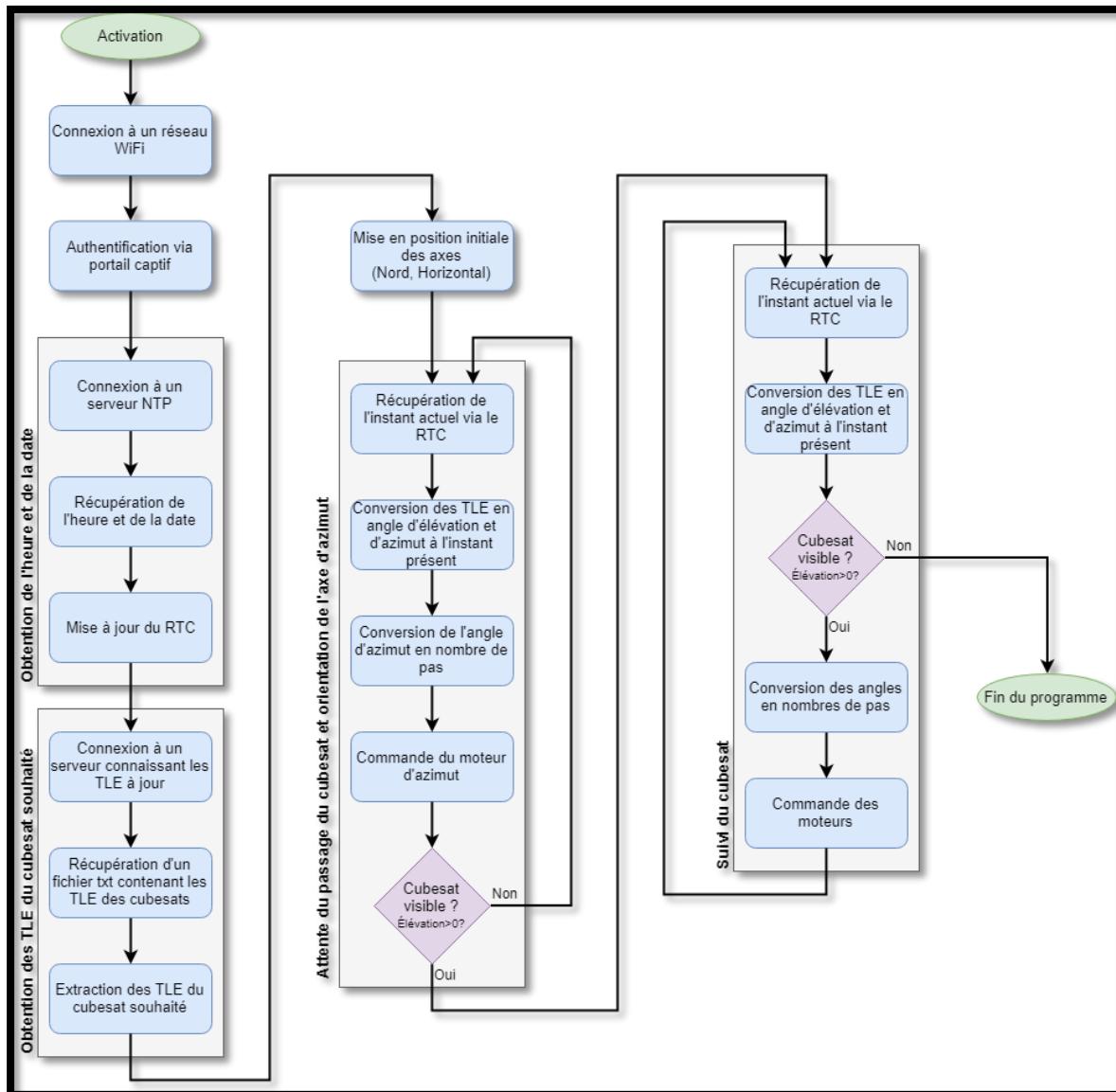


Figure 24. Organigramme illustrant les actions de l'ESP32

Il est toujours possible de connecter l'ESP32 à un ordinateur afin d'obtenir un débogage plus avancé. L'ESP32 communique par USB en liaison série à un débit de 115,2kbauds.

Les fichiers relatifs à la gestion globale de la station sont `main.cpp` et `main.h`.

## E. Réalisation

Les différents modules ont été assemblés sur une plaquette d'essai pour la programmation et l'expérimentation.

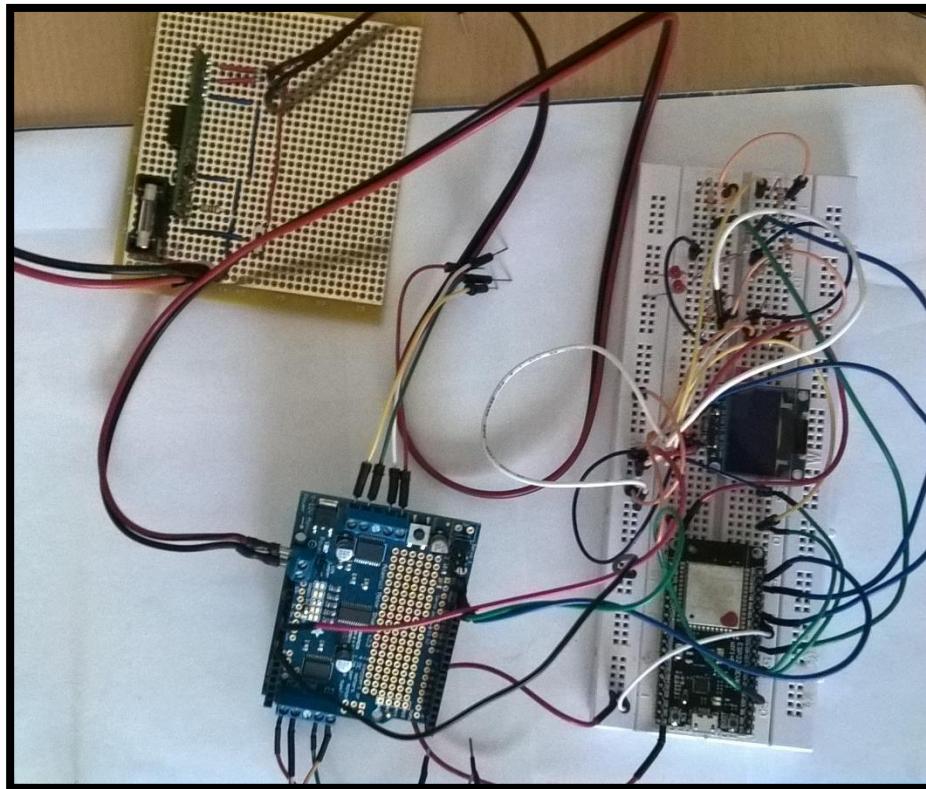


Figure 25. Prototype du système de suivi

Les différents pins de l'ESP32 ont été testés en lecture et en écriture pour déterminer lesquels pouvaient être utilisés étant donné la documentation peu compréhensible sur l'utilisabilité des pins GPIO. Certains sont en effet réservés à des fonctions spécifiques ou configurés en lecture seule.

Chaque composant a été testé individuellement d'abord pour vérifier et corriger les librairies utilisées. Pour chaque module, différentes bibliothèques ont été essayées car les fichiers de l>IDF sont quotidiennement mis à jour, ce qui casse parfois les compatibilités avec les librairies tierces.

L'écran et le driver de moteurs n'ont pas fonctionné ensemble sur le bus I2C pour des raisons inconnues bien que leurs adresses soient différentes. Il a donc fallu utiliser deux bus logiciels distincts pour les faire cohabiter.

Lorsque le programme fut terminé et fonctionnel, nous avons réalisé les schémas électriques des circuits à l'aide de KiCad :

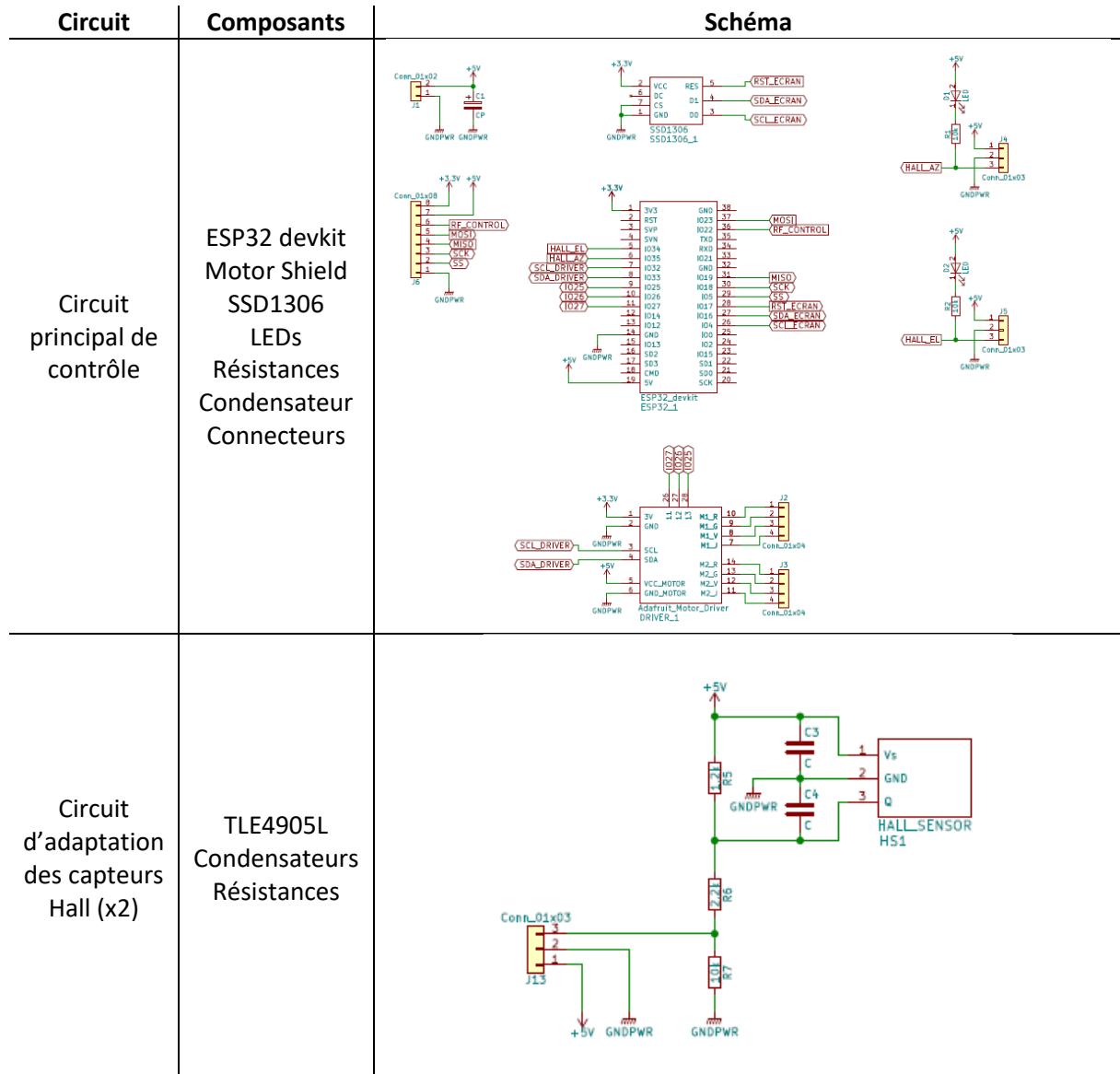


Figure 26. Circuits du système de suivi

La conversion de 5V à 3,3V du signal logique des capteurs à effet Hall a dû être revue pour permettre l'ajout des LEDs de contrôle.

Circuit	Dénomination	Description	Valeur
Adaptation Capteur Hall	R5	Résistance de pull-up	1,2kΩ
	R6	Pont diviseur de tension	2,2kΩ
	R7	tension	10kΩ
	C3, C4	Lissage	4,7nF
Principal de contrôle	R1, R2	Limitation du courant dans les LEDs	10kΩ
	C1	Lissage	470μF

Figure 27. Description et valeurs des composants

Nous avons ensuite réalisé le routage des différents circuits. Certaines connexions ont été redirigées afin d'avoir un circuit sur une seule face et un encombrement restreint pour pouvoir être lié par la suite à la structure.

Nous avons ensuite fait réaliser les PCB par le CRIS :

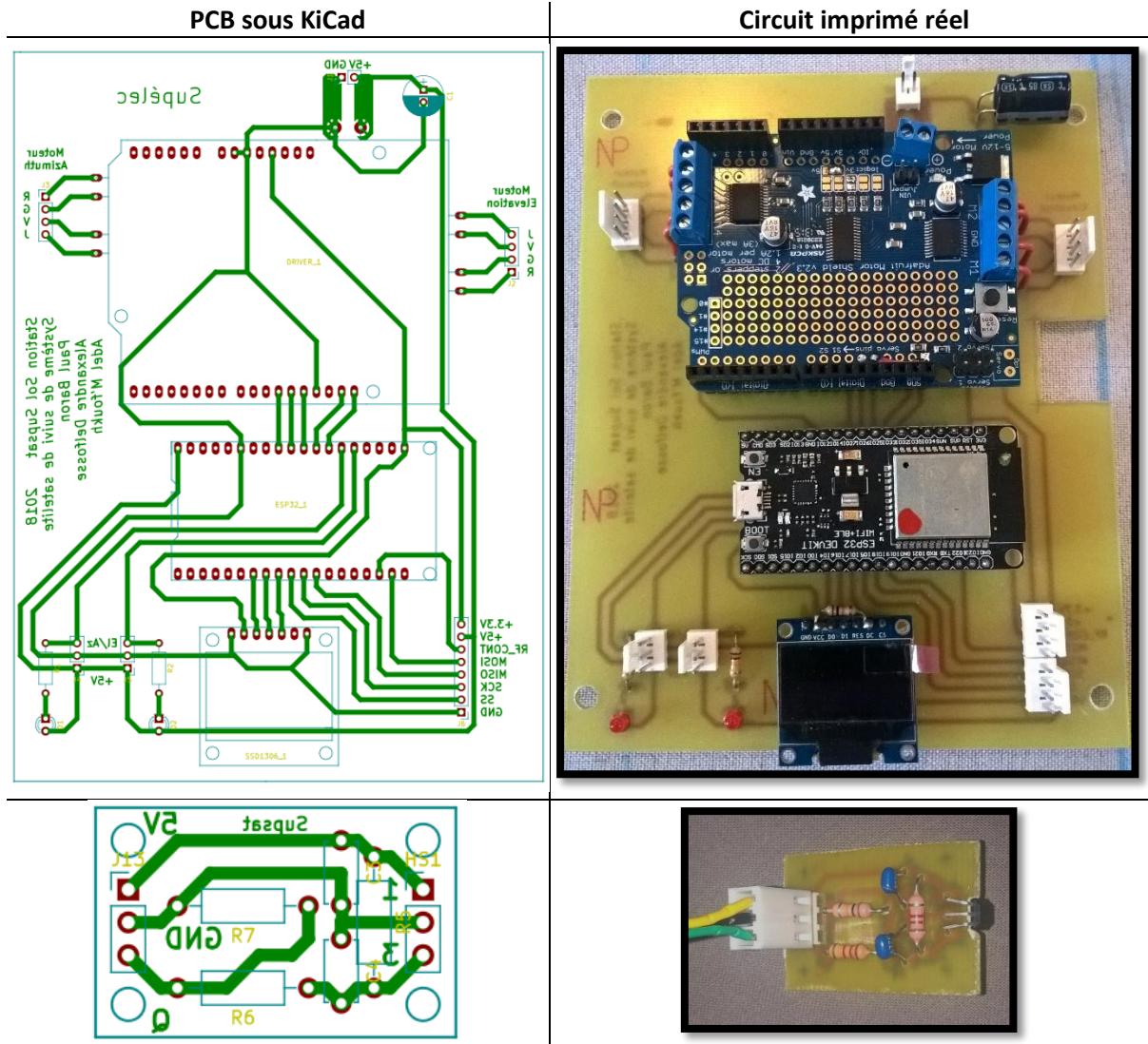


Figure 28. PCB et circuits imprimés

L'ESP32 et le Shield Adafruit sont connectés par l'intermédiaire de supports soudés afin de pouvoir les enlever au besoin.

Une encoche dans le PCB permettra le passage des câbles par la suite. Des trous à quatre coins permettent le passage de vis M4 pour une fixation ultérieure.

Les connections 11, 12 et 13 du Shield Adafruit sont connectées respectivement aux pins 27, 26 et 25 de l'ESP32 et ne sont pas utilisés. Ils sont disponibles pour des améliorations.

Cependant, des problèmes sont apparus durant l'utilisation du pin 32 pour le SCL du Shield Adafruit, nous utilisons donc le pin libre 25 en remplacement.

Voici les différentes connexions de l'ESP32 :

Pin	Fonction	Lié à
IO 34	Digital input	Capteur Hall d'élévation
IO 35	Digital input	Capteur Hall d'azimut
IO 22	Digital output	Circuit RF (contrôle)
IO 16	SDA 1	Ecran (SDA)
IO 4	SCL 1	Ecran (SCL)
IO 17	Digital output	Ecran (reset)
IO 33	SDA 0	Driver (SDA)
IO 32	SCL 0 (non fonctionnel)	Driver (SCL)
IO 25	SCL 0	Driver (13 & SCL)
IO 23	MOSI	Circuit RF (MOSI)
IO 18	MISO	Circuit RF (MISO)
IO 19	SCK	Circuit RF (SCK)
IO 5	SS	Circuit RF (SS)
IO 26	Digital input/output (inutilisé)	Driver (12)
IO 25	Digital input/output (inutilisé)	Driver (11)

Figure 29. Attribution des pins de l'ESP32

## F. Essai

Nous avons testé le logiciel de contrôle du suivi lors du passage d'un satellite dans notre zone de visibilité. Le site n2yo.com permettant de suivre les satellites passant au-dessus de nous, il nous sert de référence lors de ce test.

Nous avons choisi le cubesat Tigrisat pour faire cet essai. Nous disposons des informations suivantes issues du site :

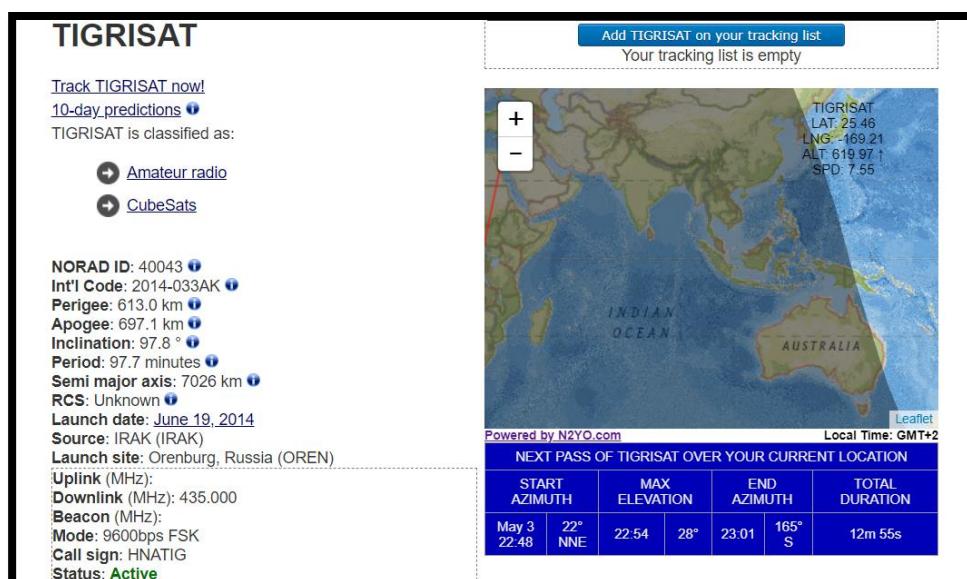


Figure 30. Données de référence pour l'essai du suivi

Nous allons essentiellement vérifier les points clés du suivi, c'est-à-dire l'azimut à l'heure d'apparition, l'élévation maximale à l'heure donnée et l'azimut finale à l'heure de disparition. Pour cela, le programme est légèrement modifié pour donner des informations toutes les 2 secondes.

Ainsi, lors de ce suivi, nous obtenons :

Informations de position du programme	Interprétation
Heure: Thu May 3 22:48:09 2018 TIGRISAT : Az:21.891096' / El:-0.132035' / Distance:3046.241494km Lat:71.960245' / Long:33.829307' / Altitude:683.221034km	Le début du suivi a débuté à 22h48:10 avec un azimut de 22,0°
Heure: Thu May 3 22:48:12 2018 TIGRISAT : Az:22.058264' / El:0.038952' / Distance:3027.359386km Lat:71.796014' / Long:33.561826' / Altitude:683.309658km	Ceci est en parfait accord avec les prédictions
Heure: Thu May 3 22:54:42 2018 TIGRISAT : Az:93.332656' / El:28.487812' / Distance:1264.678348km Lat:49.164173' / Long:16.498903' / Altitude:690.997456km	Le maximum d'élévation est atteint à 22h54:45 avec une élévation de 28,5°
Heure: Thu May 3 22:54:45 2018 TIGRISAT : Az:94.113463' / El:28.489484' / Distance:1264.663920km Lat:49.045181' / Long:16.452191' / Altitude:691.018879km	Ceci s'accorde également avec les prédictions
Heure: Thu May 3 22:54:47 2018 TIGRISAT : Az:95.284426' / El:28.481604' / Distance:1264.940200km Lat:48.866665' / Long:16.382463' / Altitude:691.050711km	Le suivi s'est terminé à 23h01:15 avec un azimut de 165,5°
Heure: Thu May 3 23:01:15 2018 TIGRISAT : Az:165.393346' / El:0.125413' / Distance:3035.356509km Lat:25.603479' / Long:9.524067' / Altitude:692.660275km	Ceci est également en accord

Figure 31. Mesures effectuées durant le suivi

Cet essai valide donc notre implémentation du logiciel de suivi.

Cependant, d'autres essais ont montré l'importance des données du site d'observation (latitude, longitude et altitude). Nous recommandons au minimum 4 chiffres après la virgule (équivalent à une précision de moins de 10m) pour la longitude et la latitude et une précision de l'ordre du mètre pour l'altitude afin de ne pas biaiser le suivi. r<sup>3</sup>r

## G. Résumés

Nous pouvons récapituler ici l'ensemble des imprécisions apparaissant dans notre système.

Nous avons déjà déterminé l'imprécision des jeux mécaniques, l'imprécision du système d'entraînement, l'imprécision issue des TLE et l'imprécision due à l'heure.

Source de l'imprécision	Ordre de grandeur
Moteur pas-à-pas	0,03°
Jeux mécaniques	1,5°
TLE	0,008° / jour d'ancienneté des TLE
Date et heure	0,06° / seconde d'écart

Figure 32. Résumé des imprécisions du système de suivi

Ainsi, on constate que l'erreur principale vient des jeux mécaniques. Cependant, les TLE doivent être récentes et le RTC précis pour ne pas augmenter l'erreur globale.

L'antenne devra donc posséder un angle d'ouverture prenant en compte cette erreur.

Nous pouvons également récapituler les licences sous lesquelles les bibliothèques utilisées dans ce projet sont distribués et en préciser les termes :

Bibliothèque	Auteur	Licence	Conditions d'utilisation
SGP4	David Vallado et Hopperpop	MIT	Droit illimité de l'utiliser, le copier, le modifier, le fusionner, le publier, le distribuer, le vendre et de changer sa licence. La seule obligation est de mettre le nom des auteurs avec la notice de copyright.
Motor Shield	ShellAddicted	BSD	Droit illimité de l'utiliser, le copier, le modifier, le fusionner, le publier, le distribuer, le vendre et de changer sa licence.
SSD1306	yanbe	Apache 2.0	Modification autorisée et distribution du code sous toute forme. Obligation du maintien du copyright lors de toute modification.
Autres	IDF d'Espressif	Apache 2.0	

Figure 33. Résumé des librairies utilisées

Notre logiciel est donc entièrement libre et utilisable légalement sous toutes les formes tant que les licences originales et les noms des auteurs sont présents.

## H. Améliorations

Il reste à réaliser la communication avec le circuit RF et à implémenter le traitement par l'ESP. Au final, il doit être en mesure de collecter les données reçues et de les rendre accessibles via sa connexion WiFi. De la même façon, il devra être capable de commander le circuit RF pour émettre des commandes de vol.

Cependant, le circuit RF n'étant pas terminé avant la fin de ce projet, l'implémentation de ces fonctionnalités n'a pas été faite en complément du logiciel de contrôle du suivi.

Le câble de connexion vers le circuit RF est réalisé du côté du circuit de contrôle. Il sera peut-être justifier de le remplacer néanmoins par des fils de diamètre plus important. En effet, la consommation du circuit RF risque d'être trop importante pour le faible diamètre de ce câble. De plus, le communication via SPI risque de poser problème sur une distance supérieure à deux mètres. Il faudra veiller à faire des essais avant la réalisation finale de la station.

Couleur	Fonction
Noir	GND
Blanc	CONTROL_RF
Jaune	+5V
Marron	GND
Vert	MOSI
Violet	SCK
Bleu	SS
Orange	MISO
Rouge	+3,3V

Figure 34. Description de la connectique vers le circuit RF

De plus, nous avons proposé à des premières années de travailler en projet court sur la possibilité d'utiliser un IMU (Inertial Motion Unit) contenant un magnétomètre, un accéléromètre et un gyroscope afin de rendre la station nomade. En effet, ce système permettrait de placer les deux axes en position initiale qu'elle que soit l'orientation de la station. Il faudra prendre en compte leur conclusion sur la faisabilité et la précision d'un tel capteur afin de l'intégrer en remplacement des capteurs à effet Hall ou non.

### III. Boîtier

Nous allons à présent confectionner une protection pour la structure et les circuits de contrôle de la station. Nous avons décidé de les imprimer en 3D à l'école pour obtenir un boîtier modulable et sur mesure. L'imprimante impose des contraintes sur les dimensions maximales des pièces à imprimer.

Chaque partie de ce boîtier possède des spécificités pour le passage des câbles, l'ajustement à la structure et l'adaptation aux circuits imprimés.

La station étant destinée à reposer à l'extérieur, nous avons choisi d'utiliser de l'ABS au lieu du PLA pour une meilleure tenue dans le temps.

Cependant, les plus grandes pièces ont posé problème durant l'impression et nous avons dû nous rabattre sur du PLA renforcé.

Nous avons choisi des matériaux bleus aux couleurs de l'école.

Nous avons utilisé le logiciel Fusion 360 pour modéliser les pièces.

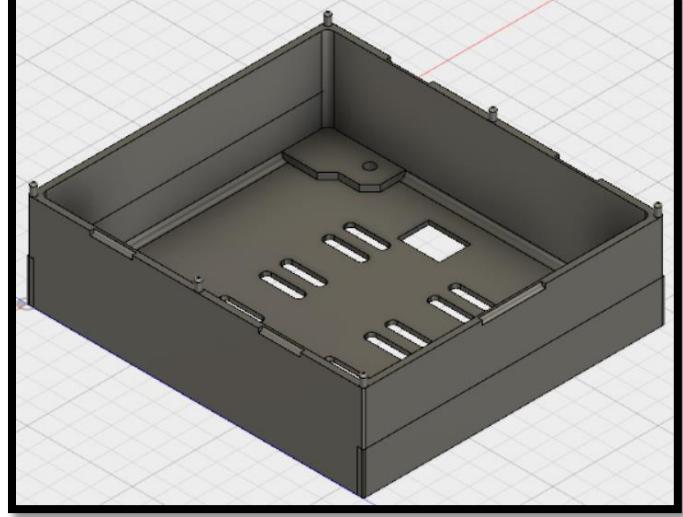
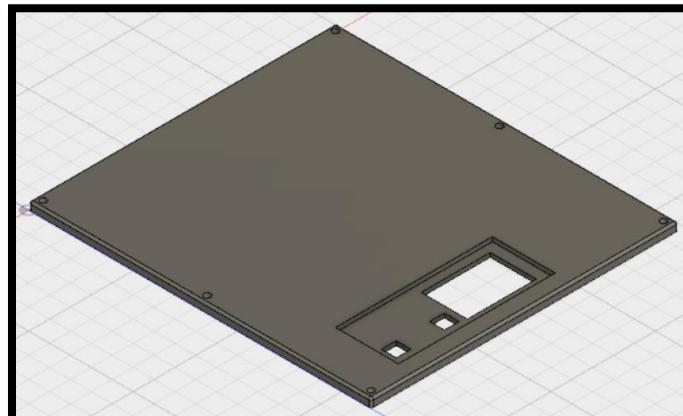
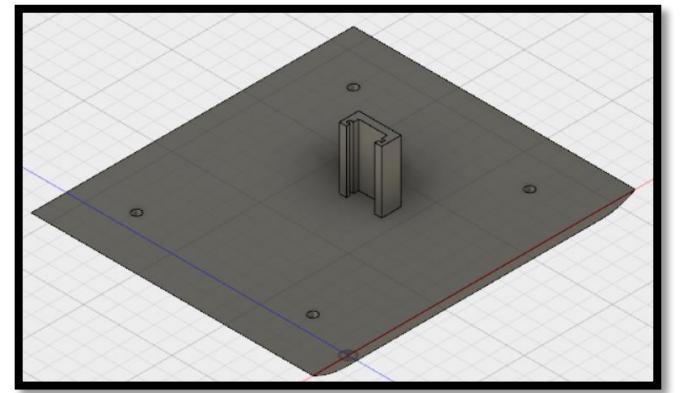


Figure 35. Imprimante 3D disponible à l'école

## A. Partie basse

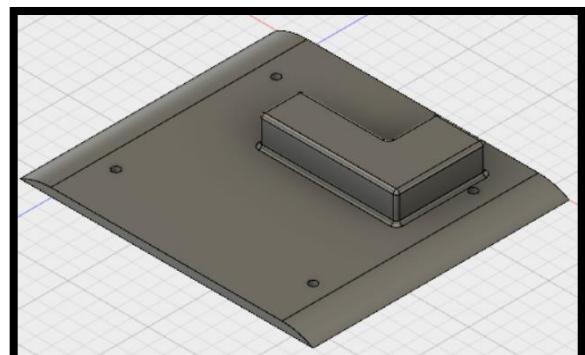
Nous commençons par la modélisation du boîtier protégeant la structure inférieure, c'est-à-dire autour de la motorisation de l'axe d'azimut.

Cette partie est la plus complexe. En effet, elle doit abriter le circuit de contrôle principal et la batterie avec son circuit de conversion en plus du capteur à effet Hall et la gestion du câblage.

Caractéristiques	Modélisation
<p><i>ci_principal_base</i></p> <ul style="list-style-type: none"><li>▪ Abrite le circuit de contrôle principal</li><li>▪ Fixations à la structure et au circuit</li><li>▪ Ouverture pour le passage des câbles</li><li>▪ Ouverture pour accrocher les câbles internes à la structure</li></ul>	
<p><i>ci_principal_capot</i></p> <ul style="list-style-type: none"><li>▪ Couvercle du circuit de contrôle</li><li>▪ Fixation au boîtier du circuit</li><li>▪ Ouverture pour un accès visuel à l'écran et aux LEDs (à protéger par un matériau transparent)</li><li>▪ Retirable facilement</li></ul>	
<p><i>hall_sensor</i></p> <ul style="list-style-type: none"><li>▪ Abrite le circuit d'adaptation du capteur à effet Hall d'azimut</li><li>▪ Fixations à la structure</li><li>▪ Glissière pour fixation du circuit</li><li>▪ Orientation pour optimiser le passage des câbles</li></ul>	

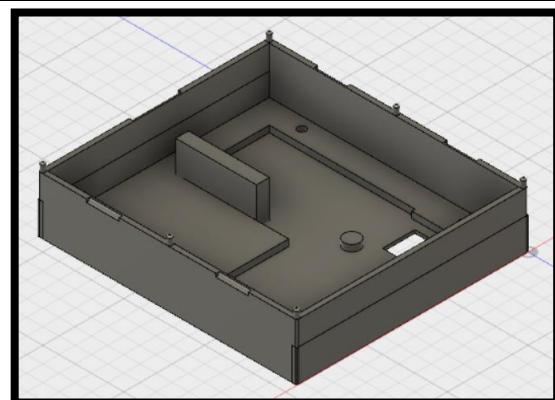
*motor\_bas*

- Permet le passage des câbles du moteur d'azimut
- Fixations à la structure
- Attaches dans le renforcement pour le maintien des câbles



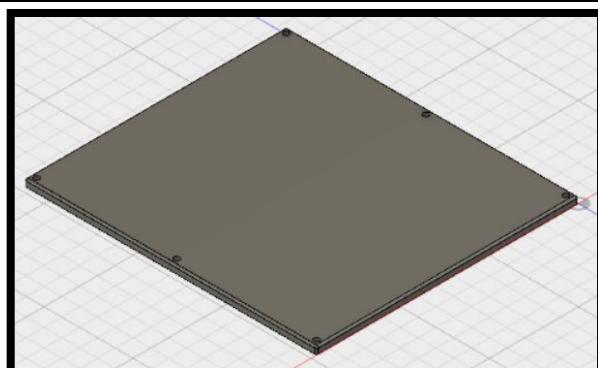
*alim\_base*

- Abrite le circuit de conversion d'énergie
- Abrite la batterie
- Fixations à la structure et au circuit
- Ouverture pour le passage des câbles
- Maintien de la batterie pouvant être facilement enlevée



*alim\_capot*

- Couvercle du circuit de conversion
- Fixation au boîtier du circuit
- Retirable facilement



*axe\_dessous\_petit*  
*axe\_dessous\_grand*

- S'agencent autour de l'axe d'azimut
- Fixations à la structure

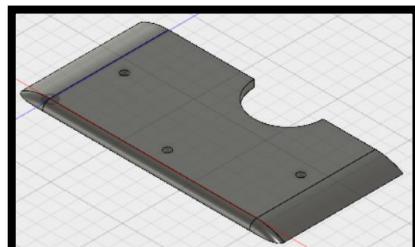
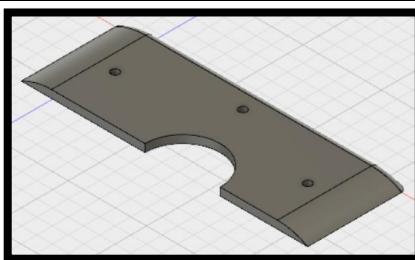


Figure 36. Résumé des différentes parties de la partie basse

En assemblant ces différentes parties, nous obtenons :

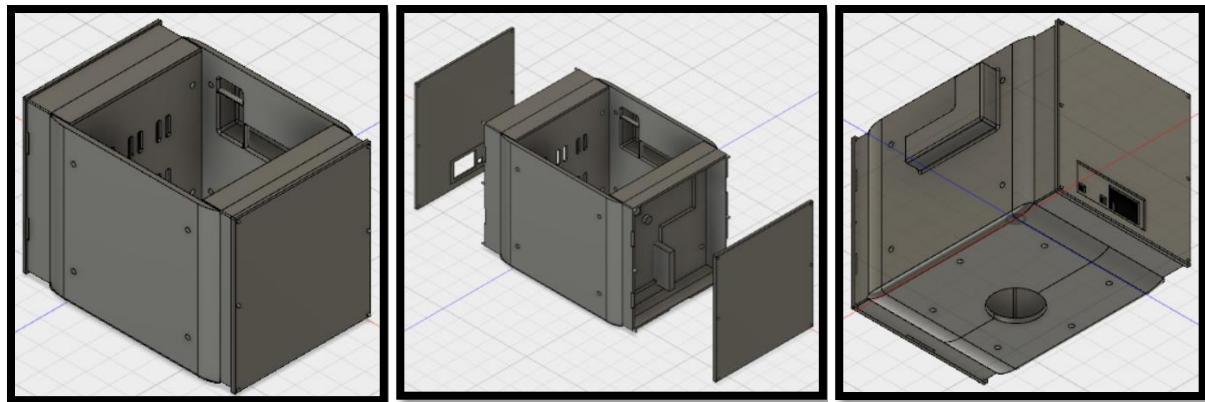


Figure 37. Assemblage de la partie basse du boitier (dessus, éclatée, dessous)

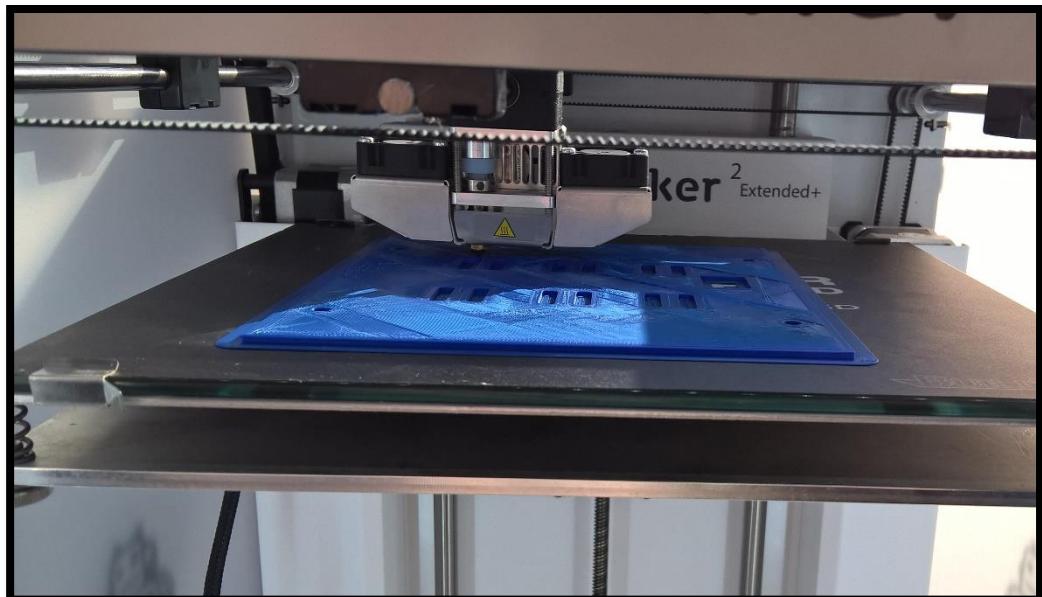


Figure 38. Impression d'une pièce

## B. Partie haute

La partie supérieure de la structure est plus simple et la plupart de ses pièces se basent sur celles de la partie basse.

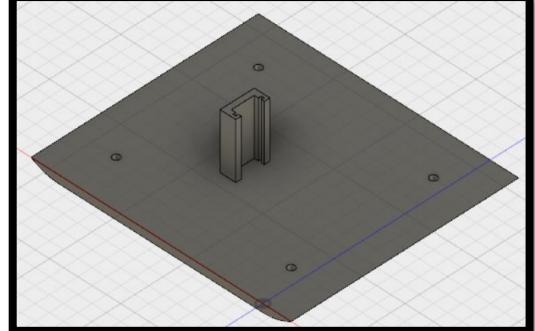
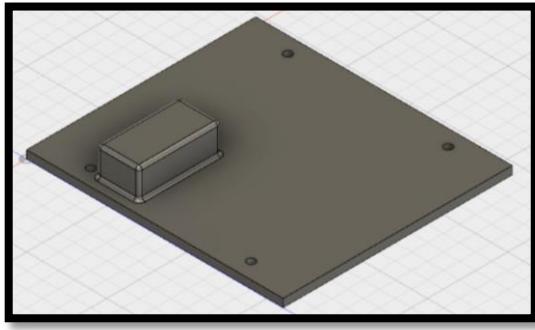
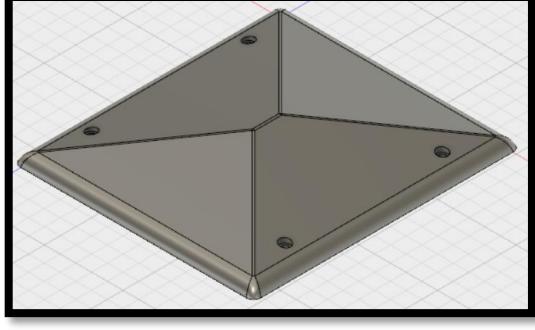
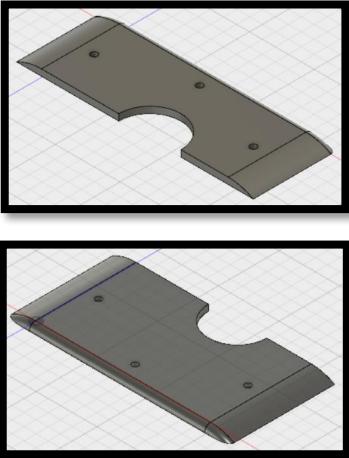
Caractéristiques	Modélisation
<p><i>hall_sensor</i></p> <ul style="list-style-type: none"> <li>▪ Abrite le circuit d'adaptation du capteur à effet Hall d'élévation</li> <li>▪ Fixations à la structure</li> <li>▪ Glissière pour fixation du circuit</li> <li>▪ Orientation pour optimiser le passage des câbles</li> </ul>	
<p><b>Miroir de la pièce de la partie basse</b></p> <p><i>motor_haut</i></p> <ul style="list-style-type: none"> <li>▪ Permet le passage des câbles du moteur d'élévation</li> <li>▪ Fixations à la structure</li> <li>▪ Attaches dans le renforcement pour le maintien des câbles</li> </ul>	
<p><i>toit</i></p> <ul style="list-style-type: none"> <li>▪ Incliné pour permettre l'écoulement en cas de pluie</li> <li>▪ Fixation à la structure</li> </ul>	
<p><i>axe_gauche_petit</i>  <i>axe_gauche_grand</i>  <i>axe_droit_petit</i>  <i>axe_droit_grand</i></p> <ul style="list-style-type: none"> <li>▪ S'agencent autour de l'axe d'azimut</li> <li>▪ Fixation à la structure</li> </ul> <p><b>Identiques aux pièces de la partie basse pour les parts de gauche et miroirs pour les parts de droite</b></p>	

Figure 39. Résumé des différentes parts de la partie basse

Nous obtenons l'assemblage pour la partie haute suivant :

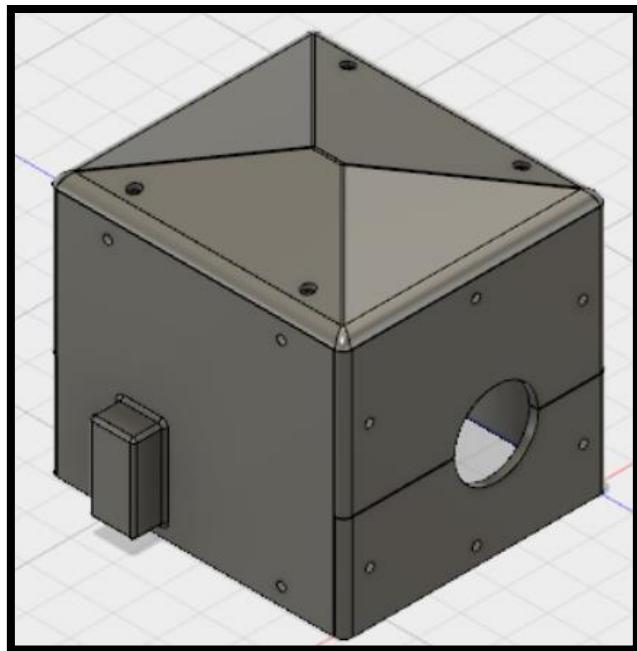


Figure 40. Assemblage de la partie haute du boitier

Globalement, notre modélisation donne pour l'ensemble de la structure :

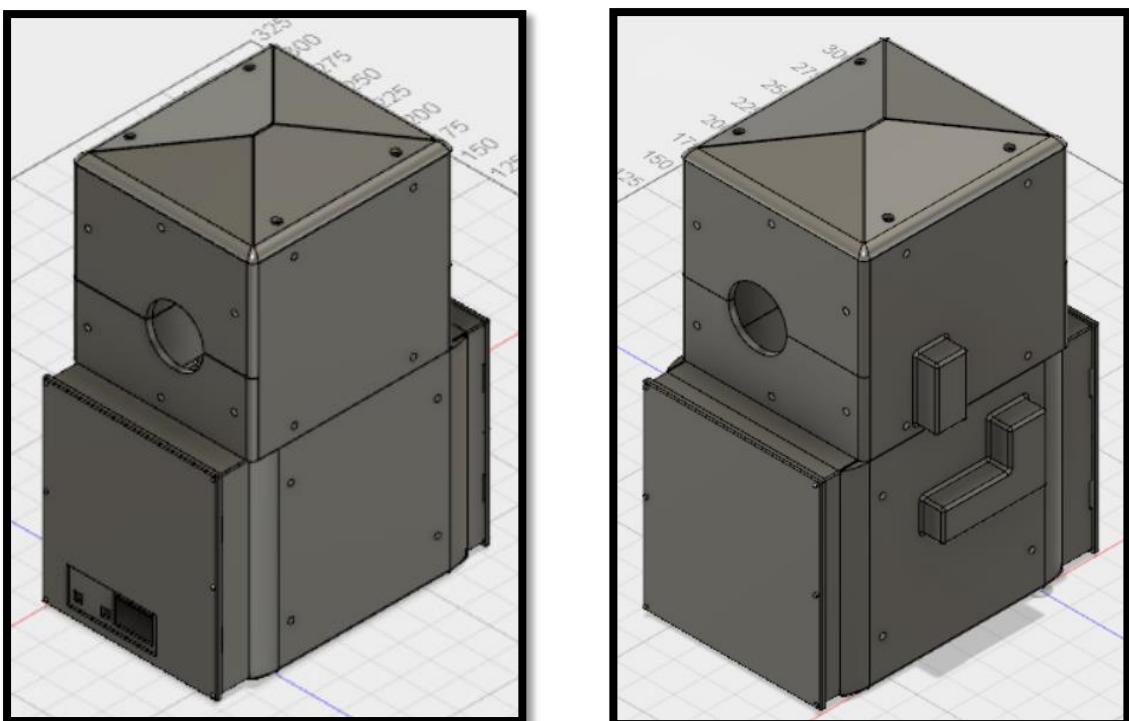


Figure 41. Assemblage global du boitier

## C. Réalisation

Nous voulions opter pour du plastique de type ABS pour l'impression du boitier. Cependant, lors des différents essais, les pièces imprimées avaient tendance à gondoler. N'ayant pas pu résoudre ce problème dans les temps, nous avons opté pour du PLA renforcé, d'avantage résistant que du PLA usuel. Ce plastique n'a ensuite posé aucun problème à l'impression.

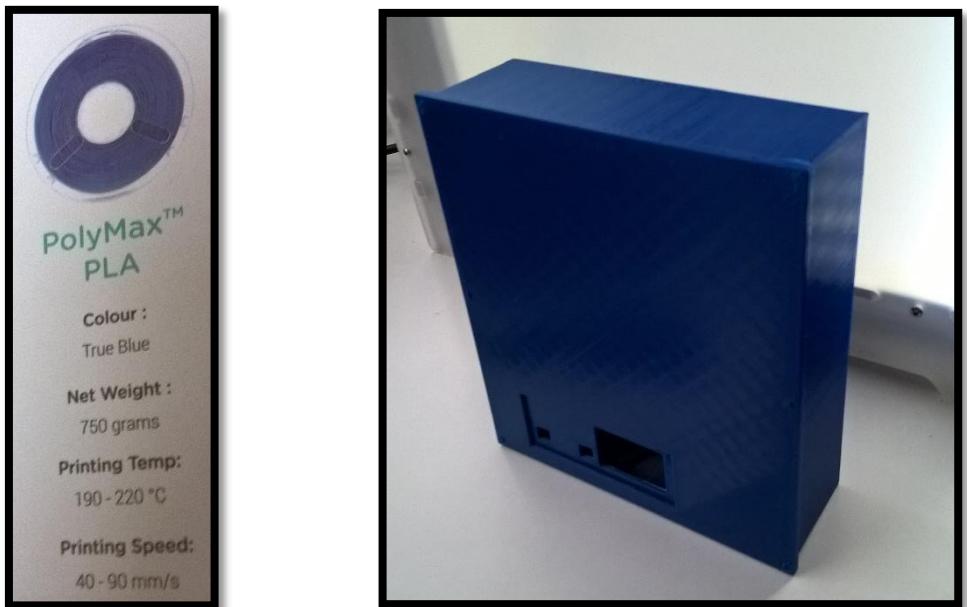


Figure 42. Boîte du PLA renforcé et boîtier du circuit principal imprimé avec couvercle

Afin d'avoir un rendu plus esthétique, nous avons essayé de poncer une pièce (sous le boitier) et d'en vernir une autre (boitier du circuit principal). Dans les deux cas, le résultat était mitigé et nous avons décidé de laisser les autres pièces sans traitement.

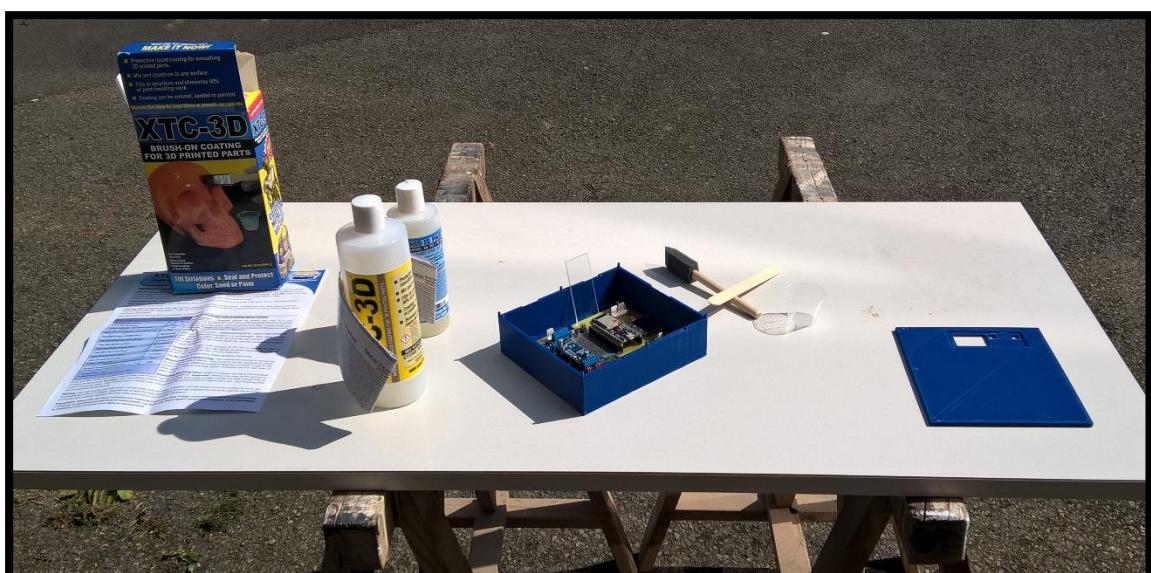


Figure 43. Vernissage du boîtier du circuit principal

Nous avons ensuite fixé les circuits aux boîtiers. La vitre du boîtier principal est en plexiglas et collée à son emplacement afin de voir l'écran et les deux LEDs de contrôle sans ouvrir le boîtier.

Pour le capteur à effet Hall, nous avons collé l'aimant sur l'axe d'azimut et collé le circuit du capteur sur la pièce imprimée correspondante afin que la distance entre le capteur et l'aimant soit adaptée à une détection sûre et précise.

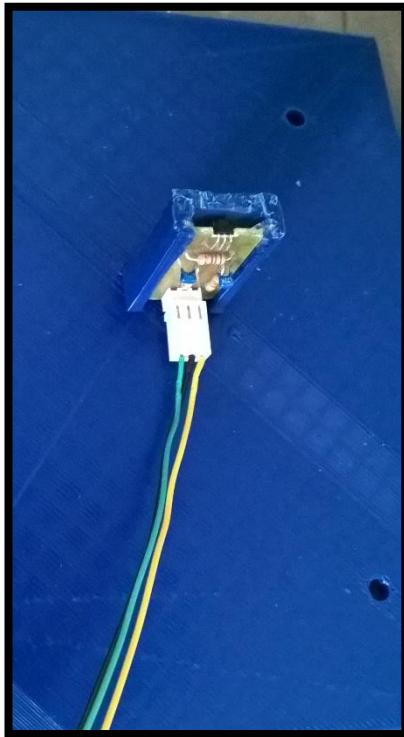


Figure 44. Circuit du capteur à effet Hall sur sa pièce imprimée

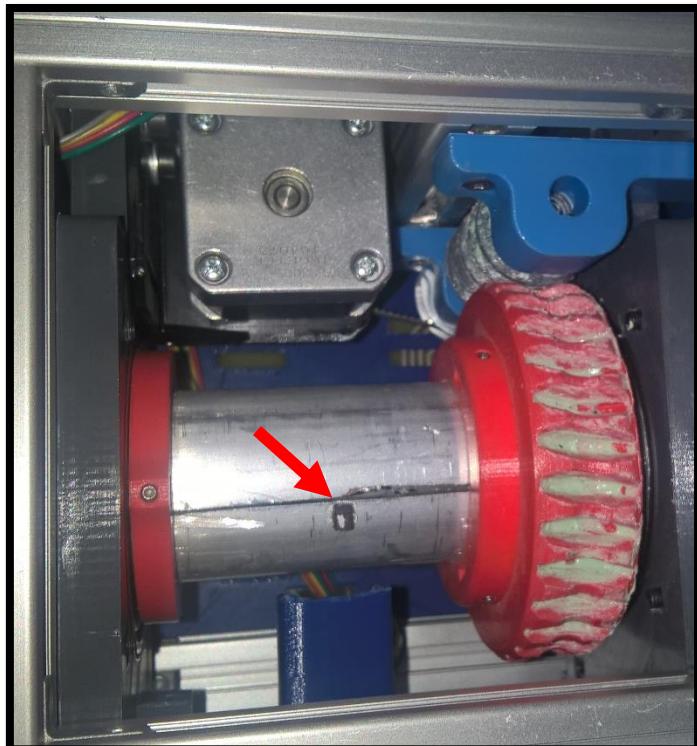


Figure 45. Aimant collé sur l'axe d'azimut

La modélisation de la partie haute du boîtier est modélisée à l'exception du trou permettant le passage du câble vers le circuit RF. Il reste à l'imprimer et à le fixer. Le capteur à effet Hall de l'axe d'élévation devra également être collé avec son aimant.

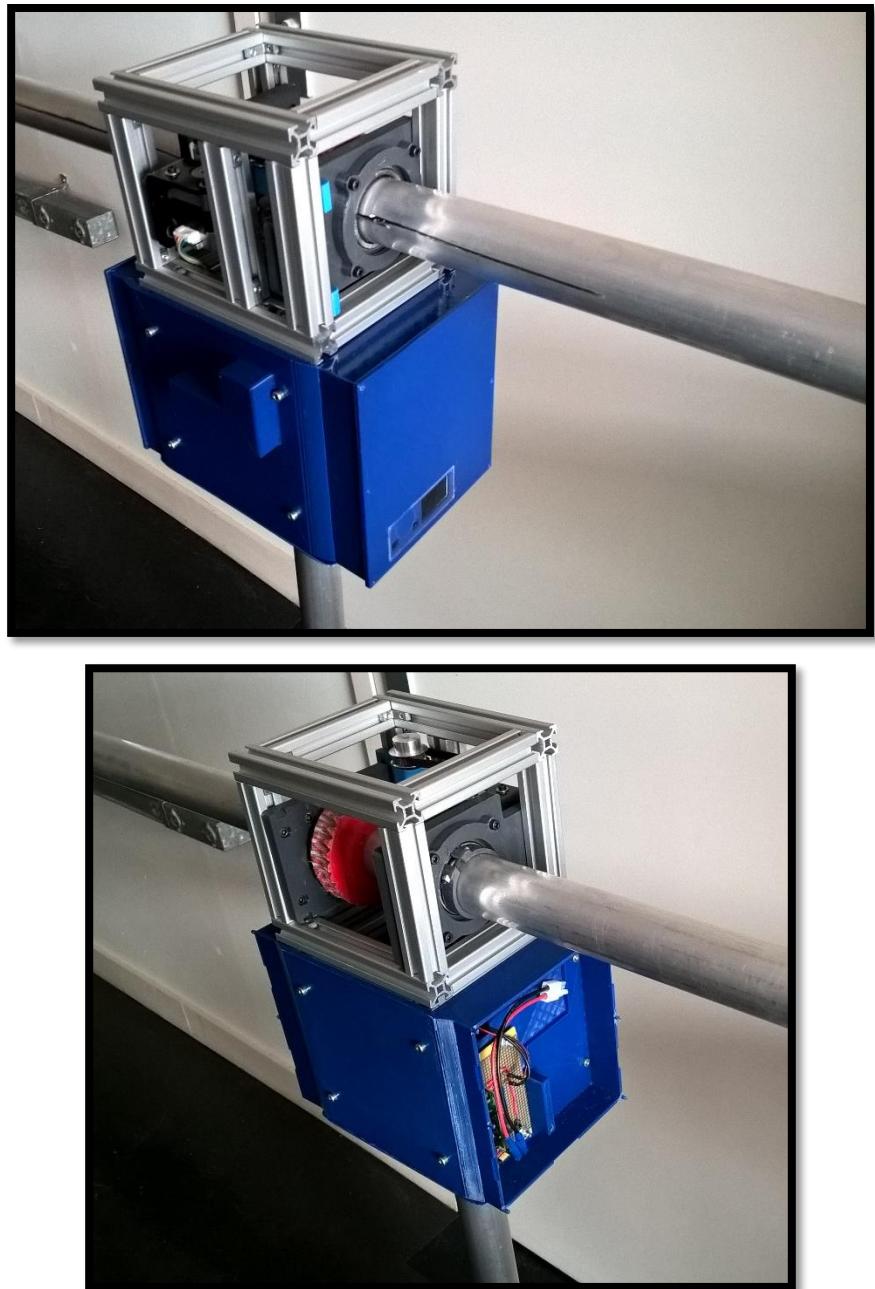


Figure 46. Boîtier bas monté

## **IV. Réception et émission**

Cette partie aborde l'aspect radiofréquence de la communication avec le future cubesat de Supsat. Nous avons travaillé en collaboration avec le groupe chargé du projet long sur la carte de communication du cubesat afin de s'accorder sur le dimensionnement du lien entre le satellite et la station sol.

Nous aborderons le dimensionnement et le choix des composants, et en particulier l'antenne.



Figure 47. Centre de communication de Pleumeur-Bodou

## A. Bilans de liaison

### 1. Lien montant

Afin d'établir le bilan de liaison, il est important de rappeler les caractéristiques principales de la mission.

Le CubeSat est conçu pour être en orbite basse (LEO : Low Earth Orbit) circulaire autour de la terre, à une altitude de 400km. On peut alors déterminer sa vitesse orbitale moyenne (en négligeant la masse du satellite devant celle de la terre, ce qui n'est pas ridicule...):

$$v = \sqrt{\frac{M \cdot G}{r + h}}$$

Avec  $G = 6,67 \cdot 10^{-11} \text{ m}^3 \cdot \text{kg}^{-1} \cdot \text{s}^{-2}$  la constante de la gravitation ;  $M = 5,972 \cdot 10^{24} \text{ kg}$  la masse de la terre ;  $r = 6371 \cdot 10^3 \text{ m}$  le rayon de la terre et  $h = 400 \cdot 10^3 \text{ m}$  la hauteur de l'orbite.

On trouve  $v = 7,7 \cdot 10^3 \text{ m.s}^{-1} = 7,7 \text{ km.s}^{-1}$

Cette vitesse est assez élevée pour provoquer un décalage de fréquence par effet Doppler d'environ 10KHz lors du passage du satellite au-dessus de la station sol, décalage qu'il faudra prendre en compte et corriger avec soin.

Le satellite étant en orbite circulaire, il est vu, par la station sol, à une distance qui dépend de l'angle d'élévation (en approximant que sa trajectoire passe quasi parfaitement « au-dessus de notre tête », ce qui n'est pas forcément le cas en réalité et rajoute des pertes. Mais afin de simplifier les calculs, on peut faire cette approximation).

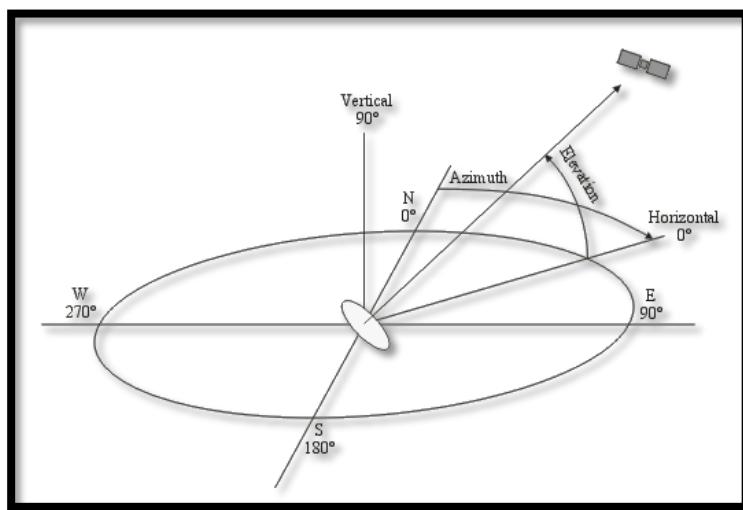


Figure 48. Angles d'azimuth et d'élévation

On peut exprimer l'un en fonction de l'autre. Cela permettra par la suite d'exprimer les pertes en espace libre (dues à la distance parcourue par le signal) en fonction de l'angle d'élévation de l'antenne, et déterminer une plage utile d'élévation.

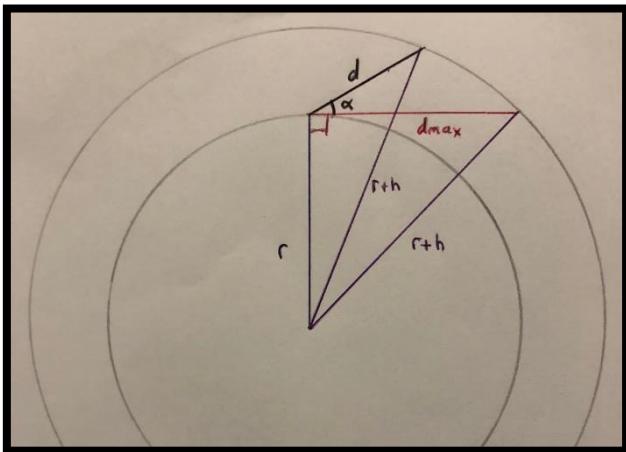


Figure 49. Détermination de la distance entre la station sol et le cubesat

Soit  $r = 6\ 371.\ 10^3 m$  le rayon de la terre et  $h = 400.\ 10^3 m$  la hauteur de l'orbite,  $\alpha$  l'angle d'élévation et  $d$  la distance séparant le satellite de la station sol.

On peut déterminer la distance maximale  $d_{max}$  entre le satellite et la station, qui correspond à un angle d'élévation  $\alpha = 0^\circ$ , à l'aide du théorème de Pythagore :

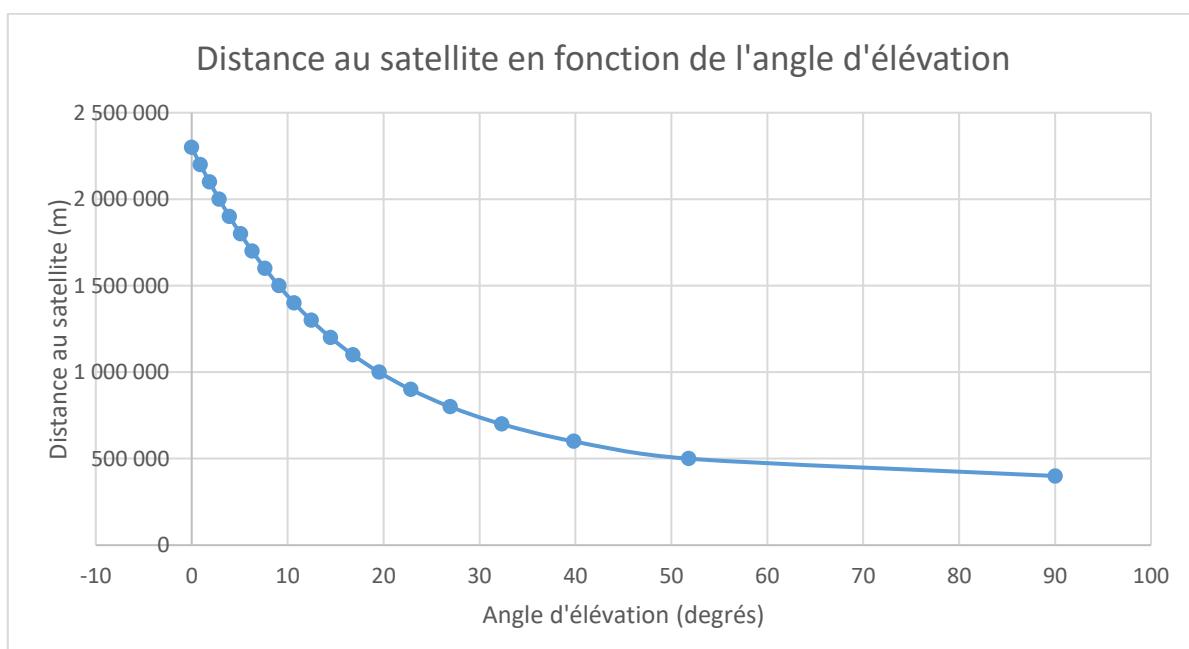
$$d_{max}^2 + r^2 = (r + h)^2$$

$$d_{max} = \sqrt{(r + h)^2 - r^2} = 2298 km$$

Par construction  $d_{min} = h = 400 km$  pour une élévation  $\alpha = 90^\circ$ . Pour toutes les autres positions intermédiaires, on doit appliquer le théorème d'Al-Kashi :

$$\alpha = \arccos \left( \frac{r^2 + d^2 - (r + h)^2}{2 \cdot r \cdot d} \right) - \frac{\pi}{2}$$

Ainsi, on est capable de déterminer les valeurs d'élévation correspondantes pour des distances satellite – station sol variant de 400 à 2300km :

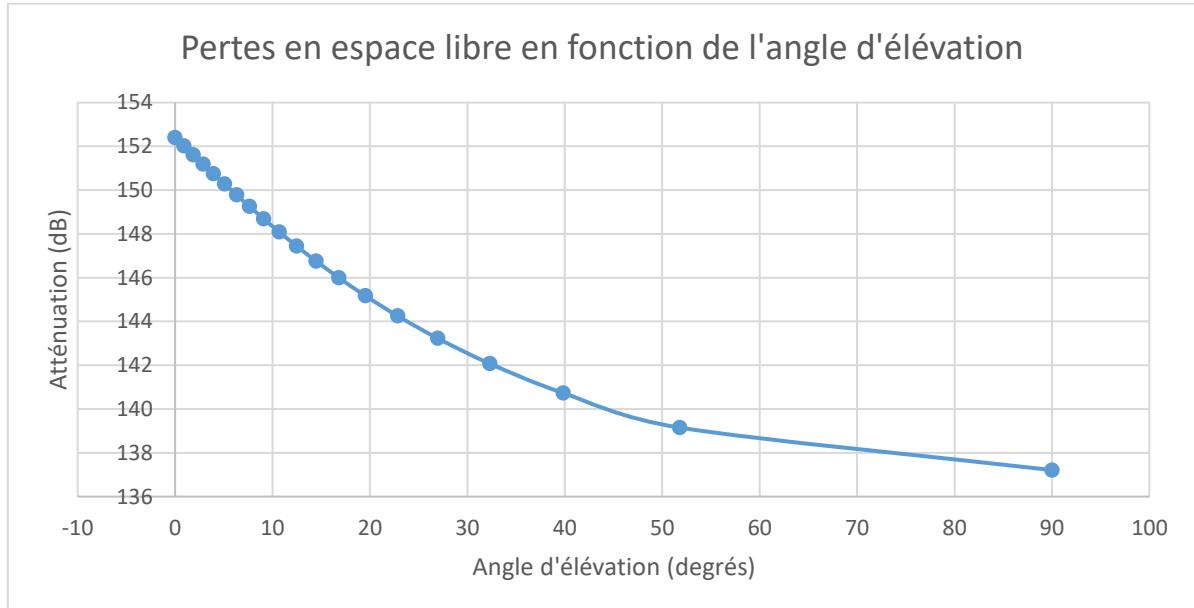


La fréquence utilisée pour la communication TM/TC (Télémesure / Télécommande) est 433,5MHz pour les deux liens, montant et descendant.

On calcule alors les pertes en espace libre (Free Space Loss) en fonction de la distance à laquelle on voit le satellite :

$$FSL = \left( \frac{4 \cdot \pi \cdot d \cdot f}{c} \right)^2 \Rightarrow FSL|_{dB} = 20 \cdot \log \left( \frac{4 \cdot \pi \cdot d \cdot f}{c} \right)$$

On peut ensuite, connaissant le lien entre la distance au satellite et l'angle d'élévation, tracer l'atténuation due aux pertes en espace libre en fonction de l'angle d'élévation :



La puissance reçue vaut en dB :

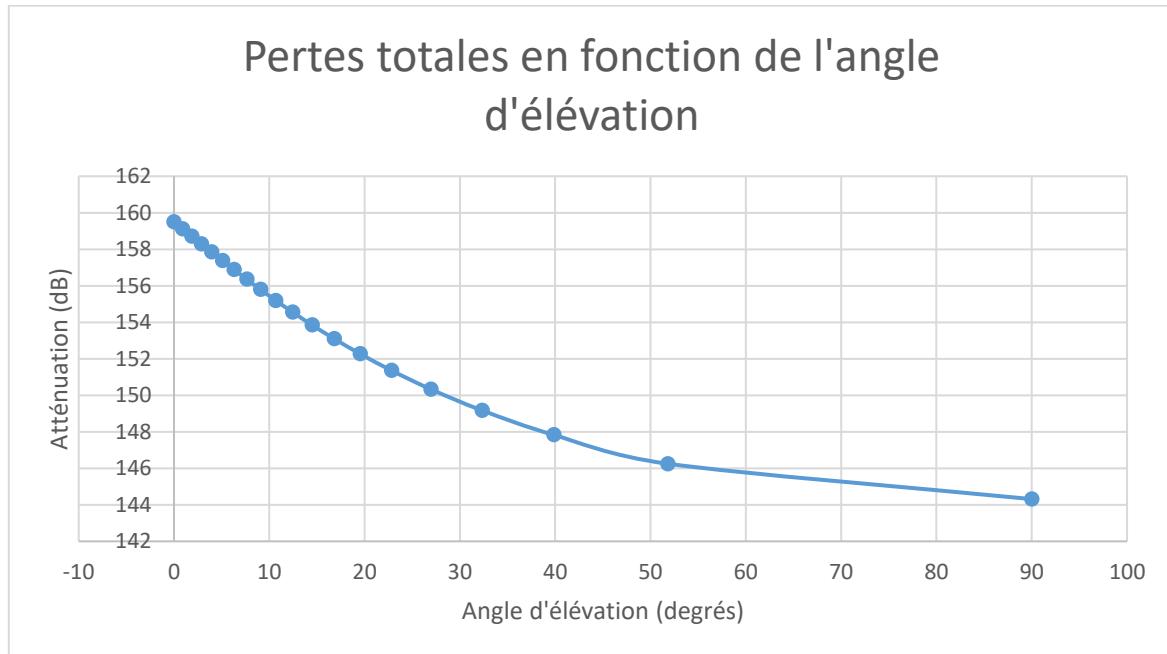
$$P_{reçue} = P_{émission} + G_{sat} + G_{sol} - Pertes$$

Le terme de pertes comprenant l'ensemble des pertes : polarisation, atmosphère, ionosphère, câbles, etc.

En ajoutant le gain du LNA et les pertes engendrées par les différents composants, on obtient au niveau de la carte de communication de la station sol :

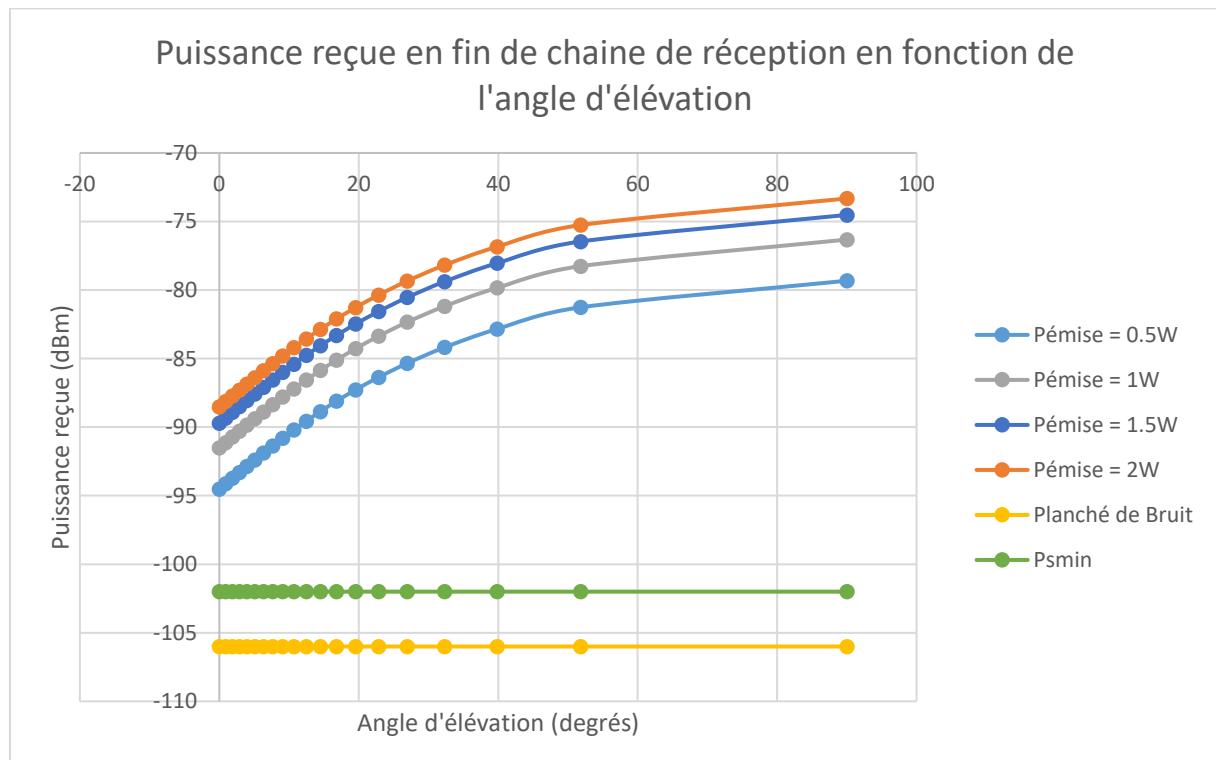
$$P'_{reçue} = P_{émission} + G_{sat} + G_{sol} - Pertes' + G_{LNA}$$

On peut alors tracer l'ensemble des pertes jusqu'à la carte communication en fonction de l'angle d'élévation :



On fixe typiquement  $G_{sat} = 0\text{dBi}$ ,  $G_{sol} = 14\text{dBi}$  et  $G_{LNA} = 24\text{dB}$ . Le matériel choisi par la suite s'approchera de ces valeurs.

On obtient alors la puissance reçue en fonction de l'angle d'élévation et de la puissance émise par le satellite :



Pour calculer la marge de liaison ainsi que le taux d'erreur binaire, il faut définir le « Carrier to Noise Ratio » - CNR :

$$\frac{C}{N} = \frac{E_b}{N_0} \cdot \frac{D}{B_w}$$

Ce ratio correspond au « Signal to Noise Ratio » - SNR (rapport signal sur bruit) mais pour le signal modulé.

$E_b$  est l'énergie par bit,  $N_0 = k \cdot T$  la densité spectrale de bruit ou  $k$  = la constante de Boltzmann et  $T$  la température de bruit de l'ensemble de la chaîne de réception,  $D$  est le débit binaire et  $B_w$  la largeur de bande considérée.

On peut exprimer :

$$\frac{E_b}{N_0} = \frac{C}{N} \cdot \frac{D}{B_w}$$

Il existe des courbes représentant le taux d'erreur binaire (BER) en fonction de ce rapport :

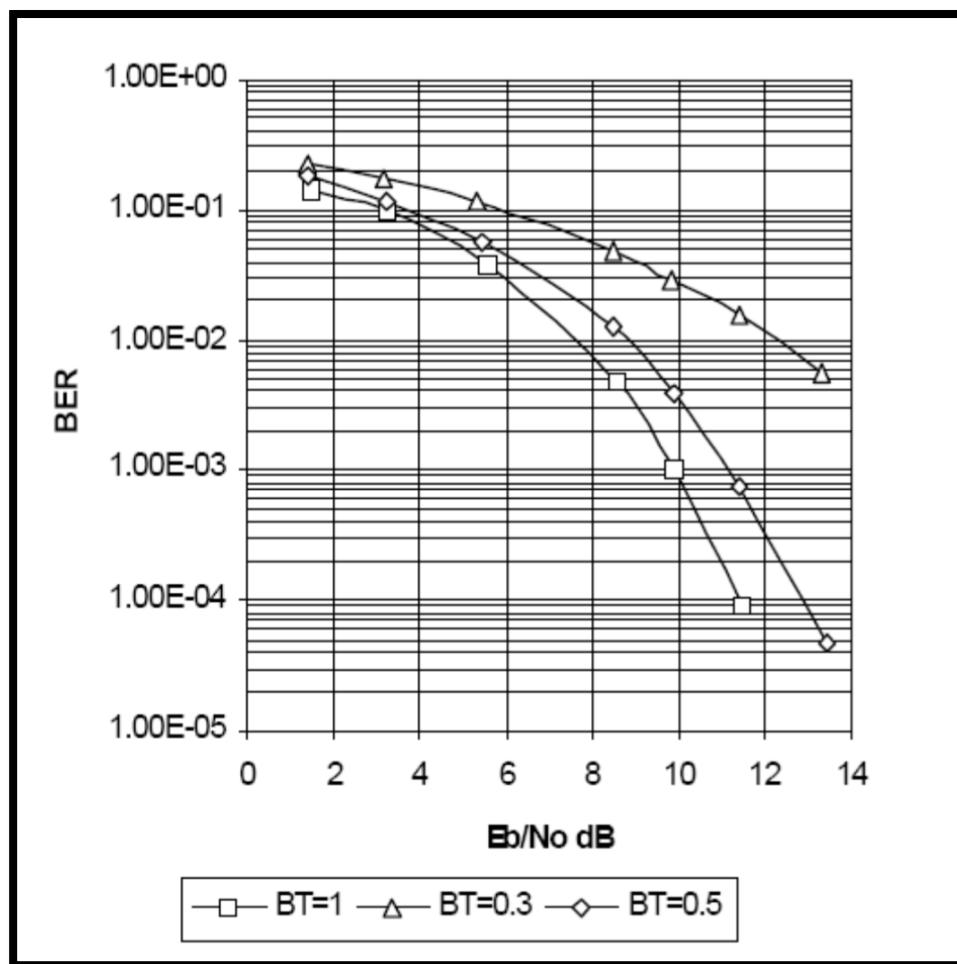


Figure 50.  $BER = f\left(\frac{E_b}{N_0}\right)$  pour un signal GMSK

On peut donc estimer celui-ci. Dans notre cas, on a :

$$\left| \frac{E_b}{N_0} \right|_{dB} = C_{dB} - N_{dB} + 10 * \log \left( \frac{D}{B_w} \right)$$

avec  $N_{dB} = 10 * \log(k \cdot B_w \cdot T)$  et  $T \approx 600K$ ,  $B_w \approx 50kHz$ , et  $D = 9,6kbps$

En sachant que pour avoir un  $BER \leq 10^{-5}$  en GMSK et sans codage canal (dans le cas où il ne serait pas implémenté par la suite), il faut une rapport  $\frac{E_b}{N_0} \geq 14dB$ , on peut calculer les rapports  $\frac{E_b}{N_0}$  pour différents angles d'élévation et différentes puissances émises par le satellite.

Élévation (°)	Puissance émise (dBW)					Eb/N0 (dB)
	-10	-3	0	1,8	3	
90	27	34	37	39	40	
52	25	32	35	37	38	
40	23	30	33	35	36	
32	22	29	32	34	35	
27	21	28	31	33	34	
23	20	27	30	32	33	
20	19	26	29	31	32	
17	18	25	28	30	31	
14	17	24	27	29	30	
12	17	24	27	28	30	
11	16	23	26	28	29	
9	15	22	25	27	28	
8	15	22	25	27	28	
6	14	21	24	26	27	
5	14	21	24	26	27	
4	13	20	23	25	26	
3	13	20	23	25	26	
2	12	19	22	24	25	
1	12	19	22	24	25	
0	12	19	22	23	25	

Figure 51. Rapports  $\frac{E_b}{N_0}$  pour différents angles d'élévation et différentes puissances émise par le satellite

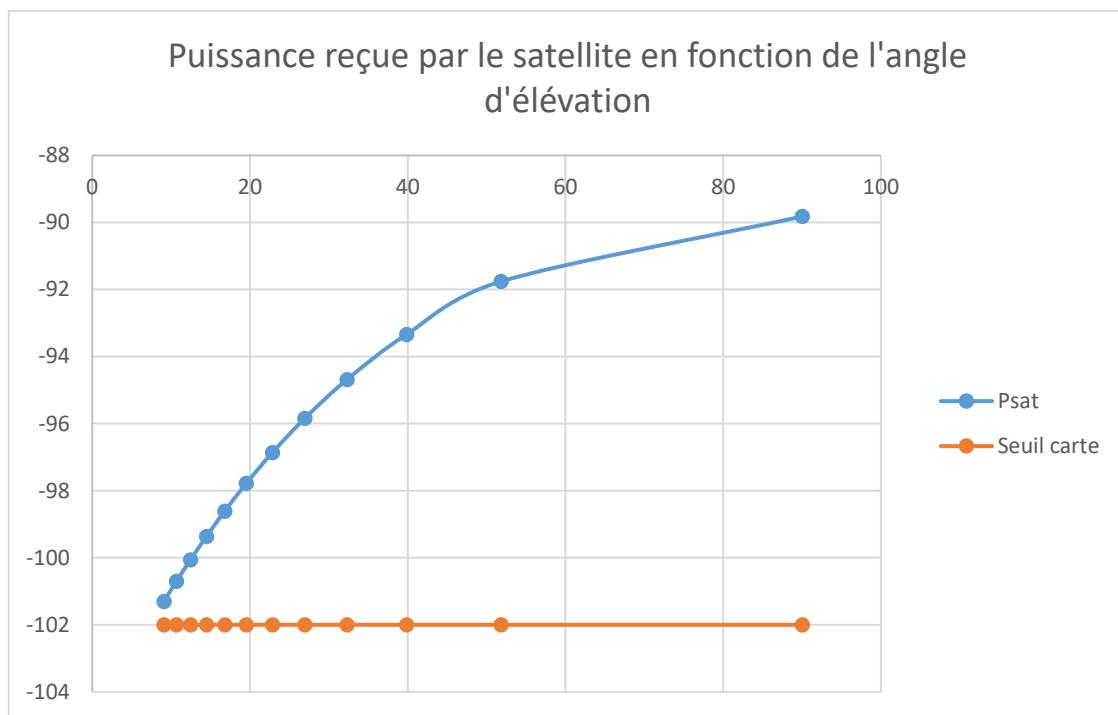
On constate que dans les cas d'utilisation habituel de la station sol (angles d'élévation  $\geq 10^\circ$ ), une puissance d'émission de 1W par le satellite est en théorie amplement suffisante, donnant la marge de liaison  $LM = \left| \frac{E_b}{N_0} \right|_{eff} - \left| \frac{E_b}{N_0} \right|_{th} > 10dB$ . De plus en rajoutant un codage du type code convolutif, pour obtenir ce même BER, le rapport Eb/N0 nécessaire devient environ 7dB, ce qui augmente d'autant plus la marge de liaison.

## 2. Lien descendant

Afin de respecter la législation, il est indispensable de se pourvoir d'une licence radioamateur, afin de pouvoir émettre plus de 10mW en puissance rayonnée. En effet, une telle licence permet de pouvoir rayonner une puissance d'environ 200W (53 dBm) au maximum.

Nous avons donc choisi un amplificateur de puissance de 5W (37dBm), ce qui nous donne une PIRE de 51 dBm (environ 125W) en prenant en compte une antenne qui a un gain de 14dB.

On peut alors tracer la puissance reçue au niveau du satellite en fonction de l'angle d'élévation, et la comparer au seuil de sensibilité de la carte communication qui sera embarquée dans le satellite (comme nous ne savons pas exactement de laquelle il s'agira exactement, nous avons pris une valeur classique des cartes du commerce, pour la modulation souhaitée (GMSK) et le débit souhaité (9,6Kbps), avec un  $BER < 10^{-5}$ .



Les résultats montrent qu'en théorie le lien montant est assez robuste, il est même encore possible d'augmenter la fiabilité de celui-ci en diminuant le débit de données (en passant par exemple à 1.2 Kbps), pour avoir une meilleure sensibilité au niveau de la carte du satellite. Cela semble envisageable car les commandes envoyées au satellite ne devraient pas être lourdes, ni demander un débit conséquent.

Nous avons également confirmé les résultats de ce bilan de liaison en utilisant une fiche de calcul Excel fournie par la société radioamateur britannique (AMSAT-UK). Ce bilan est disponible en annexe.

## B. Design de l'antenne

### 1. Cahier des charges

On utilise généralement des antennes à polarisation circulaire pour les communications spatiales car même si l'onde émise par le satellite est polarisée linéairement, on peut réceptionner celle-ci avec 3dB de pertes dans toutes les positions du satellite.

Si on utilisait des antennes à polarisation linéaire, il faudrait veiller à l'alignement des polarisations, ce qui n'est pas évident, afin d'éviter des pertes de signal trop importantes qui peuvent atteindre 30dB.

L'utilisation de la polarisation circulaire à une extrémité seulement donne une perte d'environ -3 dB. Afin de bénéficier de tous les avantages de la polarisation circulaire, les liens montant et descendant doivent l'utiliser.

Le tableau ci-dessous montre la relation entre polarisations Horizontal, Vertical, RHCP (polarisation circulaire droite) et LHCP (polarisation circulaire gauche) et les pertes en dB (la polarisation RHCP est également connue sous le nom de polarisation CW (ClockWise) et la LHCP en tant que polarisation CCW (Counter ClockWise)).

	Horizontal	Vertical	RHCP	LHCP
Horizontal	0	30	3	3
Vertical	30	0	3	3
RHCP	3	3	0	30
LHCP	3	3	30	0

Figure 52. Pertes selon les différentes polarisations

Ainsi, pour construire l'antenne de la station sol, adaptée au cubesat de Supsat, celle-ci doit respecter certains critères, à savoir :

- Fréquence centrale de 433,5MHz
- Polarisation circulaire
- Gain élevé ( $\geq 14$ dB)
- Directive pour éviter les signaux parasites
- Masse et centre de gravité limités par la structure du suivi

## 2. Choix

Pour respecter ce cahier des charges, il existe principalement deux types d'antennes utilisables : l'antenne hélice axiale et l'antenne crossed-Yagi dont les caractéristiques sont données en annexe.

Cependant, après avoir discuté avec le professeur de l'IETR M. Himdi, nous avons décidé d'innover en mettant en place la fabrication d'une antenne patch, plus précisément un réseau d'antenne patch, respectant le cahier des charges.

Nous avons donc dans un premier temps effectué des simulations de ce type d'antenne sous CST. Pour patch alimenté par connecteur coaxial 50 Ohm :

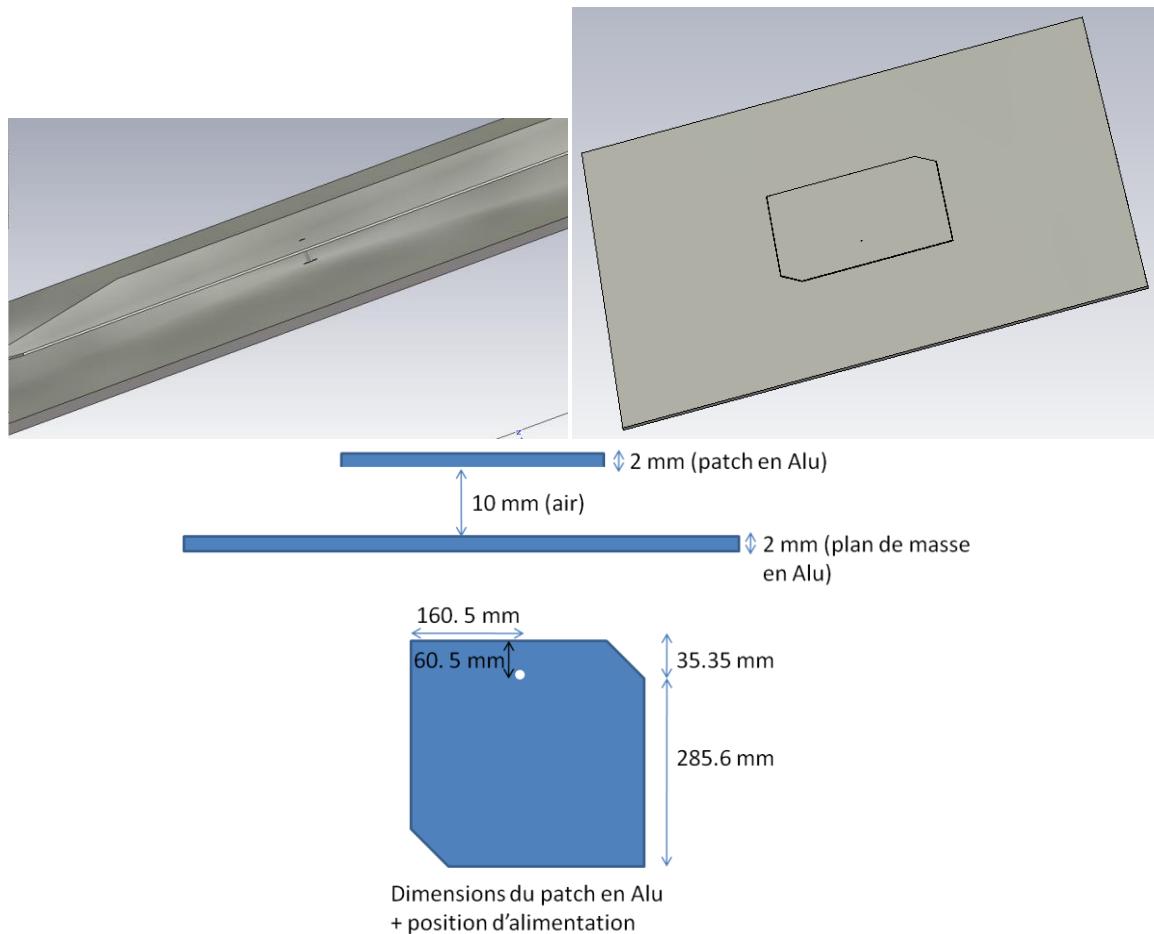


Figure 53. Caractéristiques mécaniques d'une 'antenne patch'

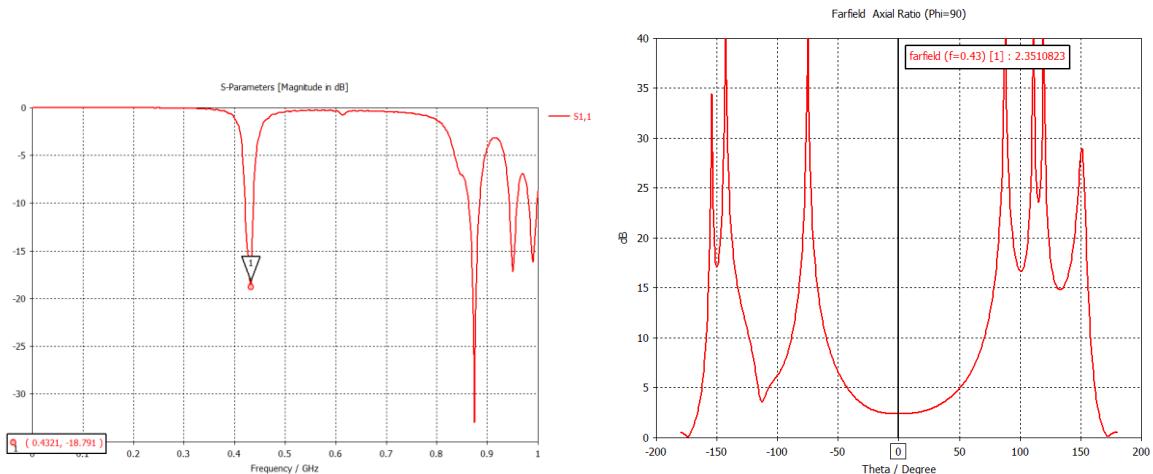


Figure 54. Adaptation d'entrée et rapport d'ellipticité

Plus le rapport d'ellipticité est faible, meilleur est l'adaptation à une polarisation circulaire.

Ensuite, pour un réseau de 4 patchs alimentés par un circuit en rotation séquentielle (ce qui permet d'améliorer la bande passante).

Les simulations du patch nous donnent alors un patch de  $30\text{cm}^2$ . L'adaptation est réalisée en plaçant l'alimentation à 60mm du bord et les graphes nous le confirment tout comme le respect de la polarisation circulaire.

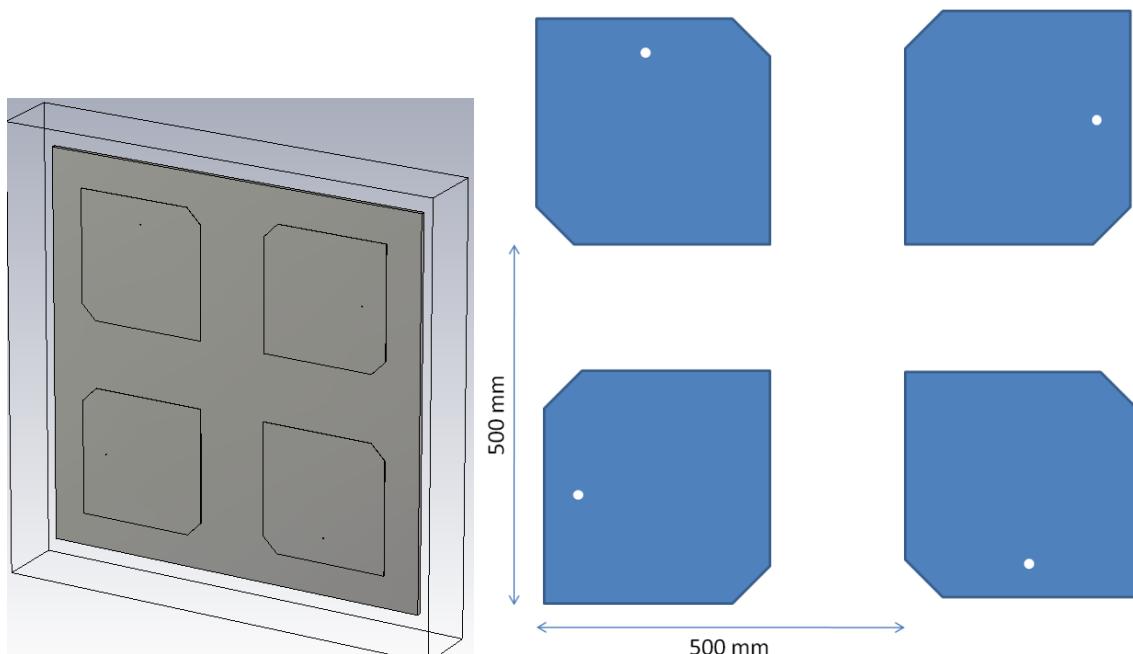


Figure 55. Caractéristiques mécaniques du réseau de patchs

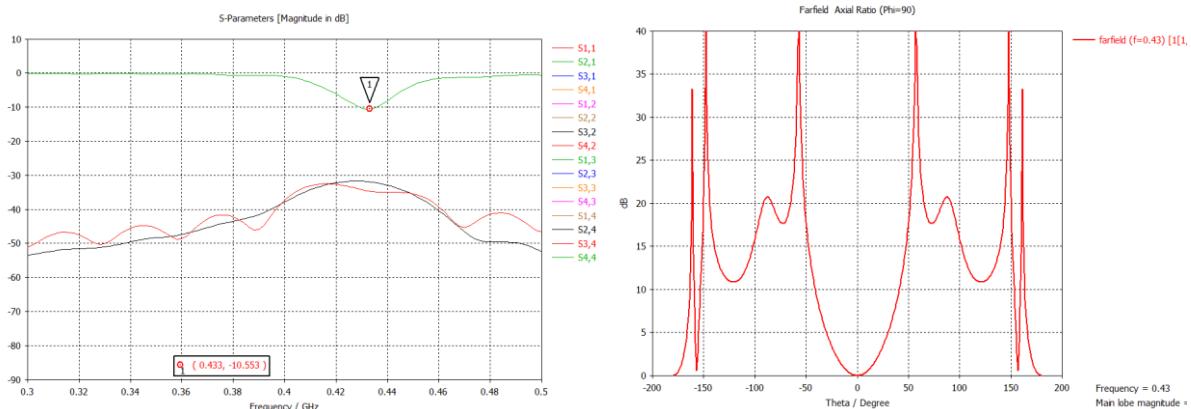


Figure 56. Adaptation d'entrée et rapport d'ellipticité

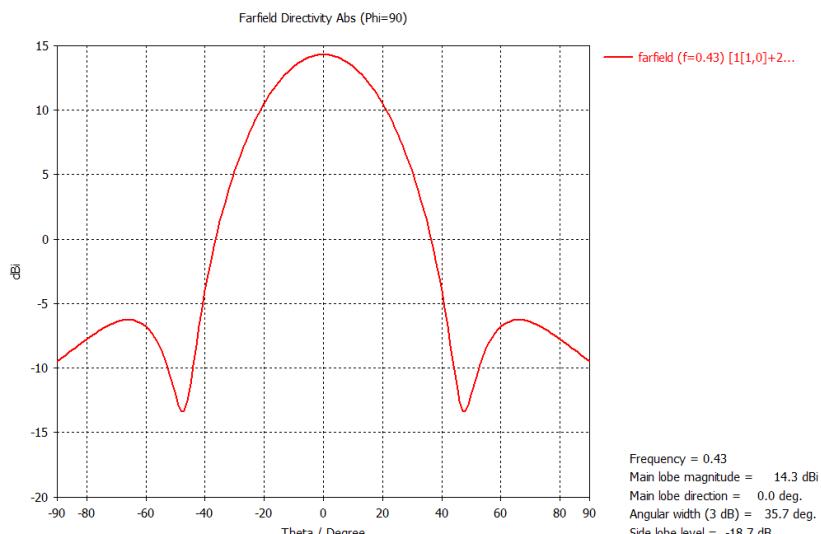


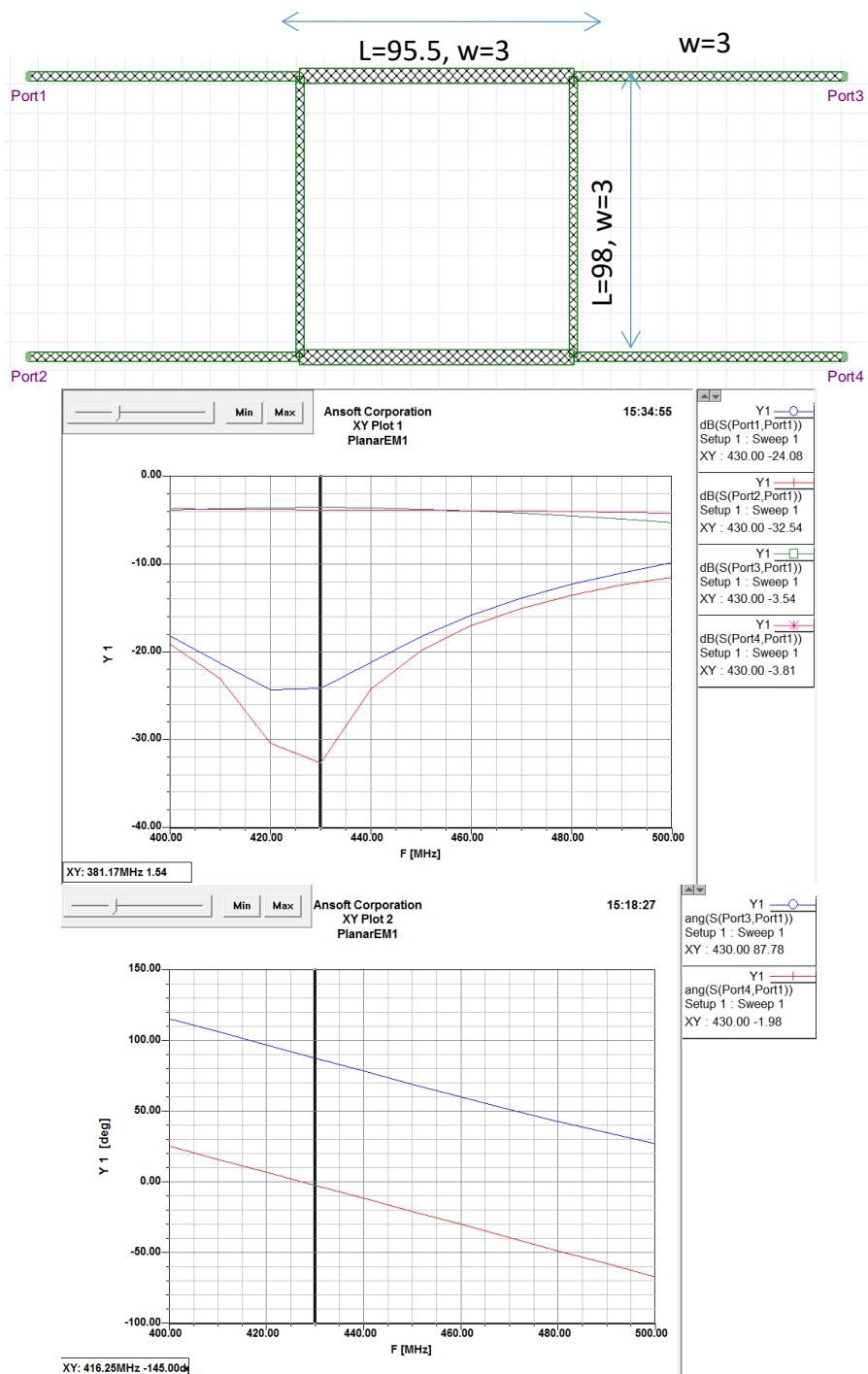
Figure 57. Diagramme de gain

Le réseau de patch, qui fera  $1m^2$ , nous fournit un gain de 14dB pour une polarisation circulaire pour la fréquence nous intéressante. Le cahier des charges est donc respecté.

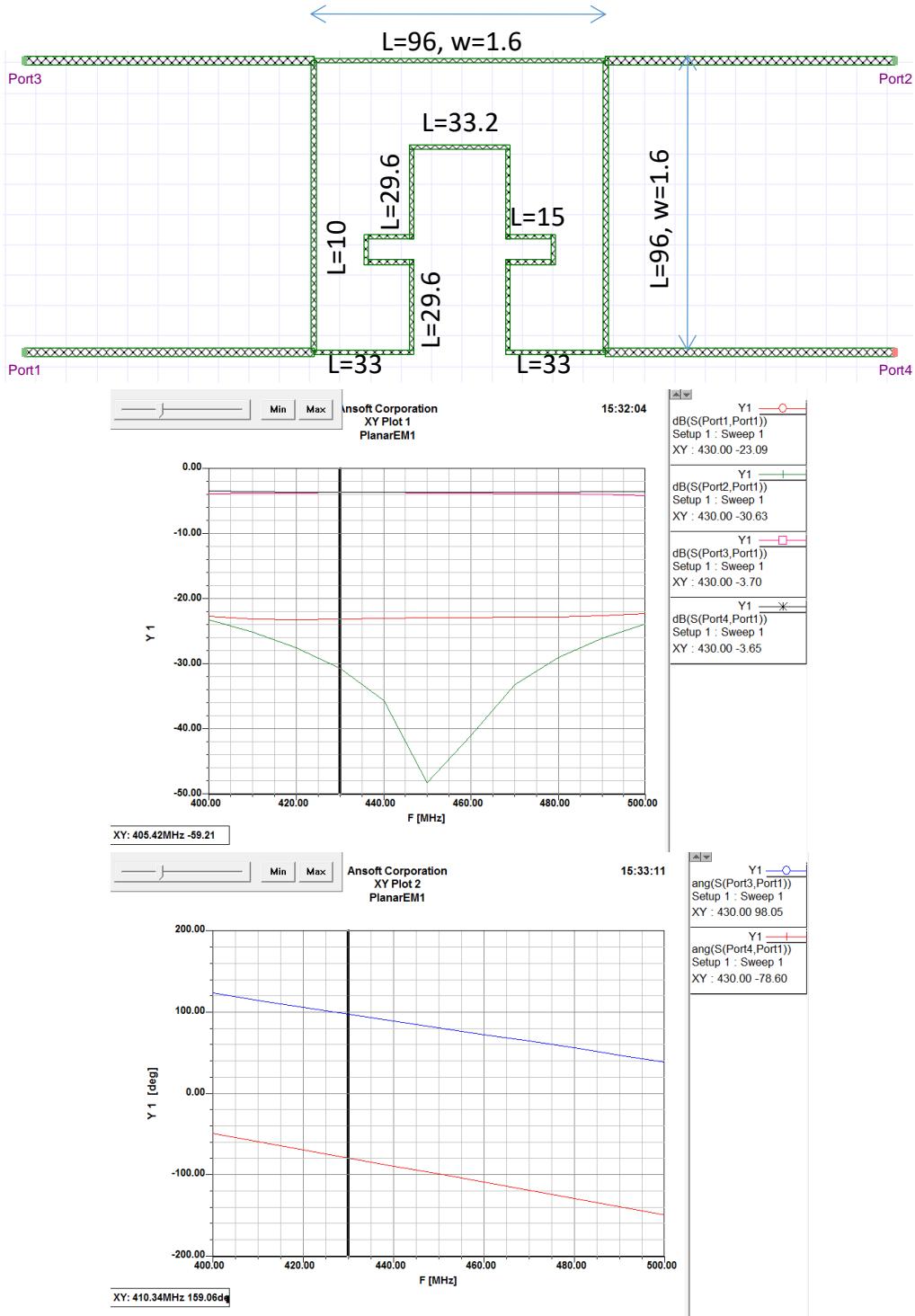
### 3. Alimentation du réseau

Nous avons également simulé à l'aide de CST, le réseau d'alimentation des 4 patchs avec : substrat FR4 :  $\epsilon_r=4.3$ ,  $h=1.586$  mm cuivre :  $18\mu m$  d'épaisseur

### Coupleur 3dB/90°

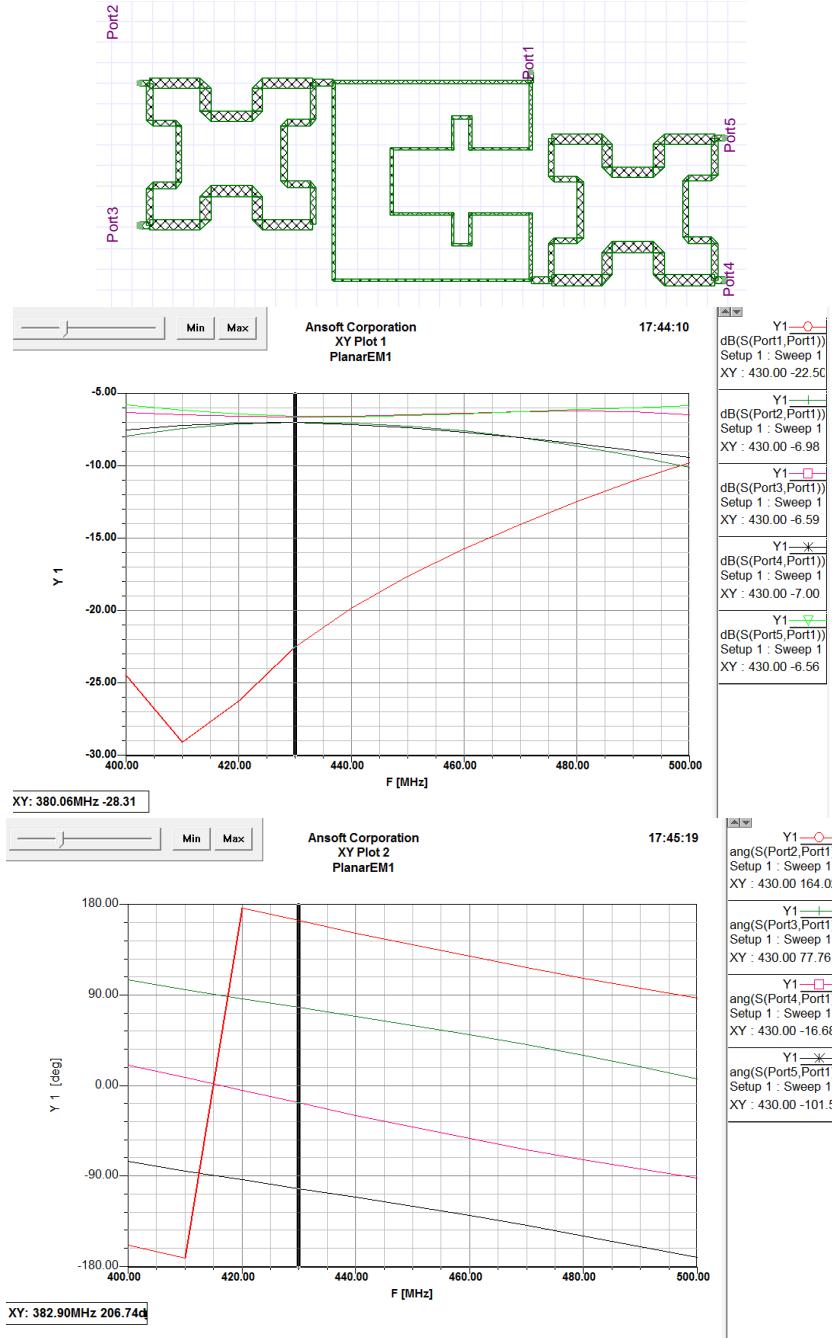


## Coupleur 3dB/180°



Les premiers graphiques correspondent au paramètre S du coupleur. Les entrées sont bien isolées pour notre fréquence de travail et la puissance aux sorties est bien équitablement répartie avec la perte de 3dB pour le premier coupleur et de 6dB pour le second. Les seconds graphiques nous montrent que les signaux sont bien déphasés de 90° pour le premier et de 180° pour le deuxième ce qui permet d'avoir tous les signaux déphasés deux à deux de 90°.

## Alimentation en rotation séquentielle des 4 patchs



Cette alimentation nous permet de répartir uniformément la puissance du signal entre chaque patch. Et c'est le cas d'après la simulation, nous avons une perte de 6dB entre chaque patch, ce qui signifie que le signal est divisé par 4. Et tous les patchs sont déphasés de 90° deux à deux, ce qui permet d'émettre et recevoir des signaux polarisés circulairement.

Dans un premier design, le plan de masse, ainsi que les patchs étaient conçus en aluminium plein, de 2mm d'épaisseur. Cependant, afin de limiter le poids total de l'antenne et également la

sensibilité au vent (qui est très importante avec de grandes plaques) l'antenne finale sera construite en aluminium grillagé, ce qui n'altère pas ses performances mais permet de palier efficacement aux problèmes suscités.

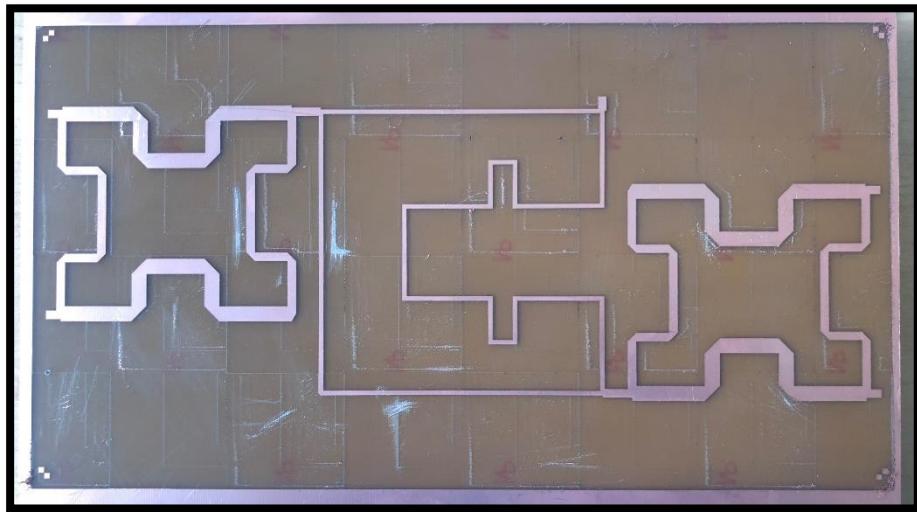


Figure 58. *Circuit d'alimentation réalisé à l'IETR*

#### 4. Essai

Nous avons pu réaliser un premier essai de l'antenne lors du passage d'un cubesat. Grâce au site N2YO (qui nous renseigne sur la position en temps réel des satellites), nous avons placé le prototype de l'antenne dehors lorsqu'un cubesat fonctionnant à la même fréquence que notre cubesat fut au-dessus de nous. En reliant l'antenne à un analyseur de spectre, nous avons pu observer les signaux envoyés par les satellites.

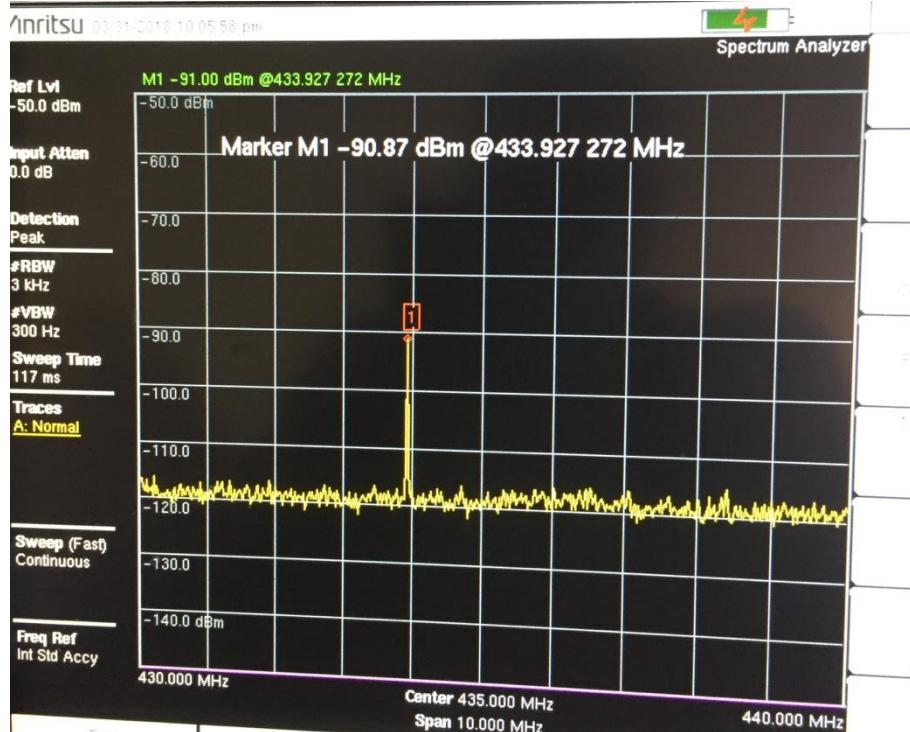


Figure 59. Pic à la fréquence d'émission du cubesat reçu par l'antenne

L'antenne semble présenter de bonnes performances de réception dans notre bande de fréquence. En effet, le pic que l'on voit sur la photo correspond au signal d'un cubesat. Cela nous confirme donc que notre antenne est correcte pour son utilisation.

## 5. Prototype final de l'antenne

Avec l'aide de l'IETR, nous avons finalement un prototype d'antenne viable et correspondant à notre cahier des charges. Les patchs sont protégés et connectés au circuit d'alimentation. L'antenne est renforcée via des profilés en aluminium et des supports en plastique permettent de la fixer à l'axe de la structure de la station sol.

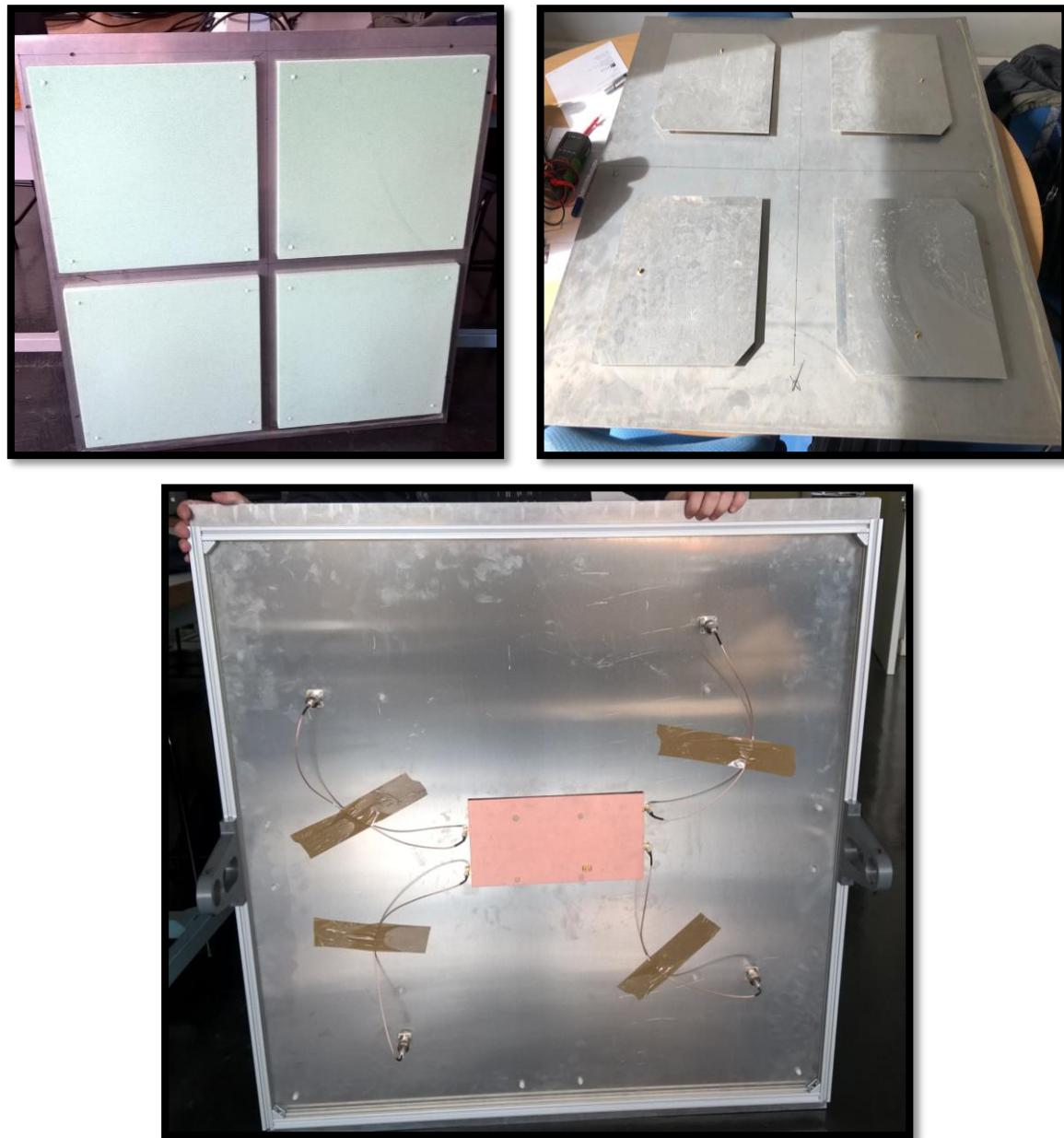


Figure 60. Prototype final de l'antenne

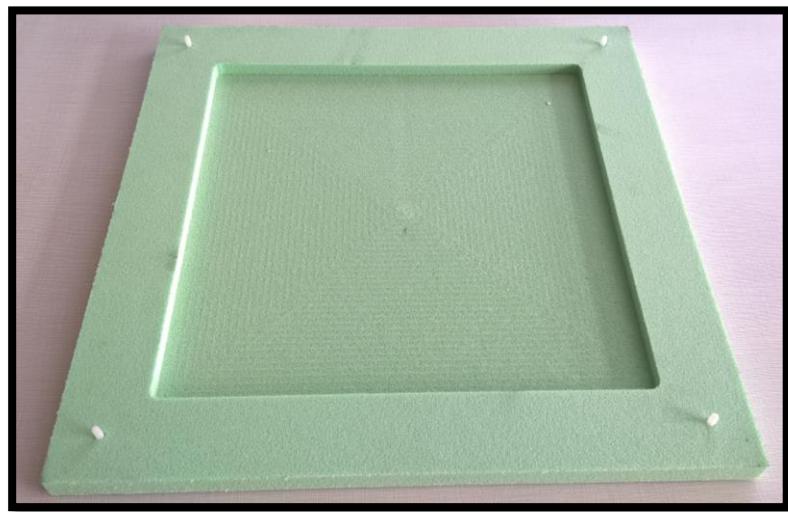
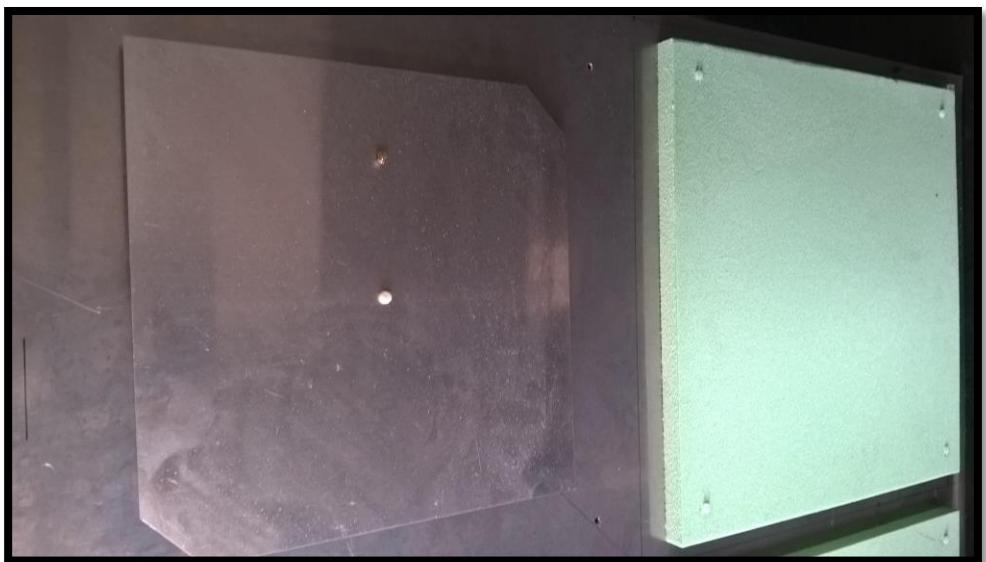


Figure 61. Aperçu d'un patch

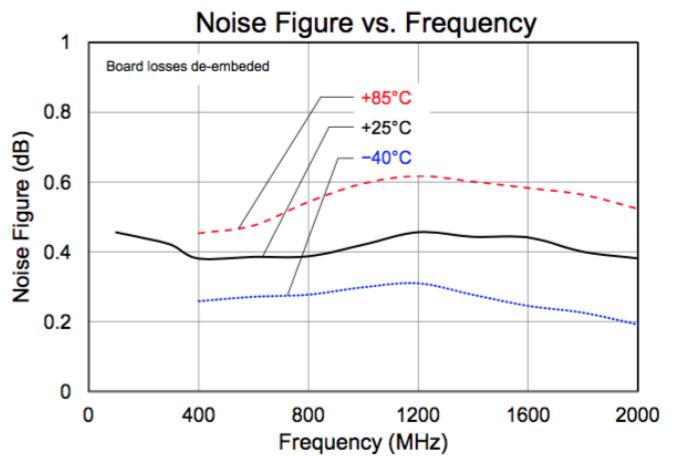
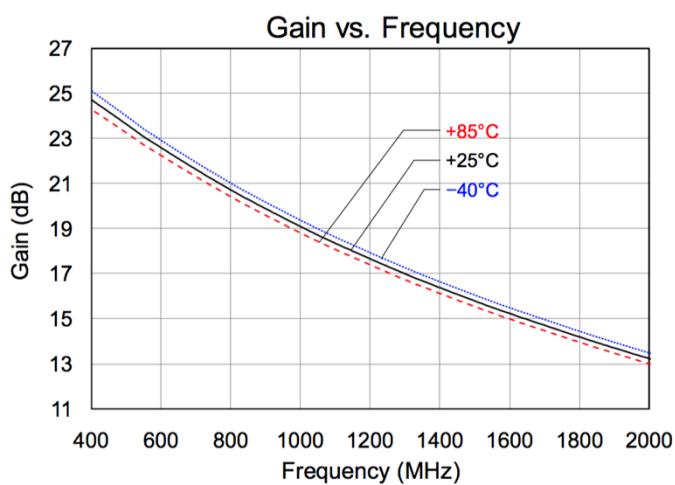
## C. Circuit radiofréquence

Afin de maximiser le potentiel éducatif et professionnalisa nt du projet, nous avons décidé de réaliser nous même le circuit RF d'émission-réception, plutôt que de simplement acheter et connecter des composants tout prêts. Cette démarche s'inscrit également dans un but de réduction des pertes et du bruit (pas de longs câbles coaxiaux qui relient les différentes parties, etc...), de coût, et de miniaturisation/portabilité.

### 1. LNA

Afin de garantir la qualité du signal qui sera traité numériquement, il faut impérativement ne pas l'abîmer après sa réception par l'antenne. Mais ce signal au niveau de l'antenne est tellement faible (-145dBm) qu'il faut tout de même l'amplifier. D'où l'utilité du LNA (Low Noise Amplifier), qui est un amplificateur à faible bruit, c'est-à-dire qu'il amplifie très bien le signal dans la bande de fréquence pour laquelle il est conçu pour fonctionner tout en ajoutant très peu de bruits (facteur de bruit < 1).

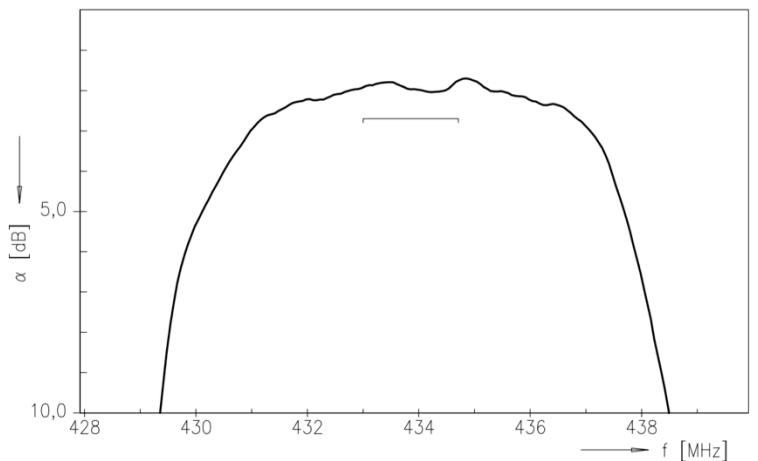
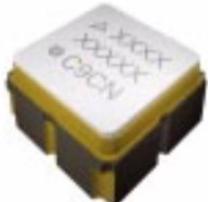
Le LNA que nous avons sélectionné est le **QORVO TQP3M9036** qui dispose d'excellentes performances radiofréquences dans la bande qui nous intéresse. Son gain est de 24dB à 433.5MHz et un facteur de bruit très faible : 0,40dB.



### 2. Filtre de réception

Pour limiter la bande écoutée et le bruit au niveau de la carte de communication, nous avons ajouté dans la piste de réception un filtre à ondes acoustique de surface (SAW), qui a une bande passante de 2MHz autour de notre fréquence de fonctionnement 433.5MHz.

Il s'agit du **B3710** de Qualcomm



### 3. Amplificateur de puissance

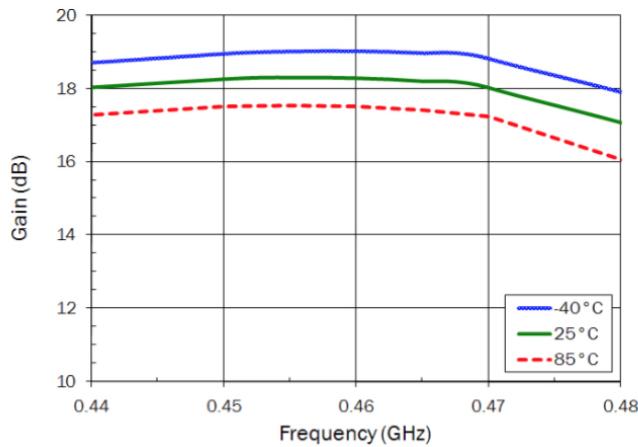
Afin de pouvoir émettre la puissance souhaitée, nous avons déterminé qu'il nous faudrait un amplificateur RF d'environ 5W. Nous avons donc choisi le **RFPA3800 de chez RFMD** qui permet d'obtenir les 5W désirés en sortie, à 433,5MHz.

Le gain de cet amplificateur est d'environ 18dB, ce qui permet d'obtenir les 5W en sortie lorsque le signal à l'entrée de celui-ci est d'environ 20dBm. Ces valeurs permettent aussi de travailler dans la zone où le rendement de l'amplificateur est le plus grand

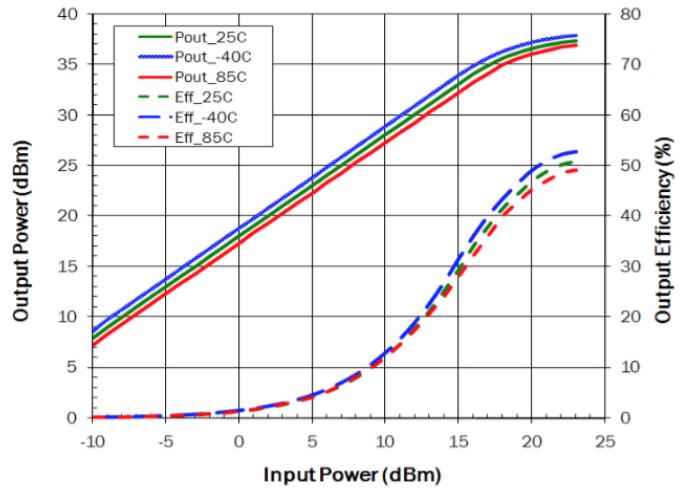
Enfin, ses S11 et S22 sont d'environ -10dB et son S12 de -30dB.



**S21 versus Frequency**



**Pout versus Pin @460MHz**

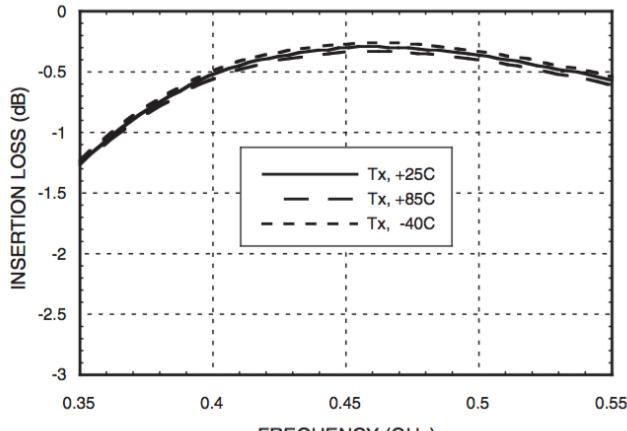


### 4. Switch

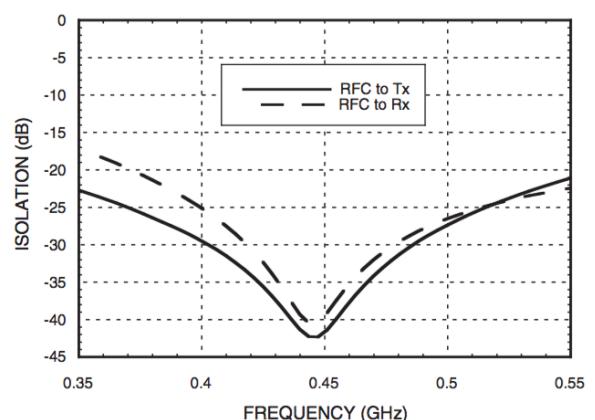
Étant donné que nous utilisons une unique antenne pour l'émission et la réception, nous avons besoin de pouvoir commuter entre ces deux voies, tout en les maintenant isolées l'une par rapport à l'autre. Il faut également que le switch n'ajoute pas trop de bruit au signal, puisqu'il est le premier composant juste après l'antenne, et enfin il doit être capable d'être traversé par au moins 5W de puissance (et de faire en sorte que cette puissance soit exclusivement délivrée à l'antenne).

Nous avons choisi le **HMC546MS8G de Analog Devices** qui a de faibles pertes par insertion (<0,4dB), peut être traversé par 10W de puissance, et offre une très importante isolation (30-40dB) ainsi qu'une excellente linéarité (IIP3 élevé).

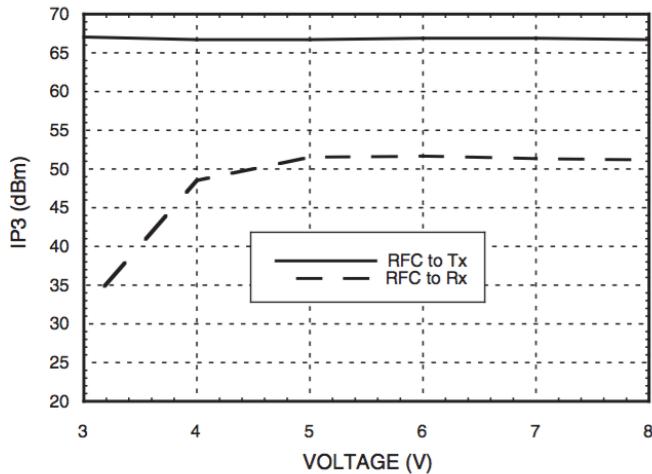
**Insertion Loss vs. Temperature, Tx with 457 MHz Tuning**



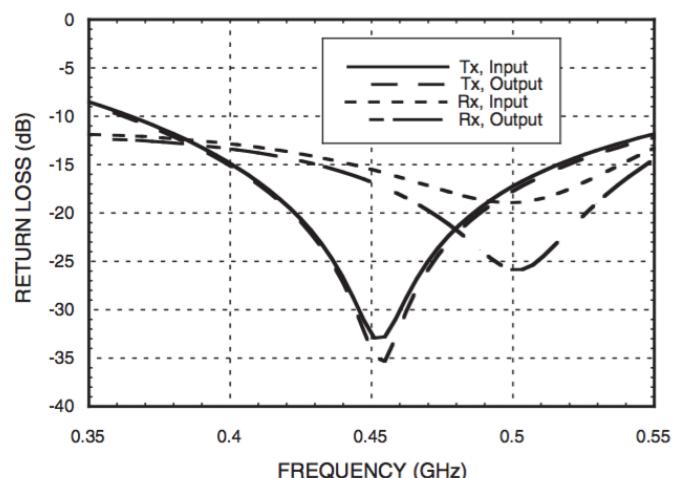
**Isolation with 457 MHz Tuning**



## **Input IP3 vs. Voltage with 457 MHz Tuning**



## **Return Loss with 457 MHz Tuning**



## **5. Tranceiver**

Afin de pourvoir aux spécifications du lien de télécommunication envisagé, nous avons choisi un module du commerce permettant de faire émission et réception. Il s'agit du **Si4463 de Silicon Labs**. Il supporte les modulations GMSK, BFSK etc... qui seront utilisées. De plus, cette carte permet d'introduire des codes correcteurs d'erreurs et dispose d'une excellente sensibilité.

Le Si4463 présente de bonnes performances énergétiques. En effet, la consommation du trameceiver est plutôt faible que ce soit en réception ou même en émission.



RX Mode Current	$I_{RXH}$	High Performance Mode	—	13.7	—	mA
	$I_{RXL}$	Low Power Mode <sup>2</sup>	—	10.7	—	mA
TX Mode Current (Si4464/63)	$I_{TX\_+20}$	+20 dBm output power, class-E match, 915 MHz, 3.3 V	—	85	—	mA
		+20 dBm output power, class-E match, 460 MHz, 3.3 V	—	75	—	mA
		+20 dBm output power, square-wave match, 169 MHz, 3.3 V	—	70	—	mA

Figure 62. Consommation du Si4463

D'autre part, celui-ci permet la mise en place des modulations de fréquence souhaitées aux débits souhaités :

(G)FSK Data Rate <sup>3,4</sup>	DR <sub>FSK</sub>	0.1	—	500	kbps
4(G)FSK Data Rate <sup>3,4</sup>	DR <sub>4FSK</sub>	0.2	—	1000	kbps
OOK Data Rate <sup>3,4</sup>	DR <sub>OOK</sub>	0.1	—	120	kbps

Figure 63. Débit des différentes modulations

Enfin, ce tranceiver présente une sensibilité très élevée en réception. Étant donné que le débit objectif est entre 1,2Kbps et 9,6 Kbps, on peut prévoir une sensibilité inférieure à -110/-115dB.

RX Sensitivity	$P_{RX\_0.5}$	(BER < 0.1%) (500 bps, GFSK, BT = 0.5, $\Delta f = \pm 250\text{Hz}$ ) <sup>3</sup>	—	-126	—	dBm
	$P_{RX\_40}$	(BER < 0.1%) (40 kbps, GFSK, BT = 0.5, $\Delta f = \pm 20\text{ kHz}$ ) <sup>3</sup>	—	-110	—	dBm
	$P_{RX\_100}$	(BER < 0.1%) (100 kbps, GFSK, BT = 0.5, $\Delta f = \pm 50\text{ kHz}$ ) <sup>1</sup>	—	-106	—	dBm
	$P_{RX\_125}$	(BER < 0.1%) (125 kbps, GFSK, BT = 0.5, $\Delta f = \pm 62.5\text{ kHz}$ ) <sup>3</sup>	—	-105	—	dBm
	$P_{RX\_500}$	(BER < 0.1%) (500 kbps, GFSK, BT = 0.5, $\Delta f = \pm 250\text{ kHz}$ ) <sup>3</sup>	—	-97	—	dBm
	$P_{RX\_9.6}$	(PER 1%) (9.6 kbps, 4GFSK, BT = 0.5, $\Delta f = \pm 2.4\text{ kHz}$ ) <sup>3,4</sup>	—	-110	—	dBm
	$P_{RX\_1M}$	(PER 1%) (1 Mbps, 4GFSK, BT = 0.5, inner deviation = 83.3 kHz) <sup>3,4</sup>	—	-88	—	dBm
	$P_{RX\_OOK}$	(BER < 0.1%, 4.8 kbps, 350 kHz BW, OOK, PN15 data) <sup>3</sup>	—	-110	—	dBm
		(BER < 0.1%, 40 kbps, 350 kHz BW, OOK, PN15 data) <sup>3</sup>	—	-104	—	dBm
		(BER < 0.1%, 120 kbps, 350 kHz BW, OOK, PN15 data) <sup>3</sup>	—	-99	—	dBm
RX Channel Bandwidth <sup>5</sup>	BW		1.1	—	850	kHz

Figure 64. *Sensibilités et bande passante*

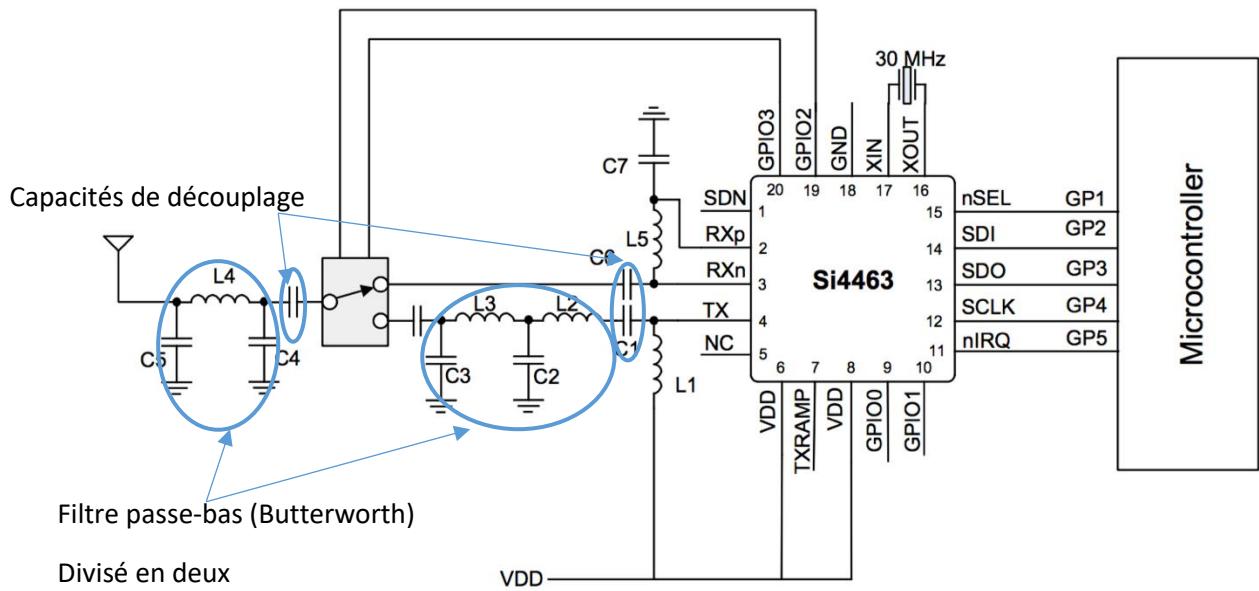
## 6. PCB

L'objectif est de réaliser un unique circuit, avec l'ensemble de ces composants, sur une unique carte.

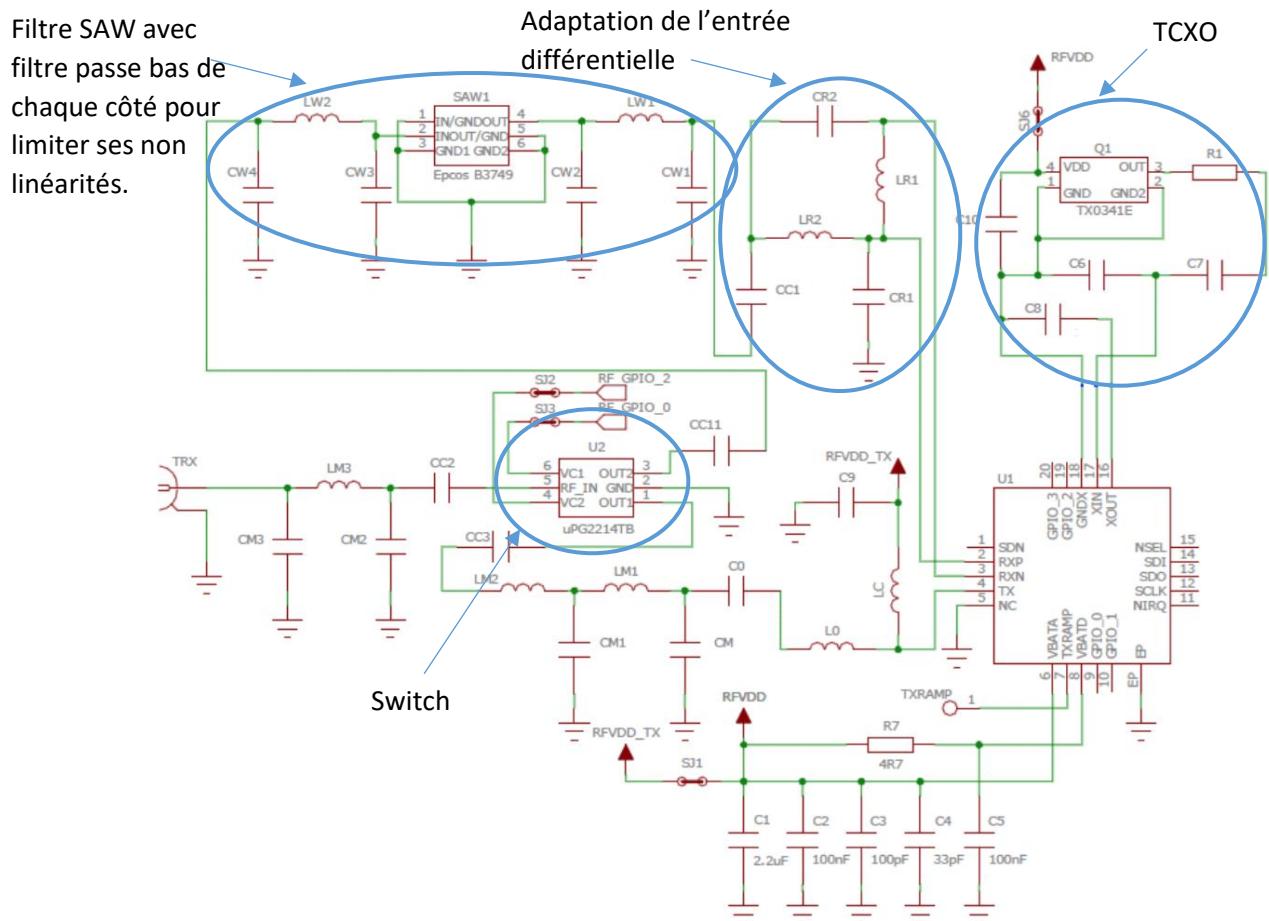
Le tranceiver sera alors contrôlé directement par l'ESP32 (qui contrôle également les moteurs) par SPI, et les commandes ainsi que les données seront envoyées par une interface WiFi, rendant le système portatif et autonome en l'alimentant par batterie.

Pour cela nous avons dû étudier très attentivement les datasheet de chaque composant afin de réaliser correctement les adaptations d'impédance de chaque composant, et à la bonne fréquence (433.5MHz). De plus, nous avons dû effectuer un gros travail de conception sur le PCB afin de respecter au mieux les règles de l'art de la conception de circuits radiofréquence.

Le principe de la conception de la carte RF est le suivant (ici sans faire apparaître les autres composants : HPA, LNA, SAW...) :



Voici un schéma plus complet et plus proche de celui que nous avons produit, celui-ci présentant un filtre SAW et un oscillateur TCXO.



La datasheet du SI 4463 nous aide à calculer les valeurs des composants pour effectuer l'adaptation.

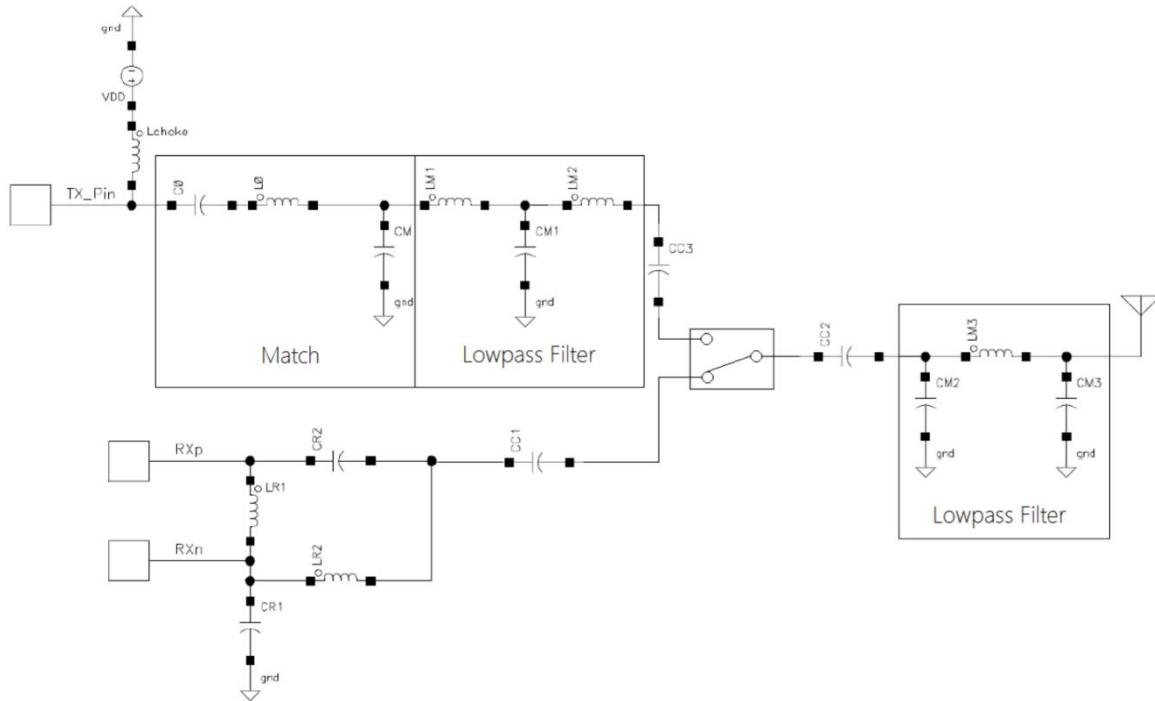


Figure 65. Topologie du circuit d'adaptation

La topologie donnée est encore une fois sans prendre en compte de filtre SAW, de HPA ou de LNA. Cependant, ces circuits permettent d'adapter à 500Ω la sortie TX et l'entrée RX, selon la fréquence. On veillera donc à faire de même avec chacun des autres composants afin que l'ensemble soit parfaitement adapté, pour notre fréquence de fonctionnement.

Freq band	RX Side				TX Side									
	LR1	LR2	CR1	CR2	Lchoke	L0	C0	CM	LM1	CM1	LM2	CM2	LM3	CM3
390 MHz	75 nH	68 nH	5.6 pF	2.4 pF	270 nH	36 nH	10 pF	18 pF	27 nH	12 pF	22 nH	12 pF	18 nH	12 pF
460 MHz	56 nH	51 nH	3.9 pF	2.2 pF	220 nH	30 nH	8.2 pF	13 pF	27 nH	9.1 pF	22 nH	9.1 pF	15 nH	9.1 pF

Figure 66. Valeurs des composants utilisés pour l'adaptation

Chacune de ces valeurs se calcule à partir de formules fournies dans la datasheet afin d'avoir des valeurs plus précises pour notre fréquence de fonctionnement. Néanmoins, lorsque l'on effectue le calcul pour 433.5MHz, on trouve des valeurs très proches de celles pour 460MHz. De plus, il faut trouver un compromis sur les valeurs des composants car toutes les valeurs ne sont pas disponibles dans le commerce (sans parler des formats qui peuvent varier selon la valeur, ce qui complique le placement des éléments, le prix qui peut être beaucoup plus élevé pour des valeurs atypiques, etc...).

Le premier composant que nous avons ajouté est le LNA.

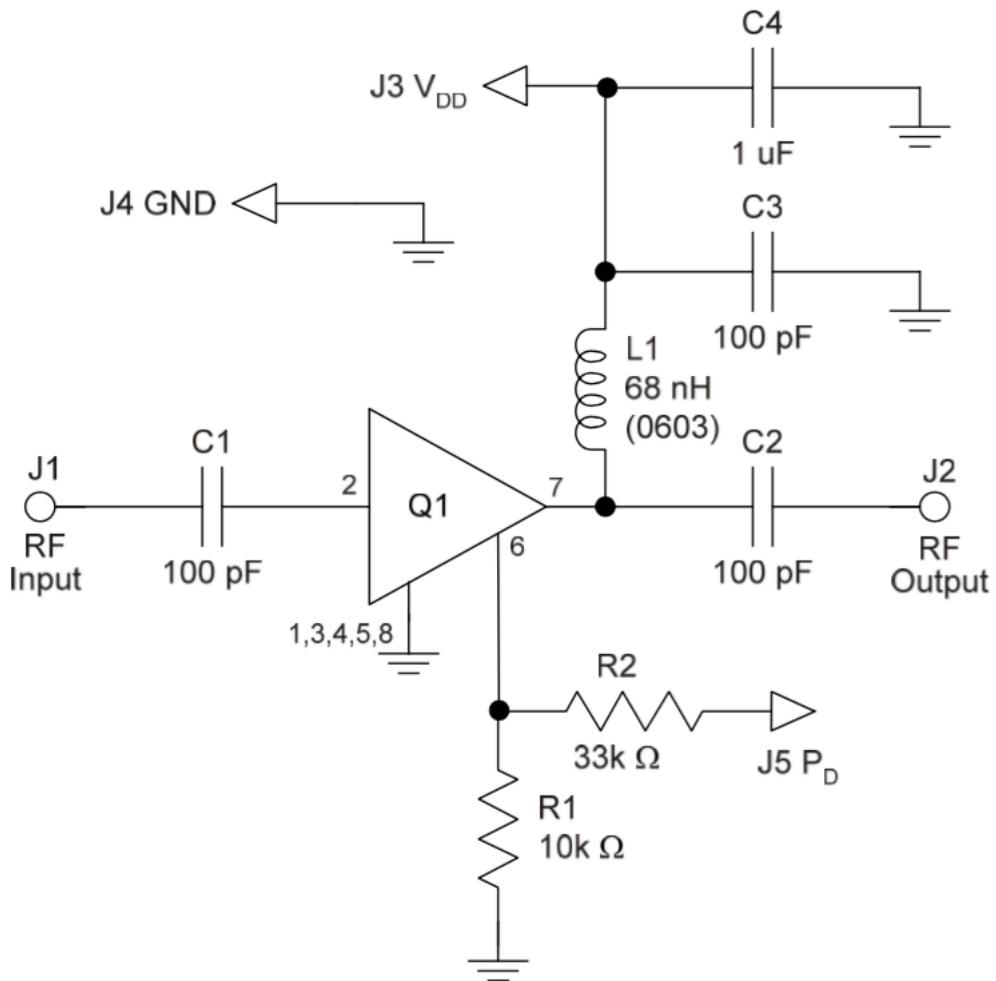


Figure 67. Topologie du circuit d'adaptation du LNA

Reference Des.	Value	Description
U1	n/a	TQP3M9037 Amplifier, 2x2 SLP Package
R1	10K Ω	Resistor, Chip, 0402, 5%, 1/16W
R2	33K Ω	Resistor, Chip, 0402, 5%, 1/16W
R3	0 Ω	Resistor, Chip, 0402, 5%, 1/16W
L1	68 nH	Inductor, 0603, 5%, Ceramic
C4	1.0 uF	Cap., Chip, 0402, 10%, 10V, X5R
C1, C2, C3, C5, C6	100 pF	Cap., Chip, 0402, 5%, 50V, NPO/COG
J3, J4, J5	n/a	Solder Turret

Figure 68. Valeurs et références des composants du circuit d'adaptation

La valeur de chaque composant est donnée afin de travailler en bande large (100MHz – 2GHz), des améliorations des performances dans la bande de fréquence utilisée (433MHz) sont donc

envisageable en adaptant les valeurs des composants, ce qui permettrait de réduire encore le facteur de bruit. Néanmoins, le facteur de bruit étant déjà très faible, ces modifications ne semblent pas nécessaires dans l'immédiat.

Devant le LNA, on trouve le switch qui permet de sélectionner la voie RX ou la voie TX.

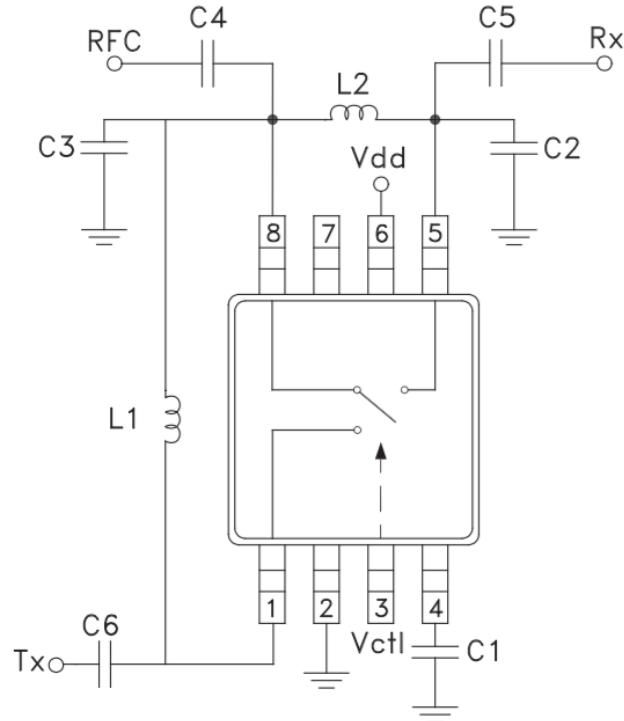


Figure 69. Topologie du circuit d'adaptation du Switch

Les spécifications techniques donnent les valeurs des composants pour différentes bandes de fréquence.

Tuned Frequency	220 MHz	457 MHz
Evaluation PCB Number	110123	110124
C1	150 pF	33 pF
C2	12 pF	6.2 pF
C3	12 pF	7 pF
C4-C6 [1]	1000pF	1000pF
L1	390 nH [2]	100 nH [4]
L2	36 nH [3]	15 nH [3]

[1] DC blocking capacitors

[2] Coilcraft 0603LS series inductor, 5% tolerance

[3] Coilcraft 0402CS series inductor, 5% tolerance

[4] Toko LL1005-FH series inductor, 5% tolerance

Figure 70. Valeurs des composants

Enfin, un dernier composant critique à installer et à adapter est le HPA.

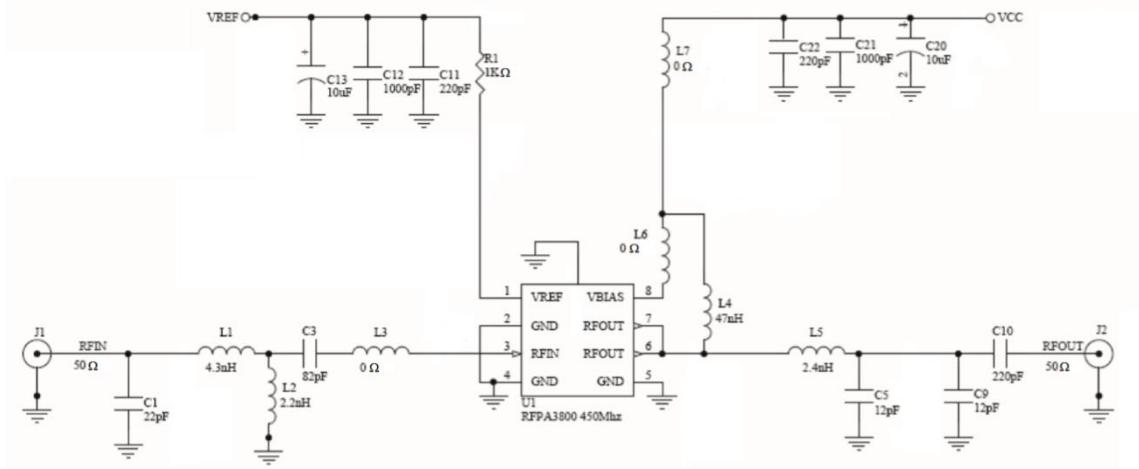


Figure 71. Topologie du circuit d'adaptation du HPA

De même, on a des valeurs proposées par la documentation relative à ce composant.

Description	Reference Designator	Manufacturer	Manufacturer's P/N
PCB		DDI	RFPA3800410(A)
RFPA3800	U1	RFMD	RFPA3800
RES, 1K, 1%, 1/16W, 0603	R1	Panasonic Industrial Co	ERJ-3EKF1001V
CAP, 10uF, 20%, 10V, TANT-A	C13, C20	Kemet	T491A106M010AT
CAP, 22pF, 5%, 50V, COG, 0402	C1		GRM1555C1H220JZ01E
CAP, 12pF, 2%, 50V, HI-Q, 0402	C5, C9	Johanson Technology	500R07S120GV4TD
CAP, 82pF, 5%, 50V, COG, 0402	C3	Murata Electronics North America	GRM1555C1H820JZ01D
CAP, 220pF, 10%, 50V, X7R, 0402	C10-C11, C22	Murata Electronics North America, I	GRM155R71H221KA01E
CAP, 1000pF, 10%, 50V, X7R, 0402	C12, C21	Murata Electronics	GRM155R71H102KA01E
IND, 4.3nH, +/-0.1nH, T/F, 0402	L1	Murata Electronics	LQP15MN4N3B02D
IND, 2.4nH, +/-0.1nH, T/F, 0402	L5	Murata Electronics	LQP15MN2N4B02D
IND, 2.2nH, +/-0.1nH, T/F, 0402	L2	Murata Electronics	LQP15MN2N2B02D
RES, 0Ω, 0402	L3, L6-L7	Kamaya, Inc	RMC1/16SJPTH
IND, 47nH, 5%, W/W, 0603	L4	Coilcraft	0603HC-47NXJLW
CONN, BANANA JACK, RED	P1-P2	JOHNSON CO	108-0902-001
CONN, BANANA JACK, BLACK	P3	JOHNSON CO	108-0903-001
CONN, SMA, ST JACK REC, FLNG MT, T/H	J1-J2	JOHNSON CO	142-0701-631
HEATSINK, POWER CELL	MP1		2020048-2
2-56 SS socket head screws, 3/16 long	S1-S10	McMaster-Carr	92196A076
DNP	C2, C4, C6-C8, C14, C16-C19, C23-C27		
DNP	R2-R5		
DNP	L8		

Figure 72. Valeurs et références des composants du HPA pour la bande 450MHz

Le dernier composant à placer et à adapter est l'oscillateur. Nous avons choisi un TCXO (oscillateur compensé en température) afin de garantir une stabilité optimale de la fréquence et pouvoir atteindre des performances élevées en termes d'émission et de réception :

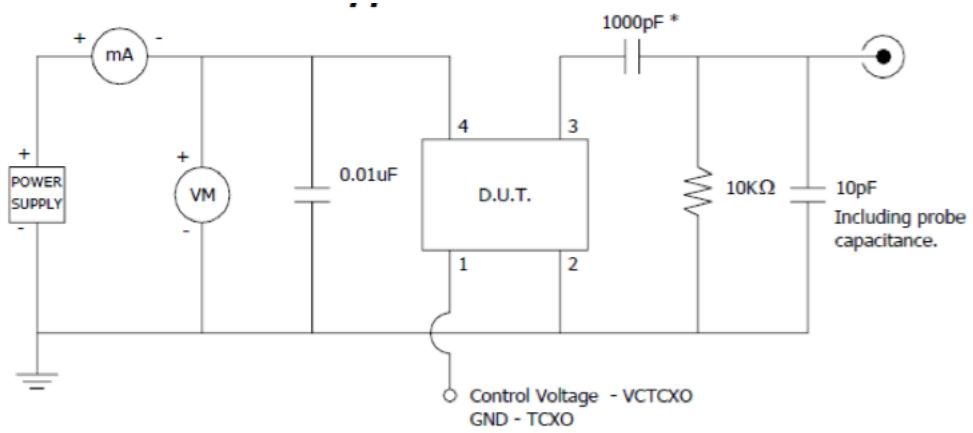
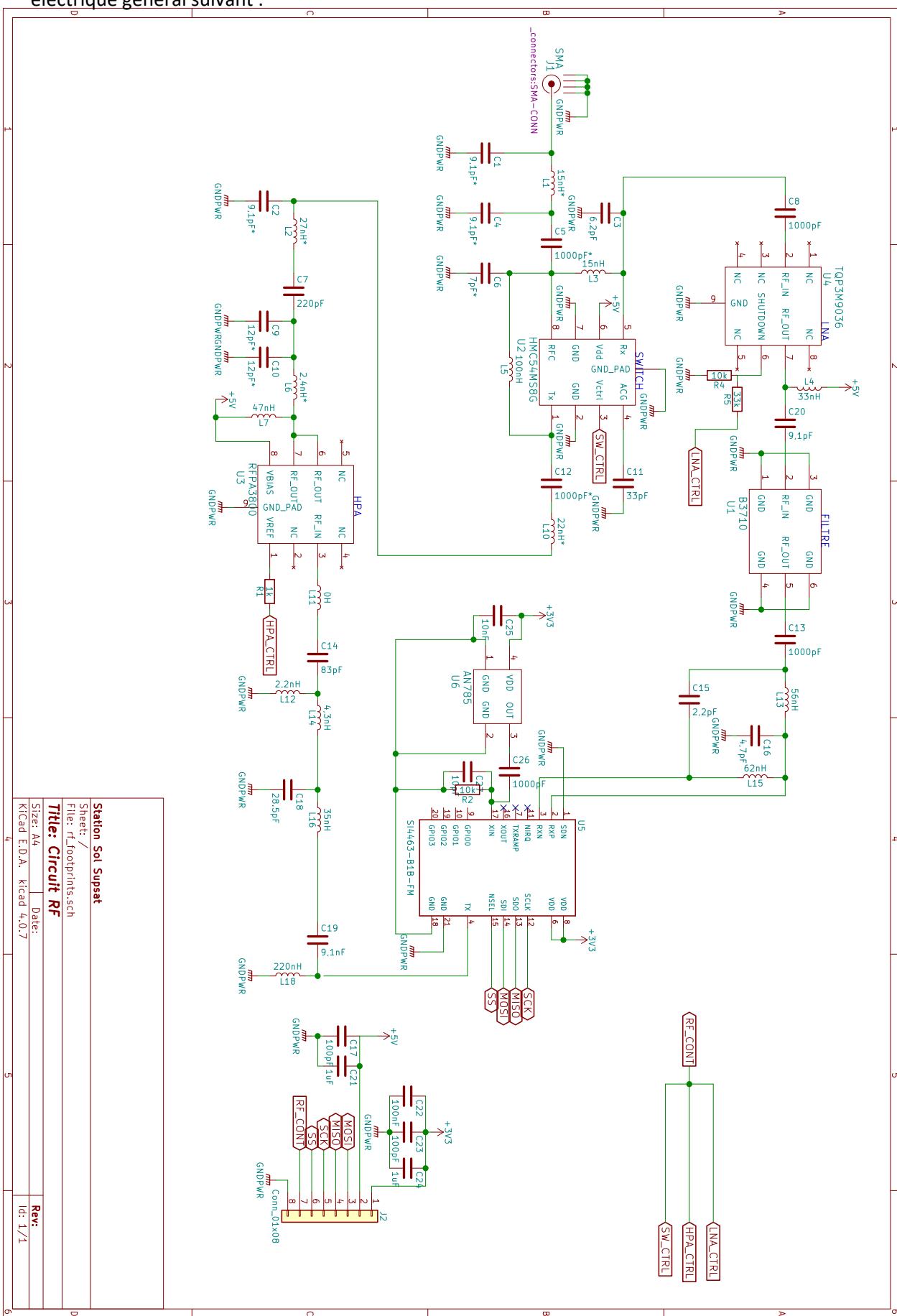


Figure 73. Topologie d'adaptation du TCXO

Au final, en plaçant tous les composants et leur circuit d'adaptation on obtient le schéma électrique général suivant :



Un autre élément extrêmement important de design à prendre en compte pour les circuits radiofréquences est le placement des composants.

En effet, afin d'éviter les couplages, il faut que chaque élément inductif (self) soit orthogonale les uns par rapport aux autres. Il faut que le plan de masse soit le plus large possible et qu'il y est le plus de surface de ce plan entre chaque élément (HPA, LNA, Tranceiver...) et surtout entre les voies RX et TX.

D'autre part, la piste VDD qui alimente le HPA est très élargie car celui-ci consomme beaucoup de puissance.

D'autre part, afin d'avoir un plan de masse plus grand et également de pouvoir dissiper plus efficacement la chaleur, le circuit entier sera percé à l'aide de trous métallisés. Ceux-ci serons plus nombreux dans des zones spécifiques : le long des lignes RX et TX, sous chaque composant consommant de la puissance (LNA, Tranceiver, HPA) et au niveau des capacités en parallèle.

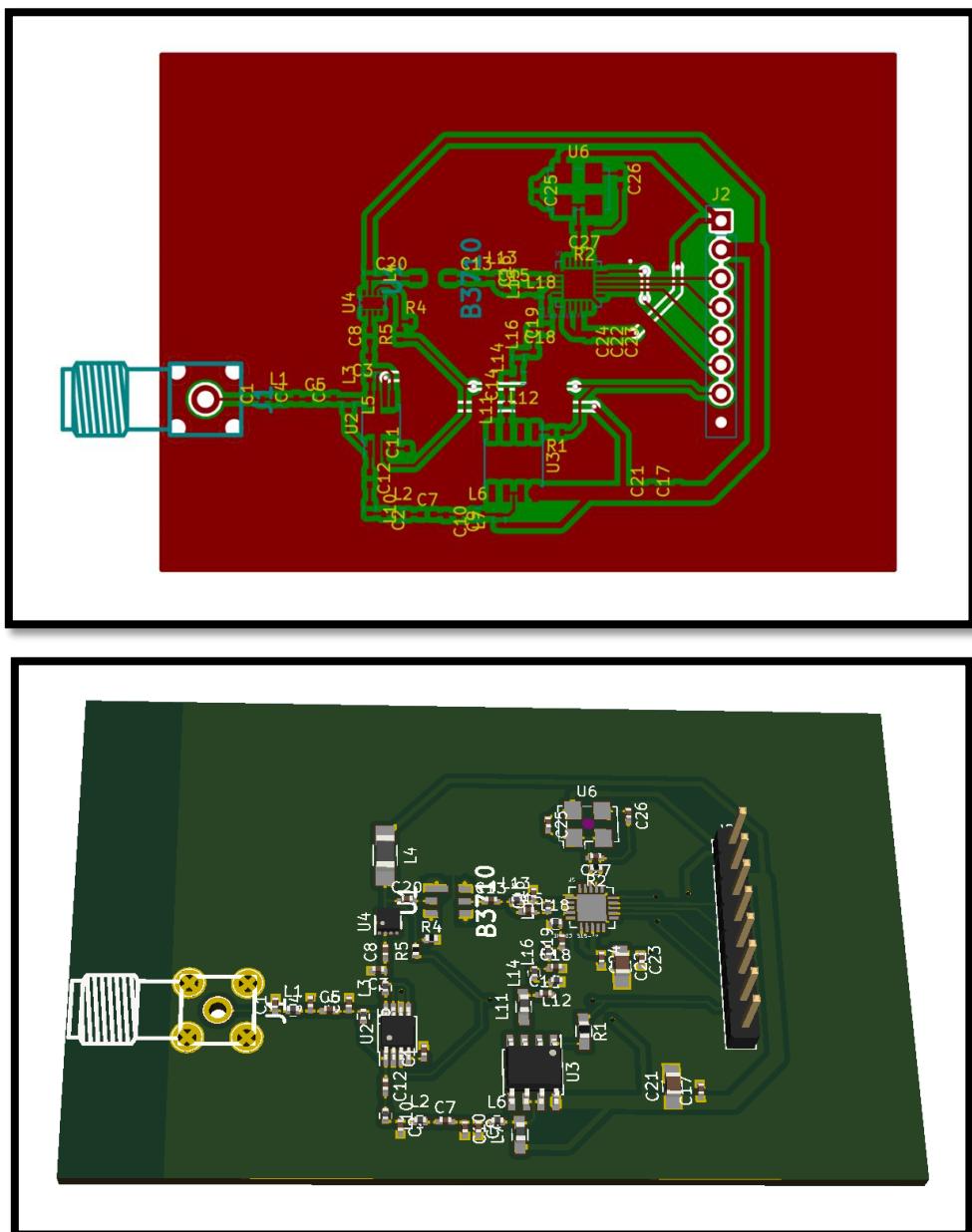


Figure 74. PCB et modèle de notre circuit RF

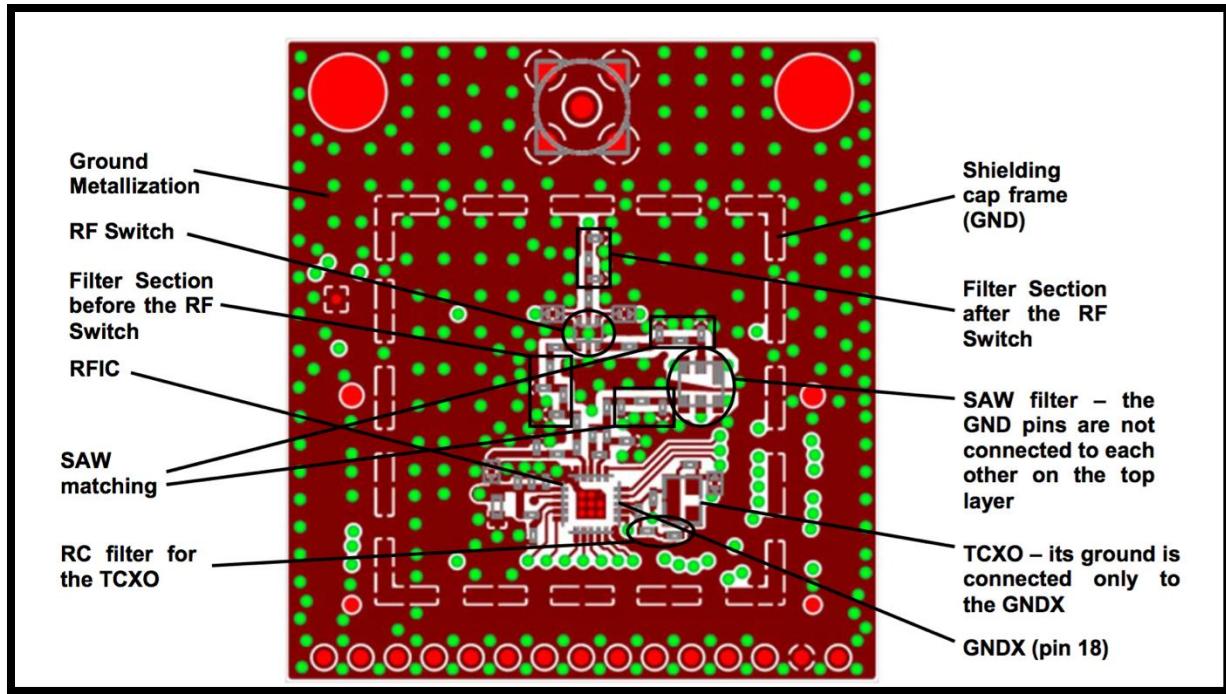


Figure 75. Exemple de PCB présentant les trous métallisés (via holes)

## D. Bilan

La tentative de fixation de l'antenne à la structure motorisée a posé des problèmes. En effet, le poids trop important de l'antenne fait glisser la vis de la roue-vis autour de son axe sur la motorisation d'élévation. Ainsi, il faudra soit mieux serrer les écrous qui tiennent la vis, soit ajouter une vis de serrage radiale sur la vis sans fin et un plat sur l'axe, soit ajouter un contrepoids à l'antenne pour limiter les contraintes mécaniques tout en restant compatible avec la puissance des moteurs pas-à-pas.

Le PCB doit encore être imprimé et les composants soudés.

Le tout devra être testé.

# *Conclusion*

Pour conclure, de nombreux problèmes ont été découverts bien que le projet fut récupéré de l'an passé. Nous avons cependant pu apporter des solutions aux anomalies et lacunes du projet précédent.

Ainsi, le mécanisme de mouvement de la station sol fonctionne correctement, son contrôle se simplifie et gagne en autonomie et une chaîne de réception viable est imaginée.

Nous avons avancé sur d'autres problématiques. En effet, l'antenne est maintenant terminée même si un second prototype, qui nous permettrait de réduire la masse des pièces en mouvement, est en cours de conception. La partie commande de la station est elle aussi terminée et mise en place dans le boitier de protection réalisé sur la station. Il ne reste plus qu'à réaliser le circuit RF, pour lequel nous avons déjà les composants et le schéma du PCB est réalisé. Nous regrettons cependant le fait de ne pas pouvoir tester la station finale mais nous essayerons de continuer le travail cet été pour finir le PCB et assembler la station. Le boitier doit être imprimé et l'implémentation de la communication via l'ESP32 est à faire.

Nous espérons que ce projet sera repris l'an prochain et qu'il pourra être terminé.

Malgré les difficultés rencontrées, ce projet reste fort intéressant par son aspect pratique comprenant une véritable finalité plus que motivante. La multidisciplinarité qu'impose la conception d'une station sol, à savoir concernant l'électronique, la mécanique et la télécom principalement, nous permet d'approfondir nos compétences et méthodes afin de découvrir des solutions viables à des problématiques réelles. L'acquisition de ces connaissances sera un atout certain dans notre future carrière professionnelle.

# *Remerciements*

Nous tenons à remercier les personnes qui nous ont aidé tout au long de ce projet, et en particulier :

- M. Himdi, professeur de Rennes 1, pour ses conseils et son implication dans la partie réception et émission de notre projet. Il nous a grandement aidés à choisir et concevoir l'antenne et la chaîne de réception.
- L'Institut d'Électronique et de Télécommunications de Rennes, pour l'utilisation de son matériel dans la réalisation de l'antenne et du circuit RF.
- Gabriel Sanchez et Philippe Houédé, du service technique de Supélec, pour l'aide sur les définitions des pièces imprimées et la visserie.
- M. Weiss et M. Nafkha, nos encadrants, pour le prêt de matériel et les conseils généraux.

# Annexes

## A. Guide d'utilisation de l'IDF de l'ESP32

Pour installer l'IDF d'Espressif sous Eclipse, il faut suivre l'un des tutoriaux suivants :

- <http://esp-idf.readthedocs.io/en/latest/get-started/>
- [https://exploreembedded.com/wiki/Setting\\_up\\_Eclipse\\_for\\_ESP-IDF](https://exploreembedded.com/wiki/Setting_up_Eclipse_for_ESP-IDF)

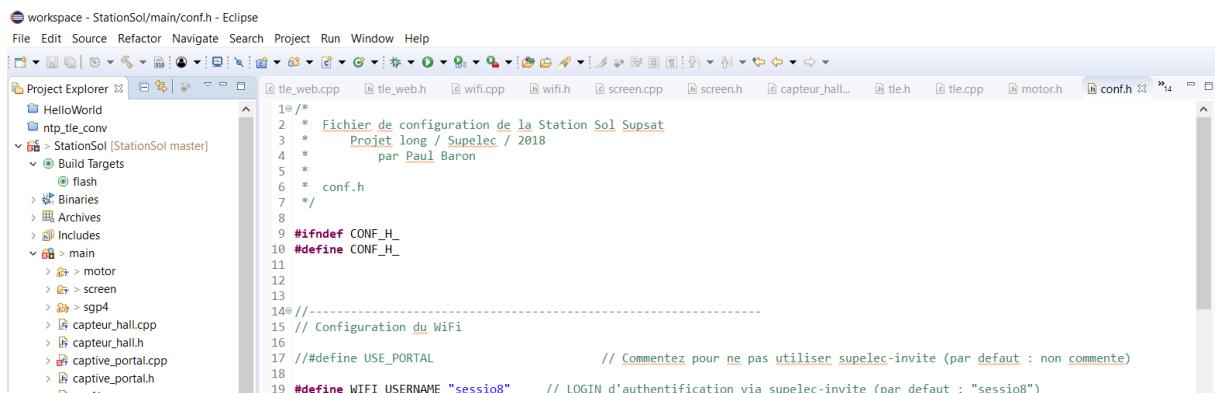
Après cela, vous pouvez importer le projet de Station Sol sous Eclipse.

Attention, ce projet s'appelle *template2*, le renommage du projet peut poser problème.

Si la compilation fonctionne, l'installation est réussie. Sinon, il faudra peut-être vérifier certains paramètres tels que les includes, la configuration avec *make menuconfig* sous *msys32*, etc...

Les différents fichiers de code sont dans le dossier *main*. La configuration se fait essentiellement via le fichier *conf.h* tel que cela a été décrit précédemment dans ce rapport.

Il faut ensuite compiler le programme. Cela peut être long la première fois (15 minutes), mais plus rapide ensuite (1 minute environ). Puis, il faut le téléverser dans l'ESP32 après l'avoir connecté sur le port USB associé via *make menuconfig* (COMx).

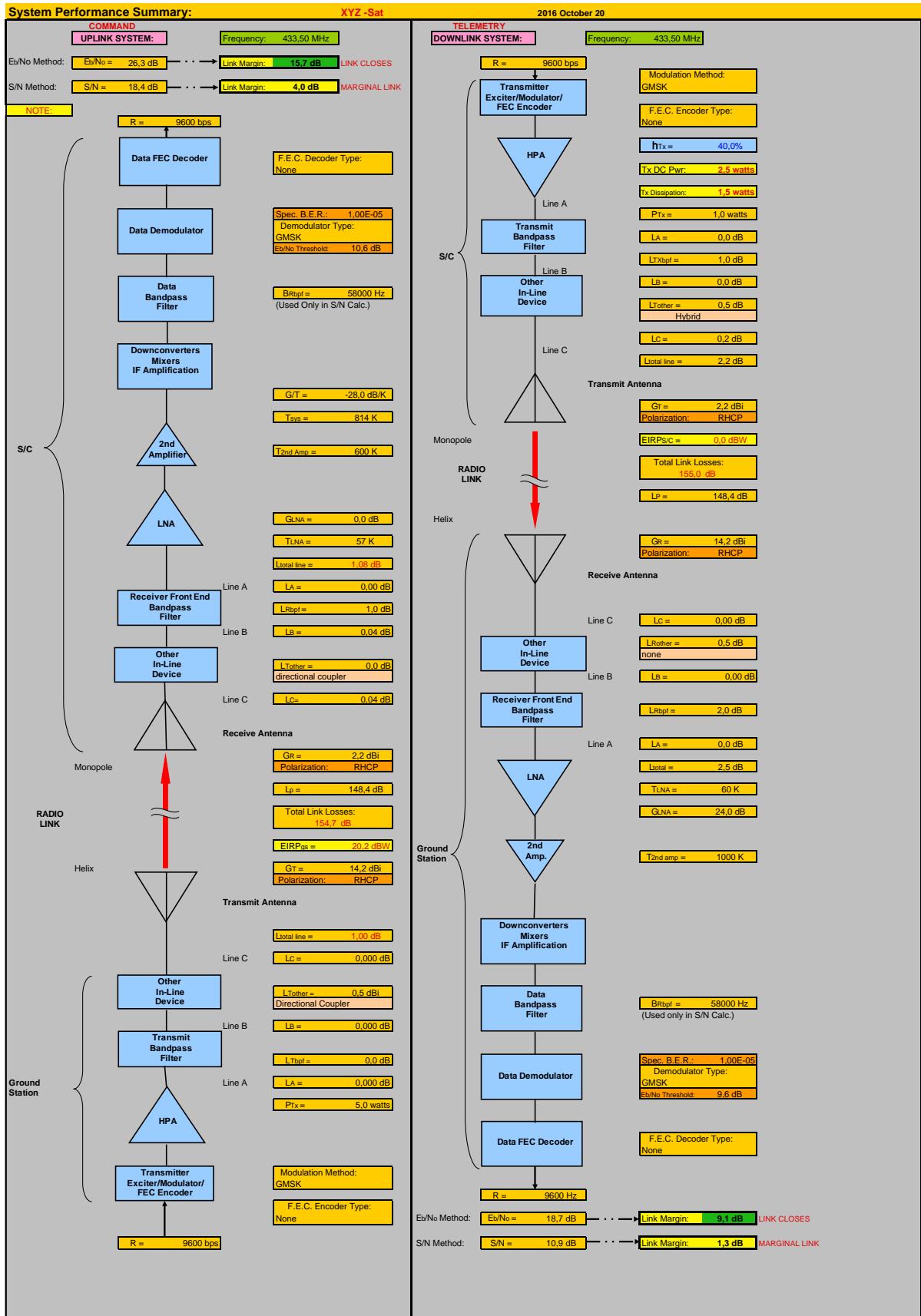


The screenshot shows the Eclipse IDE interface with the 'Project Explorer' view open. The project 'StationSol' is selected, showing its build targets: 'flash' and 'Binaries'. Under the 'main' target, there are several source files: tle\_web.cpp, tle\_web.h, wifi.cpp, wifi.h, screen.cpp, screen.h, capteur\_hall..., tle.h, tle.cpp, motor.h, and conf.h. The 'conf.h' file is currently selected and its contents are displayed in the main editor area:

```
1 /**
2 *   Fichier de configuration de la Station Sol Supsat
3 *   Projet long / Supelec / 2018
4 *   par Paul Baron
5 *
6 *   conf.h
7 */
8
9 #ifndef CONF_H_
10 #define CONF_H_
11
12
13
14 //-----
15 // Configuration du WiFi
16
17 // #define USE_PORTAL           // Commentez pour ne pas utiliser supelec-invite (par defaut : non commente)
18
19 #define WIFI_USERNAME "session0" // LOGIN d'authentification via supelec-invite (par defaut : "session0")
```

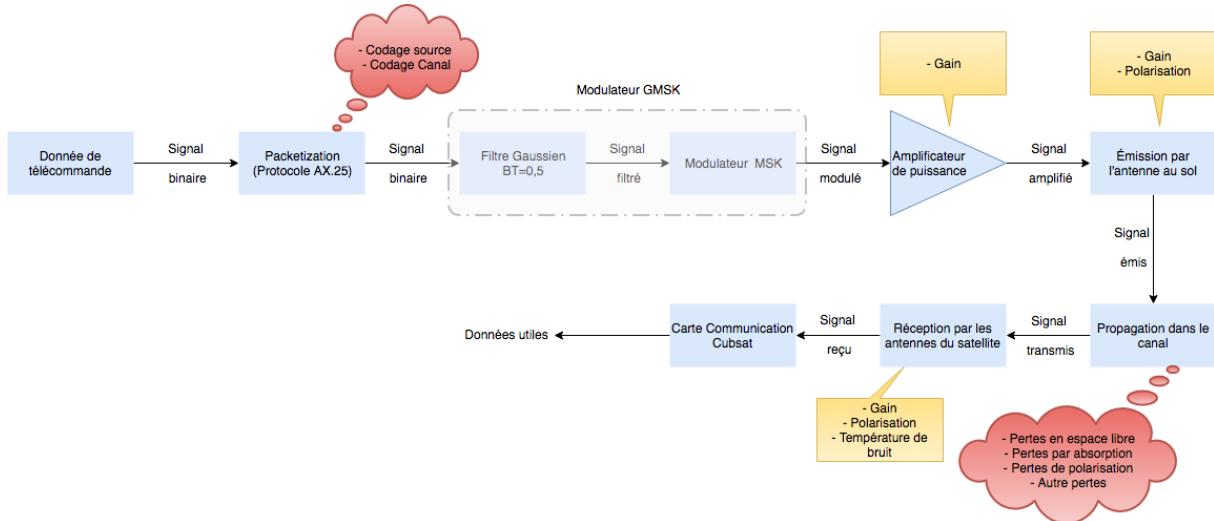
Figure 76. Aperçu du fichier de configuration sous Eclipse

## B. Bilan de liaison avec la feuille de calcul AMSAT-UK

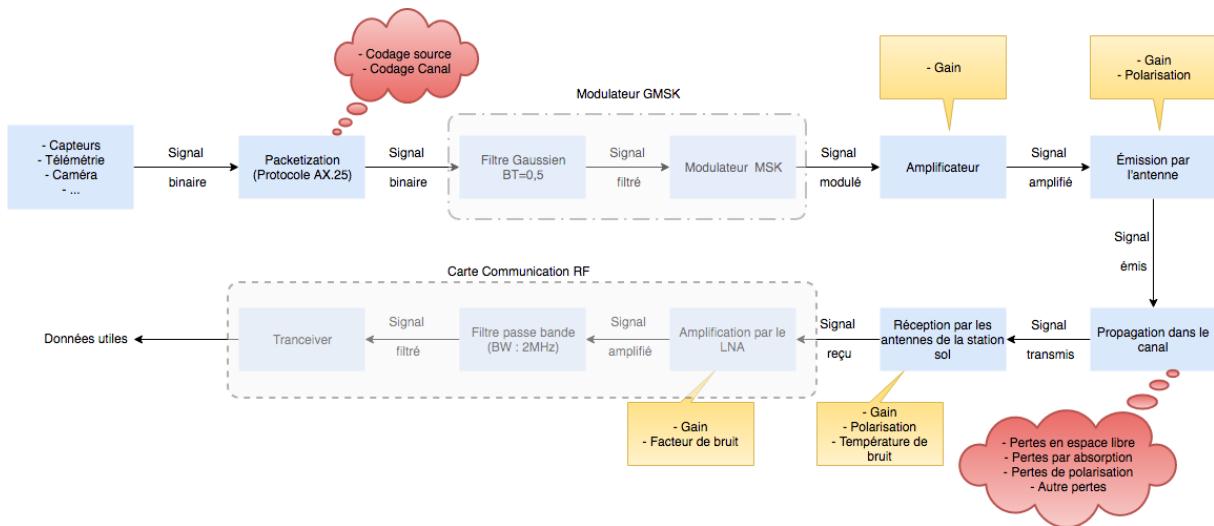


## C. Principe Télécommande et Télémétrie :

### - Lien montant (TC)



### - Lien descendant (TM)



## D. Antenne hélice axiale

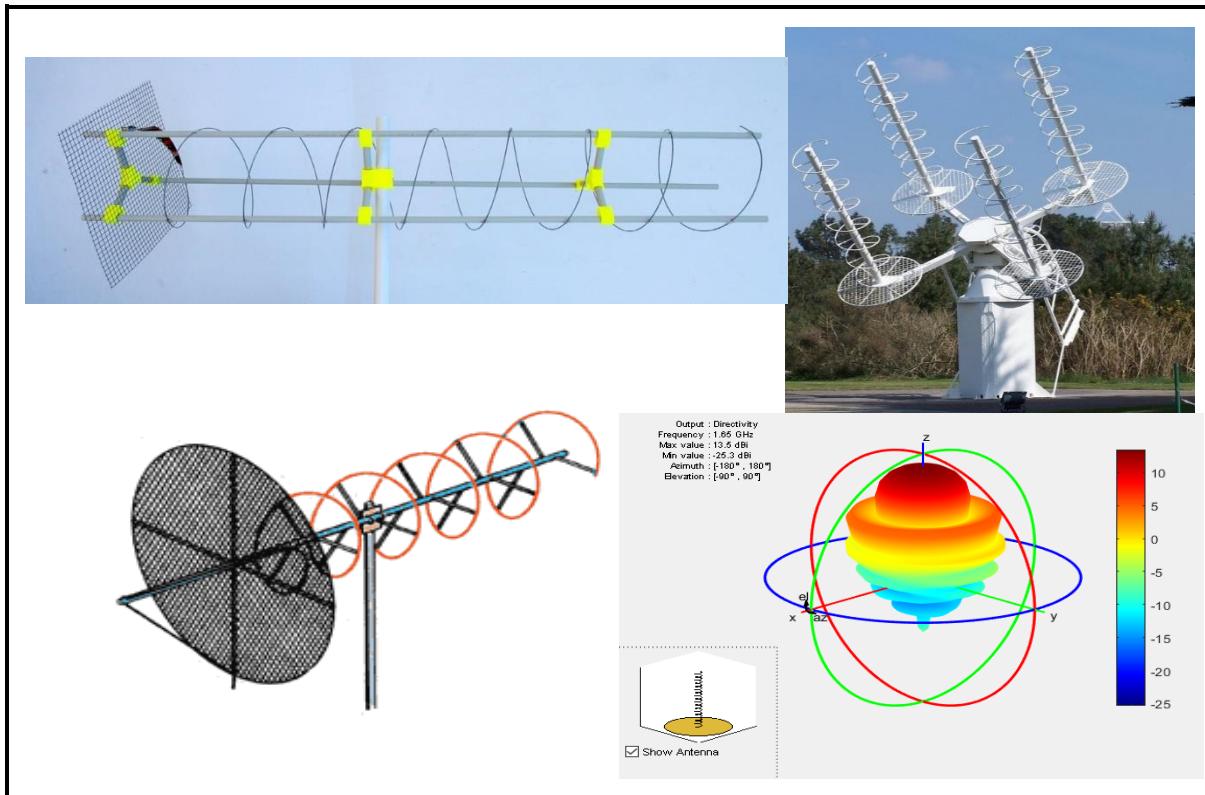


Figure 77. Exemples d'antennes hélices axiales et diagramme de directivité typique

Avantages : facilité de construction ; coût relativement faible ; possibilité de gain assez élevé si assez grande ; bonne directivité

Inconvénients : Polarisation donnée non modifiable ; pas de modèle déjà disponible à Supélec

Grace à la toolbox Antenna de MATLAB, il est très facile de designer des antennes notamment les antennes hélice axial.

Avec la commande `design(type_antenne, fréquence)` MATLAB génère tout seul un modèle d'antenne du type sélectionné, adapté à la fréquence choisie.

```

>> ant=design(helix,433.5e6)

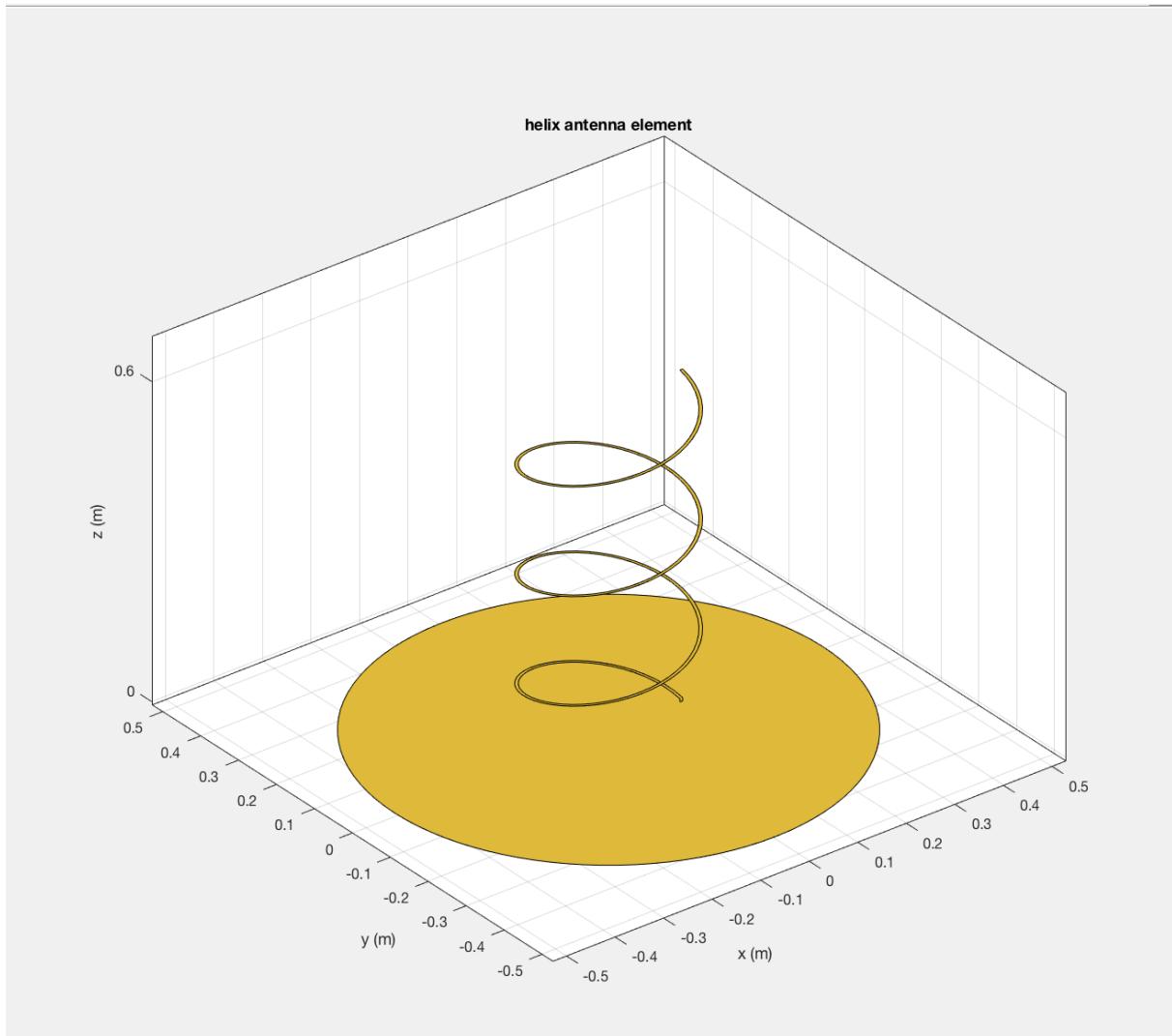
ant =

helix with properties:

    Radius: 0.1467
    Width: 0.0059
    Turns: 3
    Spacing: 0.2054
    WindingDirection: 'CCW'
    GroundPlaneRadius: 0.4401
    Tilt: 0
    TiltAxis: [1 0 0]
    Load: [1x1 lumpedElement]

```

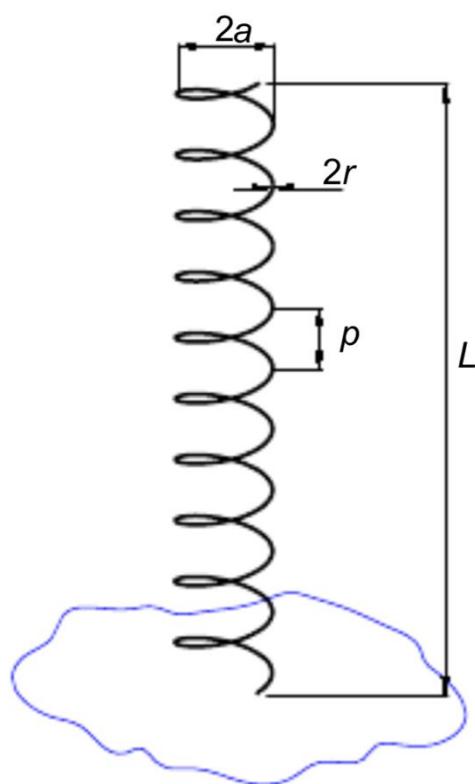
On peut alors afficher cette antenne grâce à la commande `show(ant)` :



L'ensemble des paramètres de l'antenne sont paramétrables afin d'obtenir un design qui nous convient.

- Radius : fait référence au rayon de l'enroulement, celui-ci est en général optimal pour une fréquence lorsque sa valeur est comprise entre  $0,20\lambda$  et  $0,35\lambda$
- Width : fait référence à l'épaisseur du conducteur enroulé, car MATLAB utilise l'épaisseur équivalente d'une ligne à ruban pour la modélisation. Une fonction permet de passer du rayon du conducteur à son épaisseur équivalente, il s'agit de la fonction `cylinder2strip(rayon)`
- Turns : il s'agit du nombre d'enroulement du conducteur (influe sur le gain et la directivité)
- Spacing : espace entre les enroulements, dépend de l'angle de pitch de l'antenne
- WindingDirection : permet de choisir le sens de l'enroulement et ainsi la polarisation de l'antenne (CCW <-> RHCP / CW <-> LHCP)
- GroundPlaneRadius : permet de choisir le rayon du réflecteur.

D'après les recommandations sur l'optimisation des antennes hélice axial émise par l'IEEE, on définit les paramètres suivants :



- Pour avoir un gain maximal à une fréquence donnée, il faut choisir le rayon tel que :
$$\frac{2 \cdot \pi \cdot a}{\frac{c}{f}} = 1,0 - 1,2$$
On choisit donc  $a = \frac{1,15 \cdot \frac{c}{f}}{2 \cdot \pi}$
- Le diamètre du fil conducteur enroulé doit généralement être compris entre  $5 \cdot 10^{-3}\lambda$  et  $5 \cdot 10^{-2}\lambda$  mais sans réel influence sur le gain.  
On peut donc prendre un conducteur de diamètre 1,5mm.

- Pour avoir un gain élevé, il faut augmenter le nombre de tours, on peut s'aider de la formule :  $G = 10 \cdot \log \left( 15 \cdot \left( \frac{2\pi a}{\lambda} \right)^2 \cdot \left( \frac{L}{\lambda} \right) \right)$   
On prévoit entre 15 et 20 enroulements selon la taille maxi que l'on souhaite.
- L'angle de pitch optimal est compris entre  $12^\circ$  et  $14^\circ$  et permet de définir l'espacement entre les tours de fil selon la formule  $pitch = \text{atan} \left( \frac{p}{2\pi a} \right)$   
On choisit pour cet angle  $13^\circ$ , la fonction MATLAB `helixpitch2spacing(pitch, radius)` permet de calculer l'espacement équivalent entre les tours..
- Le diamètre du réflecteur s'il est circulaire doit être environ égale à  $\lambda$ .

```

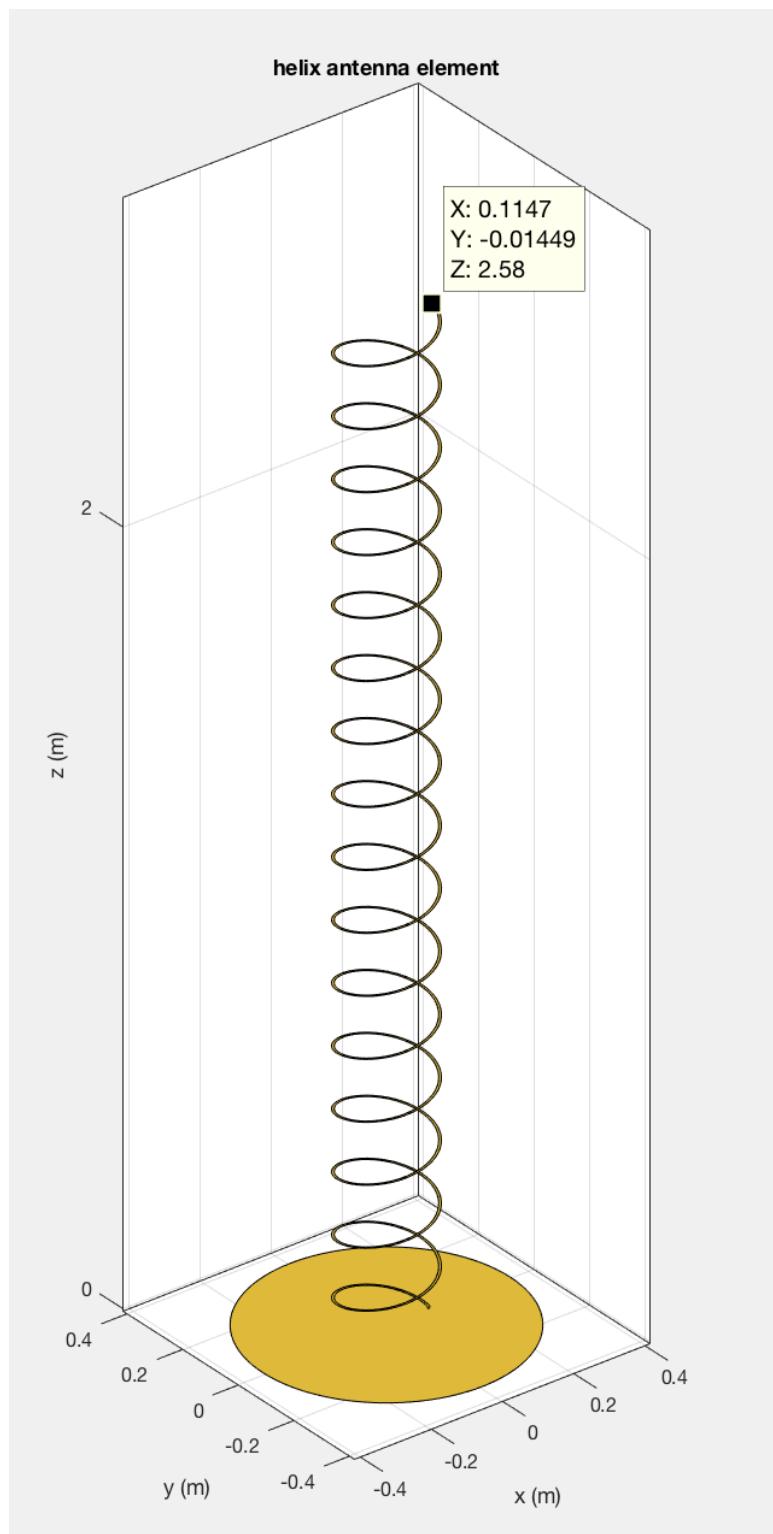
4 - c           = 3e8;
5 - fc          = 433.5e6;
6 - r           = 1.5e-3;
7 - width       = cylinder2strip(r);
8 - radius       = (c/fc*1.05)/(2*pi);
9
10 - turns      = 16;
11 - pitch       = 12.5;
12 - spacing     = helixpitch2spacing(pitch, radius);
13 - side         = c/fc;
14 - radiusGP    = 1*side/2;
15
16 - hx = helix('Radius',radius,'Width',width,'Turns',turns, ...
17 -                 'Spacing',spacing,'GroundPlaneRadius',radiusGP);
18 - figure;
19 - show(hx);
20
21 - figure;
22 - pattern(hx,fc, 'Polarization', 'RHCP');
23
24 - beamwidth(hx,fc,0,1:1:360,1)

```

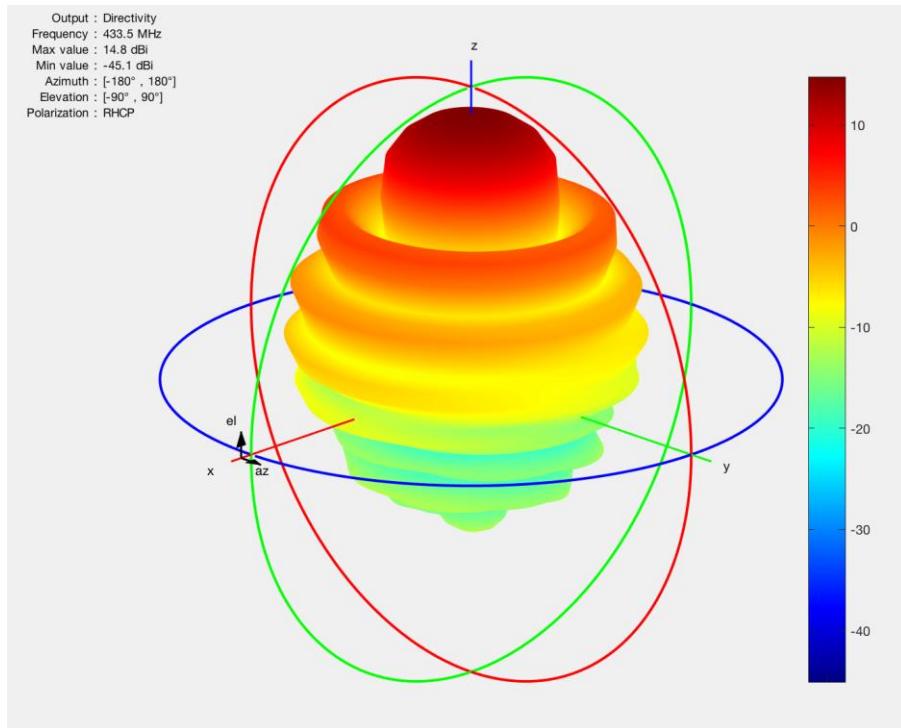
La commande `pattern(hx,fc, 'Polarization', 'RHCP');` permet d'afficher le diagramme de gain de l'antenne en 3D en supposant une polarisation circulaire droite de l'onde (RHCP).

La commande `beamwidth(hx,fc,0,1:1:360,1)` permet d'obtenir le cône de directivité à 1dB.

On obtient ainsi une antenne semblable à :

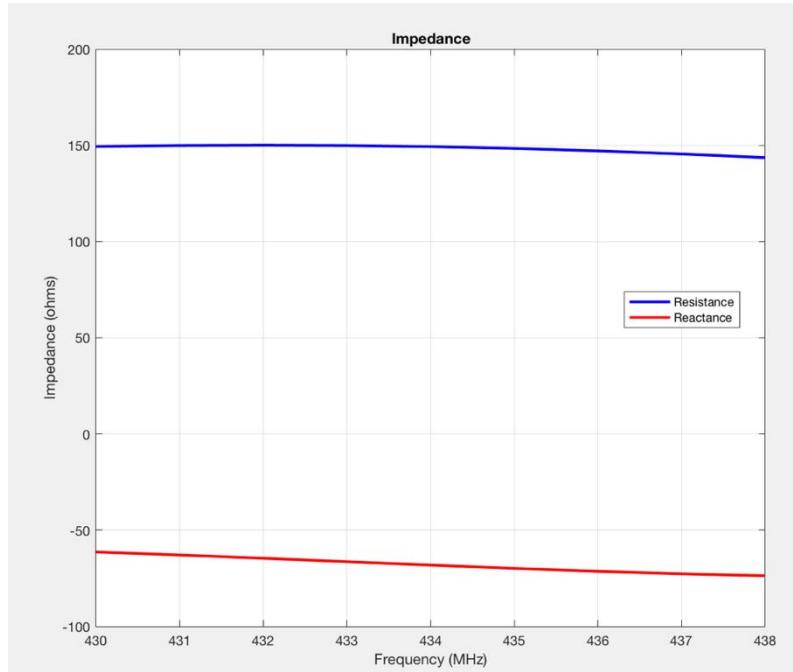


Dont la figure de directivité est :



La commande Matlab `beamwidth(hx,fc,0,1:1:360,1)` nous permet de calculer la directivité de l'antenne, prise ici avec une limite d'atténuation de 1dB, on obtient une directivité de 20°.

On peut également afficher l'impédance calculée par le logiciel en utilisant la fonction `impedance(hx, 430e6:1e6:438e6)` ce qui nous permet d'avoir une idée pour réaliser une adaptation par la suite.

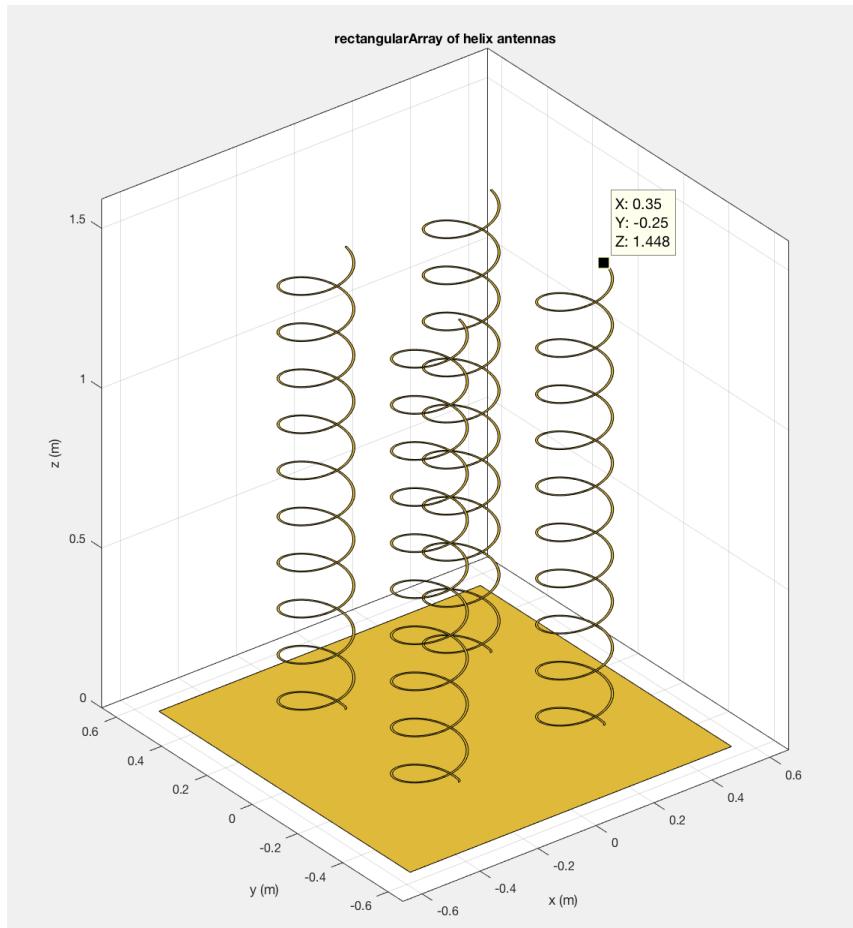


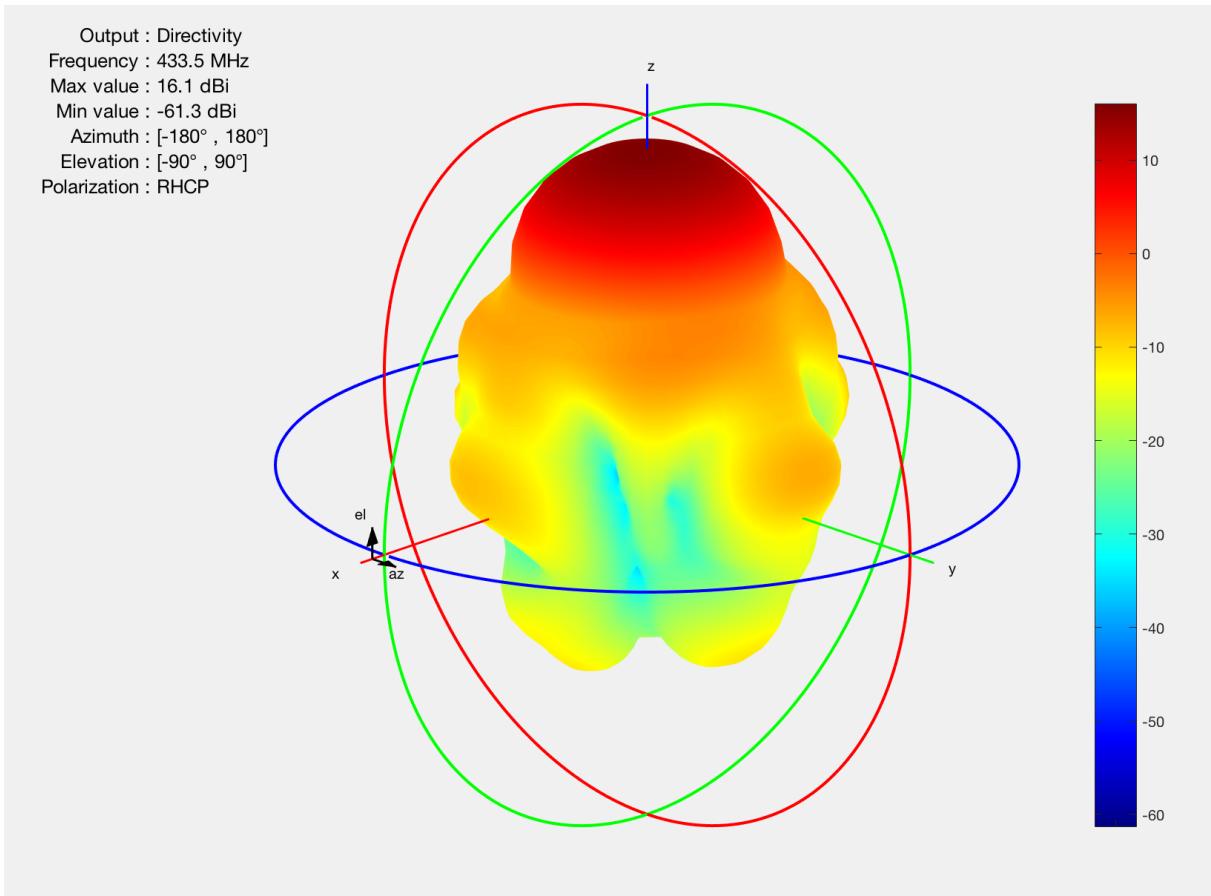
Nous avons par la suite envisagé d'utiliser un réseau d'antenne. Cette solution nous semble plus judicieuse car elle permet de réduire la taille des antennes et ainsi les contraintes mécaniques sur le système de rotation, ainsi que d'augmenter la directivité de l'antenne, et surtout le gain.

Nous avons simulé ce réseau d'antenne grâce à MATLAB :

```

4 -      c          = 3e8;
5 -      fc         = 433.5e6;
6 -      r          = 1.5e-3;
7 -      width       = cylinder2strip(r);
8 -      radius      = 0.1;
9 -      turns       = 10;
10 -     spacing     = (c/fc)/4.8;
11 -     side        = 0.8*c/fc;
12 -     radiusGP    = side/2;
13
14 -     hx = helix('Radius',radius,'Width',width,'Turns',turns, ...
15 -                  'Spacing',spacing,'GroundPlaneRadius',radiusGP);
16
17 -     mra = rectangularArray('Element',hx,'RowSpacing',0.5,'ColumnSpacing',0.5);
18
19 -     figure
20 -     show(mra)
21
22 -     figure
23 -     pattern(mra,fc, 'Polarization', 'RHCP')
24
25 -     beamwidth(mra,fc,0,1:1:360,1)
```





Rq : Le rayon des enroulements est de 10cm ce qui correspond à un rayon standard de tube PVC grâce auquel nous allons réaliser l'antenne.

On obtient bien donc un gain de 16.1dB ce qui est très correct et un cône de directivité à 1dB de 19°.

Ce modèle a été abandonné principalement pour des raisons de contraintes mécaniques importantes liées à la taille de l'antenne et à son poids (les tubes en PVC étant plus lourds que prévu).

## E. Antenne Crossed-Yagi

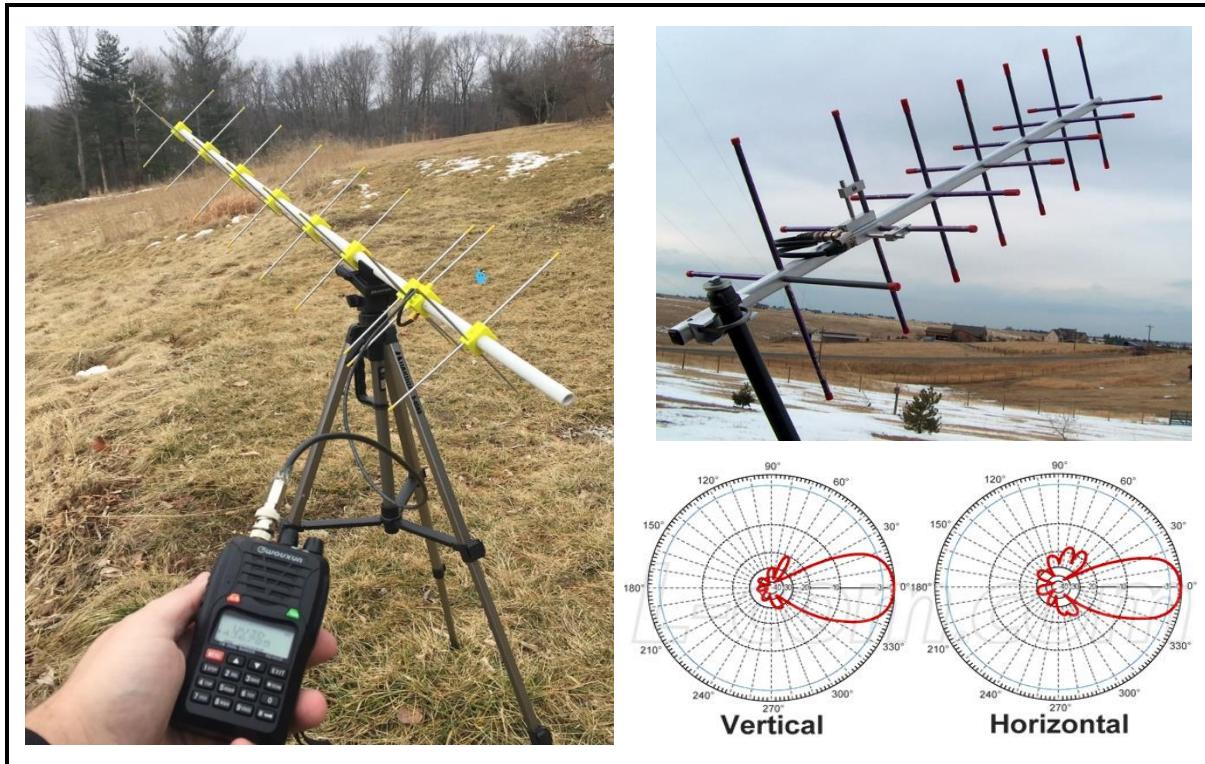


Figure 78. Exemples d'antennes Yagi croisées et diagramme de directivité typique

Avantages : Antenne Yagi linéaire à la bonne fréquence déjà disponible à Supélec ; gain élevé ; possibilité de switcher entre les deux polarisations (RHCP & LHCP) ; bonne directivité

Inconvénients : Fabrication plus technique ; coût légèrement plus élevé

L'antenne hélice axiale est naturellement polarisée circulairement. En ce qui concerne la Crossed-Yagi, il s'agit de deux antennes Yagi différentes disposées à 90°, il faut donc les « coupler » afin d'obtenir la polarisation circulaire souhaitée. Pour cela, il y a deux étapes à effectuer :

- L'adaptation des antennes pour en faire une unique antenne adaptée également

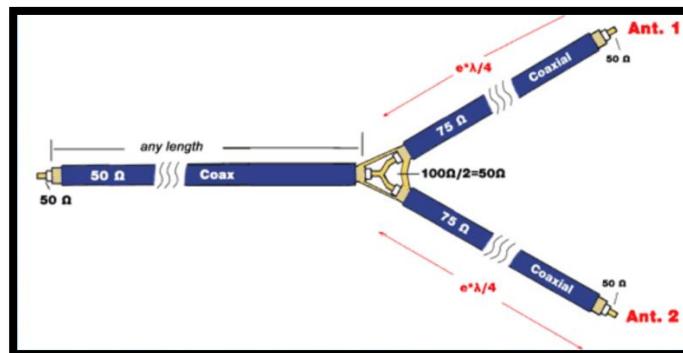


Figure 79. Matching stubs

- Il faut avoir un retard de phase (phase delay) afin de générer une polarisation circulaire  
Pour cela deux méthodes :

- Méthode électrique (à l'aide d'un stub) :

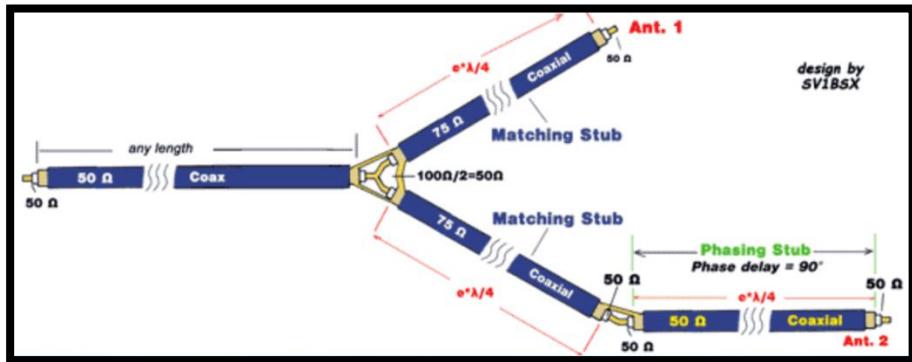


Figure 80. Méthode électrique

- Méthode physique (en espaçant judicieusement les éléments):

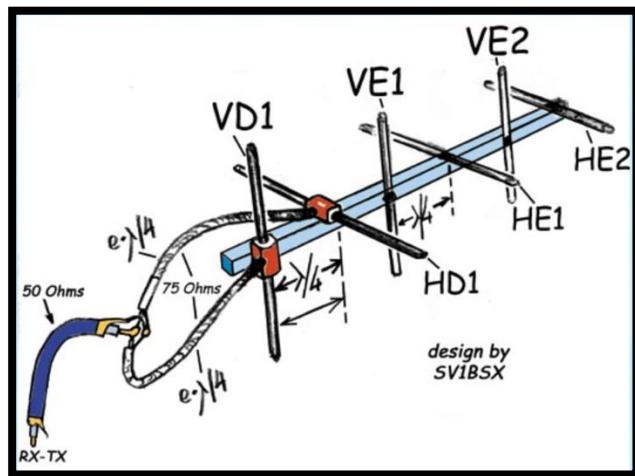


Figure 81. Méthode physique

## F. Modulation

Les deux types de modulation qui peuvent être utilisées sur la carte communication du CubeSat (Texas Instruments **CC1101**) sont les modulations de fréquence et de phase 2-FSK et GMSK.

### 1. La modulation 2-FSK

La modulation FSK (Frequency Shift Keying) est une modulation de fréquence discrète. Les différents symboles sont représentés par différentes fréquences. Dans le cas de la 2-FSK, on utilise deux fréquences pour représenter des bits : la fréquence porteuse plus ou moins une déviation en fréquence.

On appelle  $f_p$  est la fréquence porteuse et  $\Delta f$  la déviation de fréquence, on transmet :

$$\begin{cases} \text{Un «1» lorsque } f(t) = f_p + \Delta f \\ \text{Un «0» lorsque } f(t) = f_p - \Delta f \end{cases}$$

La FSK peut être cohérente ou non-cohérente.

Dans le cas de la FSK non-cohérente, il y a discontinuité dans la phase aux changements de fréquences. La FSK non-cohérente peut se réaliser par exemple, par simple commutation entre deux oscillateurs non synchrones.

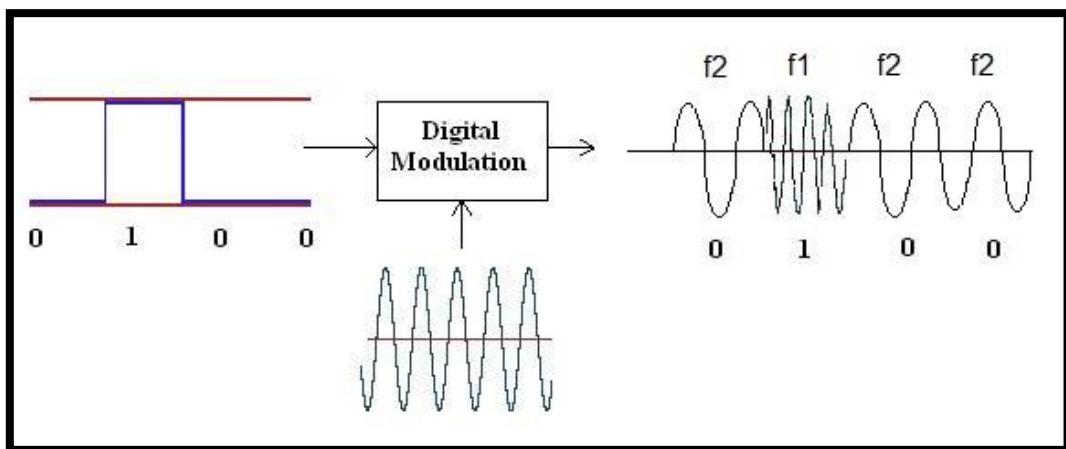


Figure 82. 2-FSK non cohérente

Dans le cas de la FSK cohérente, il n'y a pas de discontinuité de phase lors des changements de fréquences. On réalise ce type de modulation en faisant par exemple, varier la fréquence d'une PLL (boucle à verrouillage de phase). C'est ce type de FSK qui est utilisé par la carte communication.

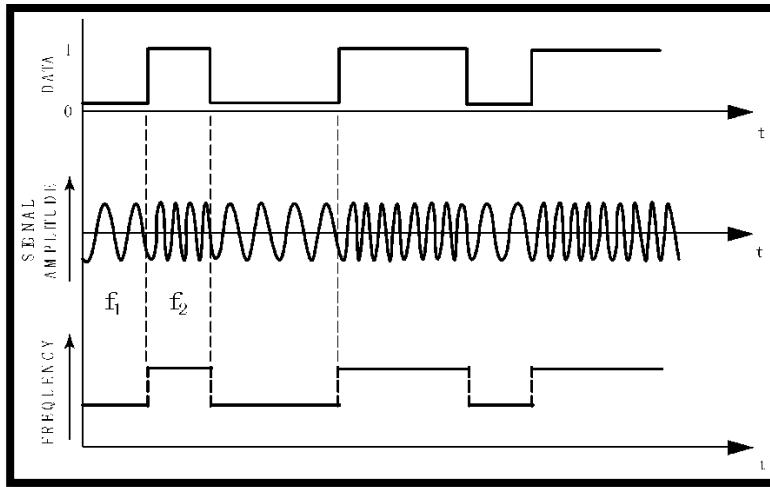


Figure 83. *FSK cohérente*

On définit l'indice de modulation  $m$  tel que :

$$m = \frac{\Delta f}{f_m} = \frac{2 * \Delta f}{D}$$

avec  $D$  le bitrate et  $f_m = D/2$  la fréquence moyenne du signal modulant.

Remarque : Pour la fréquence envisagée ici (433MHz), cela donne une déviation de l'ordre de 10KHz.

L'indice de modulation va influencer l'encombrement spectral.

On peut déterminer la bande passante utile par la règle de Carson :

$$B = 2 * (2 * f_m + \Delta f)$$

Au niveau fréquentiel, les discontinuités de phase de la FSK non-cohérente vont entraîner un élargissement du spectre.

Dans le cas  $m = 0.5$ , la largeur de bande du signal est minimale. On appelle cette modulation MSK (Minimum Shift Keying).

## 2. La modulation GMSK

Un des problèmes de la modulation FSK est sa faible efficacité spectrale, le spectre n'est pas suffisamment étroit, le niveau des lobes secondaires est trop élevé, et des débits binaires élevés induisent systématiquement des largeurs de canaux importantes. Le niveau des lobes secondaires est trop élevé.

La modulation GMSK permet d'améliorer cette efficacité spectrale en réduisant le niveau des lobes secondaires. La GMSK est utilisée dans des systèmes qui nécessitent un débit de données important et plusieurs canaux proches. Le GSM en est l'exemple le plus répandu.

Un filtre gaussien appliqué au train binaire à moduler permet de diminuer l'énergie des lobes secondaires. Ce filtre est caractérisé par la valeur du produit  $B \cdot T_b$  où  $B$  est la fréquence de coupure du filtre et  $T_b$  la durée d'un bit.

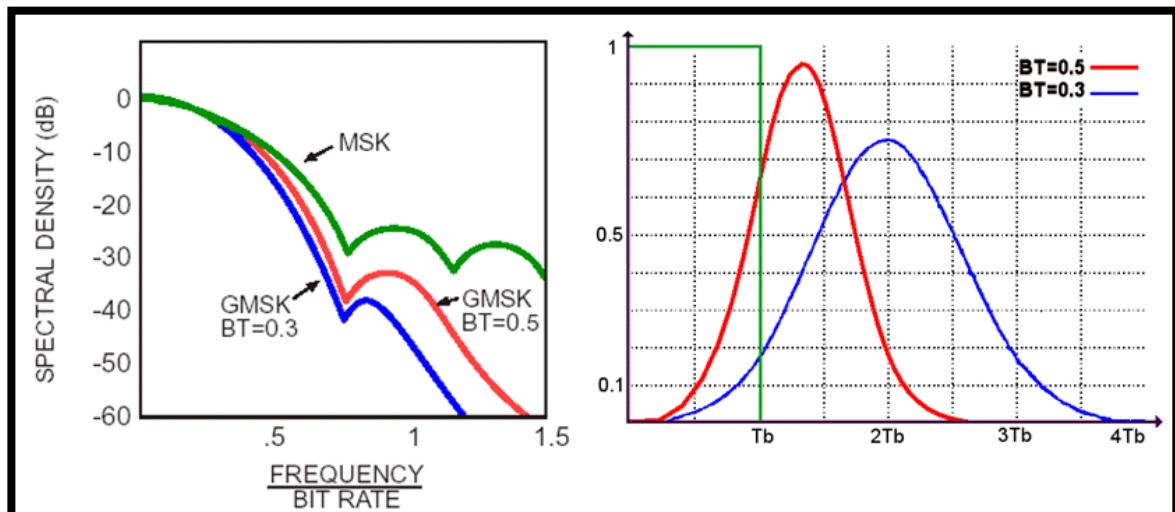


Figure 84. Comparaison des efficacités spectrales et réponse impulsionale du filtre Gaussien

On remarque que la valeur d'un bit influence la valeur du signal filtré sur plus d'une période  $T_b$ . Pour  $B \cdot T_b = 0,5$ , l'influence s'étend sur environs 2 bits. Cette période d'influence donne naissance à un phénomène d'interférence inter-symbole (ISI) qui rend plus difficile la démodulation du signal.

Ainsi, une valeur faible de  $B \cdot T_b$  rend le spectre plus compact mais requiert un meilleur rapport signal/bruit (SNR) pour démoduler les bits. C'est pourquoi généralement on choisit  $B \cdot T_b = 0,5$ .

A la sortie du filtre gaussien, on retrouve un signal NRZ dont les flancs ont été atténus plus ou moins fortement suivant la valeur de BT.

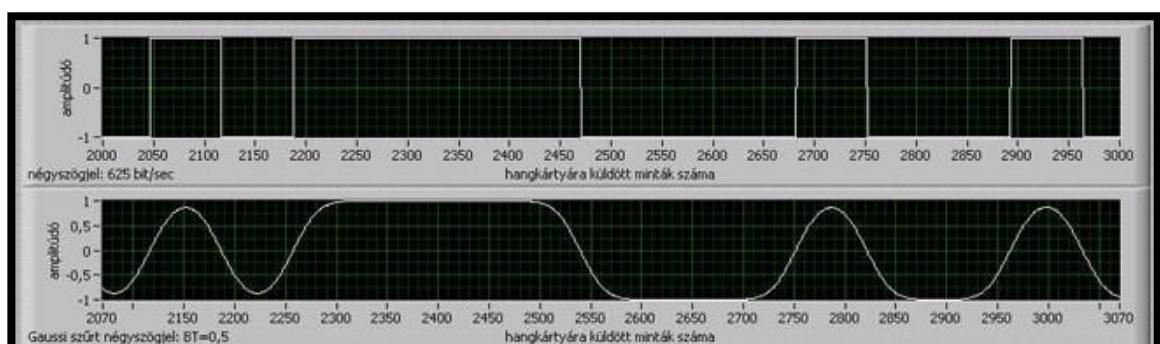


Figure 85. Signal à la sortie du filtre gaussien

### 3. Performance

On évalue la qualité d'une transmission numérique à l'aide du taux d'erreur binaire (BER pour Bit Error Rate). Ce BER est le rapport entre le nombre de bits erronés et le nombre de bits transmis. C'est le protocole utilisé et l'application qui vont imposer le BER minimum à atteindre pour que les données transmises soient exploitables.

Le BER dépend de la modulation et du rapport  $\frac{E_b}{N_0}$ .  $E_b$  est l'énergie par bit à l'entrée du démodulateur et  $N_0$  est la densité de puissance du bruit.

$$E_b = P_s \cdot T_b \text{ et } P_N = N_0 \cdot B \text{ avec } \frac{1}{T_b} = D \text{ le data rate et } B \text{ la largeur de bande du signal.}$$

De plus, la largeur de bande du signal est proportionnelle au data rate :  $B = K \cdot \frac{1}{T_b}$

$$\frac{E_b}{N_0} = \frac{P_s \cdot T_b}{N_0} = \frac{P_s \cdot K}{N_0 \cdot B} = K \cdot \frac{P_s}{P_N}$$

On réutilise le même graphe que précédemment :

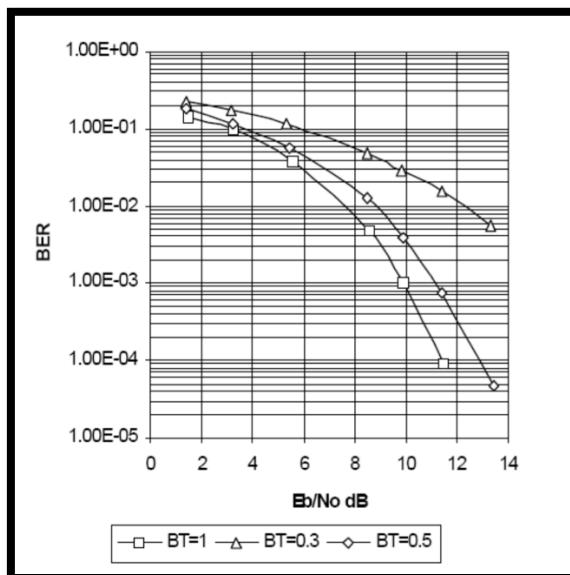


Figure 86.  $BER = f\left(\frac{E_b}{N_0}\right)$  pour un signal GMSK

Il faut trouver un compromis entre efficacité spectrale et rapport signal/bruit...

On peut également exprimer le taux d'erreur par paquets (PER pour Packet Error Rate) :

$$PER = f(L, BER) = 1 - (1 - BER)^L$$

avec L la longueur du message en bytes.

## G. Protocole de communication

Le protocole qui sera utilisé pour la communication avec le CubeSat est le AX.25, protocole radioamateur dérivé du X.25 (utilisé par le minitel), qui est largement utilisé pour les télécommunications avec les Cubesat.

Flag	AX.25 Transfer Frame Header (128 bits)				Information Field	Frame-Check Sequence	Flag
	Destination Address	Source Address	Control Bits	Protocol Identifier			
8	56	56	8	8	32-2048	16	8

Figure 87. Trame du protocole AX.25

Il est également envisageable d'utiliser le protocole NGHam, protocole lui-même basé sur le AX.25 et modifié afin d'être plus robuste aux bruits et avoir une meilleure efficacité spectrale ainsi qu'un haut débit utile.

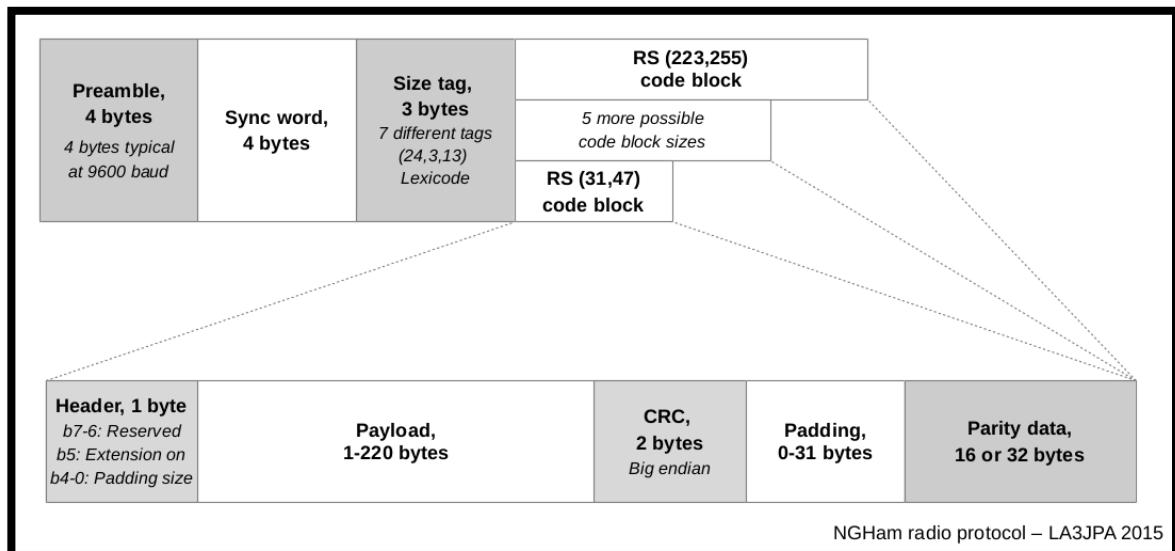


Figure 88. Trame du protocole NGHam

Nous souhaitons également implémenter un **codage source** ainsi qu'un **codage canal** (code cyclique ou convolutif) afin de rendre la communication plus efficace. Pour cela, nous sommes actuellement en train de nous concerter avec le groupe qui travaille au développement de la carte communication du cubesat.

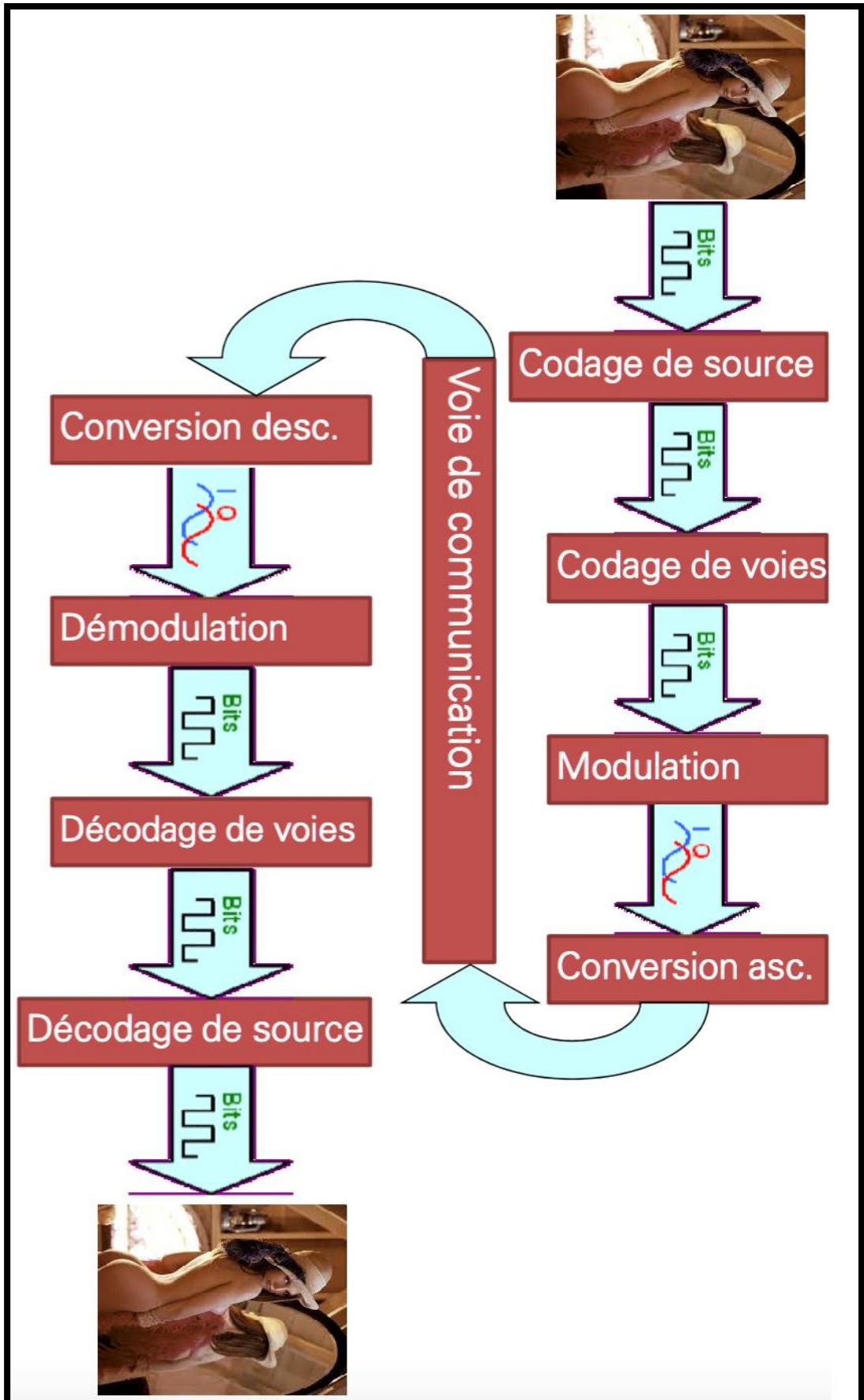


Figure 89. Système de communication numérique

# Sources

Ces sources peuvent s'avérer utiles pour toute continuation du projet :

<http://www.qsl.net/sv1bsx/antenna-pol/polarization.html>

<http://nuts.cubesat.no/om>

<https://satnogs.org>

<https://www rtl-sdr.com/>

<https://www.ettus.com/>

<https://wiki.gnuradio.org/i>

<https://github.com/zacmanchester/kicksat/wiki/Setting-Up-A-Ground-Station>

<http://www.space.aau.dk/cubesat/>

<https://fr.mathworks.com/>

<http://swisscube.epfl.ch/>

<http://www.n2yo.com/satellites/?c=18>

<http://cubestar.no>

<http://igosat.in2p3.fr/fr/page-d-exemple/>

[https://services.renater.fr/ntp/serveurs\\_francais](https://services.renater.fr/ntp/serveurs_francais)

<http://celesttrak.com/NORAD/>

<http://exploreembedded.com/wiki/>

<https://esp-idf.readthedocs.io/>

<https://github.com/nkolban/esp32-snippets/>

<https://github.com/AnalogLamb/ESP32-ApplicationNote/>

<https://github.com/espressif/esp-idf/>

<http://www.zeptomoby.com/satellites/>

<https://www.n2yo.com/>