



Université Sidi Mohamed Ben Abdellah
Faculté des Sciences Dhar El Mahraz - Fès
Centre d'Etudes Doctorales
"Sciences et Technologies"



Formation Doctorale : Sciences et Technologies de l'Information et de la Communication (STIC)

Laboratoire : Laboratoire d'Intégration des Systèmes et des Technologies Avancées (LISTA)

Spécialité : Génie électrique

THESE DE DOCTORAT

Présentée par :

HANAFI Ahmed

Sous la Direction du :

Pr. KARIM Mohammed

Sous le titre :

ETUDE ET CONCEPTION DU SEGMENT SPATIAL DU NANOSATELLITE UNIVERSITAIRE MASAT1 AVEC ORDINATEUR DE BORD SECONDAIRE RECONFIGURABLE A BASE DE FPGA

Soutenue le 07 / 10 / 2017 devant le jury composé de :

Pr. EL BEKKALI Moulhime	Université Sidi Mohamed Ben Abdellah - Fès	Président
Pr. EL MHAMDI Jamal	ENSET - Rabat	Rapporteur
Pr. TISSIR El houssaine	Université Sidi Mohamed Ben Abdellah - Fès	Rapporteur
Pr. DEROUICH Aziz	Ecole Supérieure de Technologie - Fès	Rapporteur
Pr. RACHIDI Taje-eddine	Université Al Akhawayn - Ifrane	Examineur
Pr. MOUDDEN Zakaria	Centre Royal d'Etude et de Recherche Spatiale - Rabat	Examineur
Pr. AKRAM Rafiq	Centre Royal d'Etude et de Recherche Spatiale - Rabat	Invité
Pr. KARIM Mohammed	Université Sidi Mohamed Ben Abdellah - Fès	Directeur de thèse

DEDICACES

À ma chère mère, mon frère Amine et ma sœur Hind
ainsi qu'à ma chère épouse Bouchra et mes enfants Saad et Nada,
ma plus profonde reconnaissance
pour votre soutien, vos encouragements, votre amour et votre patience.

REMERCIEMENTS

Je tiens à remercier vivement et fortement tous ceux qui ont contribué de près ou de loin à la concrétisation de ce modeste travail.

Tout d'abord, je tiens à témoigner ma profonde et sincère reconnaissance à mon Professeur et Directeur de recherche, *Pr. Karim Mohammed*:

- Pour la grande confiance qu'il m'a accordée et sa patience durant toute la durée de mes études doctorales.
- Pour ses directives sages et clairvoyantes qui m'ont permis de mener à bien ce modeste travail.

J'exprime toute ma gratitude au Pr. EL BEKKALI Moulhime pour m'avoir fait l'honneur de présider le jury qui va évaluer ce modeste travail.

J'exprime également toute ma gratitude et mes remerciements au Pr. EL MHAMDI Jamal, au Pr. TISSIR El houssaine et au Pr. DEROUICH Aziz de m'avoir honoré en acceptant la charge d'être rapporteurs de cette thèse et en me faisant part de leurs remarques constructives.

De même j'adresse mes remerciements au Pr. RACHIDI Taje-eddine, au Pr. MOUDDEN Zakaria et à Mr. AKRAM Rafiq de m'avoir honoré en acceptant d'être membres du jury afin d'examiner ce modeste travail.

J'associe également à ce moment très particulier mon professeur, ami et collègue Pr. *El Hassan El Mimouni*, compagnon de route, qui a initié ma relation scientifique et humaine avec le Pr. Karim Mohammed, et avec qui je partage les mêmes valeurs humaines, ainsi que la passion et l'amour de l'électronique.

RESUME

Ces dernières années, les nanosatellites ou Cubesats connaissent une croissance considérable vu qu'ils permettent de réaliser, à faible coût et dans un délai réduit, une large gamme de missions spatiales (industrielles ou scientifiques) en orbite terrestre basse. Dans ce contexte, ce travail de thèse a pour objectif de concevoir le segment spatial du premier nanosatellite universitaire marocain MASAT1, avec comme charge utile un ordinateur de bord secondaire reconfigurable en orbite. Ce travail de recherche a aussi pour ambition de positionner les universitaires marocains dans un créneau à forte valeur ajoutée, à savoir les technologies spatiales.

Ainsi, nous avons, dans un premier temps, réalisé une analyse de la mission comprenant une étude orbitale et une évaluation des contraintes spatiales et des contraintes de lancement. Ce qui nous a permis de déduire les contraintes de conception des différents sous-systèmes du segment spatial du nanosatellite MASAT1.

Dans un deuxième temps, nous avons réalisé l'étude et la conception des différents sous-systèmes du segment spatial, en tenant compte des contraintes liées aux dimensions, poids, consommation d'énergie, contraintes spatiales et contraintes de lancement. Nous avons opté pour une architecture distribuée utilisant le bus I2C comme bus de communication. Cette approche s'est avérée la meilleure possible car elle offre une modularité au niveau de la conception matérielle et logicielle, ainsi qu'une testabilité indépendante des différents sous-systèmes.

La contribution innovante de notre travail de recherche réside dans la charge utile qui consiste à utiliser le nanosatellite comme une plateforme permettant de tester la fiabilité d'un ordinateur de bord reconfigurable en orbite. L'architecture matérielle et logicielle de ce dernier a été conçue selon une approche System-on-Chip en utilisant le soft processeur Microblaze et les fonctionnalités de reconfiguration du FPGA (Field Programmable Gate Array) Xilinx Spartan-6 à mémoire SRAM (Static Random Access Memory).

Nous avons pu, à travers cette thèse, acquérir la technologie et les compétences de base nécessaires pour l'ingénierie spatiale, en plus de proposer une architecture temps réel embarquée à faible consommation et flexible au niveau de la reconfiguration en orbite de ses fonctionnalités matérielles et logicielles.

Mots clés : Nanosatellite, cubesat, reconfiguration, FPGA SRAM, ordinateur de bord, Microblaze.

ABSTRACT

In recent years, nanosatellites and cubesats market have grown considerably, making it possible to carry out, at low cost and in a short time, a wide range of spacecraft missions (industrial or scientific) in Low Earth Orbit. In this context, this thesis aims at designing the spatial segment of the first Moroccan university nanosatellite MASAT1, with as a payload an On-Board secondary computer reconfigurable in orbit. This research also aims to position Moroccan academics in a segment with high added value, namely space technologies.

Thus, we first carried out a mission analysis including an orbital study and an assessment of spatial and launch constraints. This allowed us to deduce the design constraints of the MASAT1 space segment subsystems.

Secondly, we studied and designed the different subsystems of the space segment, taking into account constraints related to dimensions, weight, energy consumption, spatial constraints and launch constraints. We opted for a distributed architecture using the I2C bus as a serial communication bus. This approach proved to be the best possible because it offers modularity in hardware and software design, as well as independent testability of the various subsystems.

The innovative contribution of our research work lies in the payload by using the nanosatellite as a platform to test the reliability of a reconfigurable on-board computer. The hardware and software architecture was designed using a System-on-Chip approach based on Microblaze soft processor and the Xilinx SRAM-based FPGA Spartan-6.

Through this thesis, we were able to acquire the technology and the basic skills required for space engineering, in addition to proposing a low-power embedded real-time architecture that is flexible in terms of in-orbit reconfiguration of its hardware and software capabilities.

Keywords: Nanosatellite, cubesat, reconfiguration, FPGA SRAM, On-Board Computer, Microblaze

TABLE DES MATIERES

DEDICACES	<i>i</i>
REMERCIEMENTS	<i>ii</i>
RESUME	<i>iii</i>
ABSTRACT	<i>iv</i>
INTRODUCTION GENERALE	1
1. CONTEXTE DE LA RECHERCHE	1
2. ETAT DE L'ART DES NANOSATELLITES	2
2.1. Classification des petits satellites.....	2
2.2. Description système d'un nanosatellite	3
3. LE CONCEPT CUBESAT	5
3.1. Configurations	5
3.2. Technologies.....	7
3.3. Systèmes de déploiement	7
4. HISTORIQUE DES MISSIONS CUBESAT	9
5. PROJET DU NANOSATELLITE UNIVERSITAIRE MASAT1	12
6. MOTIVATION ET OBJECTIFS DE LA RECHERCHE	13
7. PLAN DE LA THESE	15
CHAP. 1 : ANALYSE DE LA MISSION DU NANOSATELLITE MASAT1	16
1. PHASE DE DEVELOPPEMENT D'UNE MISSION SPATIALE	17
2. APPLICATION DES STANDARDS ECSS SUR LA MISSION	18
2.1. Architecture et applicabilité du système ECSS.....	18
2.2. Gestion du projet à l'aide du standard ECSS-M	20
2.3. Détermination des exigences de la mission à l'aide du standard ECSS-E	24
3. ETUDE ORBITALE	27
3.1. Etat de l'art de la dynamique orbitale.....	27
3.1.1. Référentiels utilisés	27
3.1.2. Le temps	29
3.1.3. Les lois de Kepler	29
3.1.4. Problème de deux corps.....	30
3.1.5. Equation de la trajectoire (1ère loi de Kepler)	31
3.1.6. Détermination de l'orbite à l'aide des paramètres orbitaux.....	31
3.2. L'orbite du nanosatellite MASAT1.....	34
3.3. Perturbations orbitales	36
3.3.1. Perturbations géopotential dues à l'aplatissement de la terre	37
3.3.2. Perturbations dues à la trainée atmosphérique	38
3.3.3. Perturbations du 3ème corps.....	40
3.3.4. Evaluation des perturbations orbitales avec MATLAB	41
3.4. Couverture de la terre.....	45
3.5. Temps de communication	48

3.6. Durée d'éclipse	53
4. ETUDE ET EVALUATION DES CONTRAINTES SPATIALES.....	56
4.1. Environnement et sources radiatives.....	56
4.1.1. Ceintures de radiations de Van Allen	56
4.1.2. Les radiations solaires	57
4.1.3. Rayons cosmiques galactiques (GCR).....	57
4.2. Les effets des radiations sur les circuits électroniques	58
4.2.1. Dose cumulée	58
4.2.2. Dose de Dommage de Déplacement (DDD)	59
4.2.3. Effets des évènements singuliers (SEE)	60
4.3. Dégazage	62
5. EVALUATION DES CONTRAINTES DE LANCEMENT.....	63
5.1. Charges quasi-statiques (CQS)	63
5.2. Vibrations sinusoïdales.....	64
5.3. Spectre acoustique	65
5.4. Les chocs mécaniques	65
5.5. Pressions statiques	66
5.6. Facteurs de sécurité	67
6. CONTRAINTES THERMIQUES	67
7. CONCLUSION	69
CHAP. 2 : ARCHITECTURE DE LA PLATEFORME MASAT1	70
1. INTRODUCTION	71
2. DECOMPOSITION FONCTIONNELLE DE MASAT1	72
3. DEFINITION DES SOLUTIONS DE CONCEPTION DE MASAT1	73
3.2. Architecture du segment spatial	73
3.3. Bus de communication.....	75
3.4. Structure et configuration.....	79
3.5. Communication	83
3.6. L'approche des composants COTS pour le choix des solutions technologiques.....	85
3.7. Concept des opérations de MASAT1.....	86
4. DESCRIPTION DE LA PLATEFORME MASAT1	89
4.2. Diagramme bloc matériel et électrique	89
4.3. Diagrammes Bloc des flux de données.....	91
CHAP. 3 : ETUDE ET DESCRIPTION DU SOUS-SYSTEME ENERGIE DE BORD	93
1. ETAT DE L'ART.....	94
1.1. Fonctionnalités d'un sous-système EdB.....	94
1.2. Les cellules photovoltaïques	94
1.3. Fonction MPPT de recherche du point de puissance maximale	97
2. APPROCHES D'IMPLEMENTATION ET ARCHITECTURE.....	99
3. DESCRIPTION DU SOUS-SYSTEME EDB DE LA MISSION MASAT1	101
3.1. Exigences du sous-système EdB.....	101
3.2. Interfaces électriques.....	102
4. CHOIX DE LA SOLUTION TECHNOLOGIQUE.....	103
4.1. Cellules photovoltaïques	103

4.2. Module Energie de Bord.....	107
4.3. Batteries	111
5. BILAN DE PUISSANCE.....	112
5.1. Contraintes et hypothèses	112
5.2. Puissance générée par les cellules solaires.....	113
5.3. Puissance consommée en mode nominal.....	115
5.4. Scénario de consommation.....	116
 CHAP. 4 : ETUDE ET DESCRIPTION DU SOUS-SYSTEME COMMUNICATION 118	
1. INTRODUCTION	119
2. EXIGENCES DU SOUS-SYSTEME COM POUR LA MISSION MASAT1 ...	120
3. INTERFACES ELECTRIQUES.....	121
4. ARCHITECTURE DU SOUS-SYSTEME COM DE LA MISSION MASAT1.	121
5. BILAN DE LIAISON.....	122
5.1. Contraintes orbitales.....	123
5.2. Spécifications du module de transmission du segment spatial	123
5.2.1. Débit de transmission.....	124
5.2.2. Technique de modulation	124
5.2.3. Consommation d'énergie	125
5.2.4. Fréquence de fonctionnement.....	125
5.3. Modélisation du canal de transmission.....	125
5.4. Paramètres des antennes.....	127
5.5. Pertes de propagation radiofréquence dans l'espace	128
5.6. Bruit thermique	131
5.7. Equations utilisées dans le bilan de liaison	133
5.8. Bilan de la liaison montante (station sol vers segment spatial).....	135
5.9. Bilan de la liaison descendante (Segment spatial vers station sol).....	136
5.10. Interprétation des résultats	136
6. CHOIX DE LA SOLUTION TECHNOLOGIQUE.....	137
 CHAP. 5 : ETUDE ET DESCRIPTION DES SOUS-SYSTEMES ORDINATEUR DE BORD, STRUCTURE ET CONTROLE D'ATTITUDE 140	
1. ETUDE ET DESCRIPTION DU SOUS-SYSTEME ORDINATEUR DE BORD	141
1.1. Exigences du sous-système OdB pour la mission MASAT1	141
1.2. Spécifications matérielles d'un OdB.....	144
1.3. Architecture des OdB utilisés dans d'autres missions Cubesat	148
1.4. Interfaces électriques.....	149
1.5. Architecture du sous-système OdB de la mission MASAT1	150
1.6. Choix de la solution technologique	151
1.6.1. Carte mère CubeSat Kit de Pumpkin avec processeur MSP430F2618 :.....	151
1.6.2. Module NanoMind A712D de GOMSpace.....	152
1.6.3. Module NanoMind A3200 de GOMSpace	153
2. DESCRIPTION DE LA CARTE MERE NANODOCK DMC-3 DE GOMSPACE	156
2.1. Présentation	156
2.2. Fonctionnalités matérielles	157

3. ETUDE ET DESCRIPTION DU SOUS-SYSTEME STRUCTURE (STR).....	161
3.1. Fonctionnalité de la structure d'un Cubesat	161
3.2. Exigences du sous-système STR pour la mission MASAT1	162
3.3. Choix de la solution technologique	163
3.4. Analyse de la structure.....	165
3.4.1. Bilan de masse.....	165
3.4.2. Centre de gravité et centre d'inertie.....	166
4. DESCRIPTION DU SOUS-SYSTEME CONTROLE D'ATTITUDE (ATT)....	167
4.1. Présentation du contrôle d'attitude passif	167
4.2. Etapes pour la conception du sous-système ATT.....	168
4.2.1. Détermination des couples perturbateurs agissant sur le CubeSat.....	168
4.2.2. Choix et dimensionnement des aimants permanents	169
4.2.3. Choix et dimensionnement des barres hystérésis	171
CHAP. 6 : DESCRIPTION DE L'ORGANISATION LOGICIELLE.....	174
1. VUE D'ENSEMBLE.....	175
2. LES MODES OPERATIONNELS	175
2.1. Description	175
2.2. Transitions entre les modes opérationnels.....	176
3. ALGORITHMES PRELIMINAIRES DES MODES FONCTIONNELS	178
3.1. Mode Séparation.....	178
3.2. Mode Initialisation	179
3.3. Mode Sans Echec.....	180
3.4. Mode Nominal.....	180
4. LA REFERENCE TEMPORELLE	181
5. ARCHITECTURE DE COMMUNICATION (PROTOCOLES CSP)	181
5.1. Topologie.....	181
5.2. Architecture logicielle.....	183
6. PROTOCOLES POUR LA COMMUNICATION RADIO	185
6.1. Protocole AX25.....	185
6.2. Protocole PUS (Packet Utilisation Standard).....	186
7. SYSTEME D'EXPLOITATION TEMPS REEL FREERTOS	188
7.1. Noyau FreeRTOS.....	190
7.2. Mécanisme de gestion des tâches	191
7.3. Gestion de la mémoire	192
8. IDENTIFICATION DES FONCTIONNALITES LOGICIELLES DE L'ODB..	193
9. ORGANISATION LOGICIELLE EN COUCHES ET MODULES	194
CHAP. 7 : ETUDE ET CONCEPTION DE L'ODB SECONDAIRE RECONFIGURABLE	
A BASE DE FPGA.....	196
1. DESCRIPTION DE LA CHARGE UTILE DE LA MISSION MASAT1	197
1.1. Motivation.....	197
1.2. Exigences de l'ODB secondaire de la mission MASAT1	198
2. ETAT DE L'ART DES ARCHITECTURES RECONFIGURABLES.....	200
2.1. Introduction aux FPGAs.....	200

2.2.	Technologies des FPGAs	201
2.3.	Utilisation des FPGAs dans les missions spatiales	203
2.4.	Types de défaillances dans les architectures reconfigurables à base de FPGA	204
2.5.	Moyens de prévention et d'atténuation des effets des radiations.....	206
2.5.1.	Approches de préventions	206
2.5.2.	Approches d'atténuation.....	207
2.6.	Fonctionnalités de reconfiguration des FPGA type SRAM	210
2.6.1.	Activité du FPGA lors de la reconfiguration	210
2.6.2.	Quantité reconfigurable des ressources du FPGA.....	210
2.6.3.	Format du fichier "Bitstream" et mécanisme de téléchargement.....	211
2.6.4.	Interface et mode de téléchargement du fichier "bitstream"	212
3.	ARCHITECTURE DES SYSTEMES RECONFIGURABLES	
	DYNAMIQUEMENT.....	213
3.1.	Description	213
3.2.	Processeurs reconfigurables	214
3.3.	Moteurs de reconfiguration	214
4.	CONCEPTION MATERIELLE DE L'OdB SECONDAIRE.....	216
4.1.	Architecture adoptée	216
4.2.	Choix du FPGA SPARTAN 6 de XILINX.....	217
4.2.1.	Les arguments	217
4.2.2.	Le processeur SoftCore MicroBlaze.....	219
4.2.3.	Logique de contrôle de configuration du FPGA Spartan-6.....	220
4.2.4.	Reconfiguration dynamique du FPGA Spartan-6.....	221
4.3.	Choix des mémoires	222
4.4.	Conception du bloc d'alimentation	226
4.5.	Réalisation de la carte OdB secondaire de la mission MASAT1	228
5.	CONCEPTION LOGICIELLE DE L'OdB SECONDAIRE	230
5.1.	Outils de développement.....	230
5.2.	Les bibliothèques logicielles.....	232
5.2.1.	Le système d'exploitation temps réel XilKernel	232
5.2.2.	La bibliothèque LibXil ISF.....	233
5.3.	Implémentation de l'auto reconfiguration dynamique de l'OdB secondaire	234
5.3.1.	Mode et interface externe de configuration.....	234
5.3.2.	Implémentation des fonctionnalités Multiboot/fallback.....	235
5.3.3.	Implémentation de la duplication matérielle avec comparaison	237
5.3.4.	Architecture matérielle implémentée dans le FPGA Spartan-6	238
5.3.5.	Test de l'injection et la récupération d'erreur	240
5.3.6.	Test des fonctionnalités Multiboot/fallback	244
5.4.	Implémentation de la mise à jour à distance de l'OdB secondaire.....	248
5.4.1.	Préparation des fichiers d'images mis à jour	249
5.4.2.	Téléchargement du fichier image dans la mémoire système SDRAM	249
5.4.3.	Programmation de la mémoire Flash SPI	251
5.4.4.	Déclenchement à distance de la reconfiguration dynamique du FPGA.....	253
5.4.5.	Mise en œuvre, test et résultats	253
6.	CONCLUSIONS ET PERSPECTIVES.....	255
	CONCLUSION GENERALE.....	256

ANNEXES	259
ANNEXE A	260
Tableau des données CIRA72 d'intervalle d'altitude pour la densité atmosphérique [23] ...	260
ANNEXE B	261
Code source Matlab permettant l'évaluation des perturbations orbitales	261
ANNEXE C	268
Benchmark fonctionnel des OdB de différentes missions Cubesat	268
ANNEXE D	270
Code Matlab pour les caractéristiques I-V et P-V des panneaux photovoltaïques	270
REFERENCES	272
Bibliographie	273
Webographie	279

LISTE DES FIGURES

Figure 1 : Vues des segments spatiaux des missions Sputnik I, Explorer I et Vanguard 1 [5].....	3
Figure 2 : Vue globale du système englobant segment spatial et segment au sol	4
Figure 3 : Sous-systèmes du segment spatial.....	4
Figure 4 : Spécifications de conception d'un CubeSat Format 1U [10].....	6
Figure 5 : Système de déploiement P-POD [10].....	8
Figure 6 : Système de déploiement KiboCUBE [139]	9
Figure 7 : Etat des nanosatellites lancés à la date du 08/01/2017 [140].....	10
Figure 8 : Types d'organisations qui participent aux lancements des nanosatellites [140]	10
Figure 9 : Classement des nanosatellites lancés par type de configuration [140].....	11
Figure 10 : Répartition géographique des développeurs de nanosatellites [140].....	11
Figure 11 : Les structures universitaires participantes au projet.....	12
Figure 1-1 : Les phases de développement d'une mission spatiale	17
Figure 1-2 : Normes d'utilisation ECSS structurées en Branches [18].....	18
Figure 1-3 : Disciplines du système des normes ECSS [18]	19
Figure 1-4 : Chaîne Client-Fournisseur du projet MASAT1	20
Figure 1-5 : Tâches à effectuées et Planning de la mission.....	21
Figure 1-6 : Arborescence "Produit" du projet MASAT1.....	22
Figure 1-7 : Structure de répartition du travail (WBS)	23
Figure 1-8 : Système de coordonnées du référentiel géocentrique [25].....	28
Figure 1-9 : Système de coordonnées du référentiel périodique [25]	28
Figure 1-10 : (a) 2 masses situées dans un référentiel galiléen (b) Corps libre [24].....	30
Figure 1-11 : Paramètres de l'orbite képlérienne en coordonnées cartésiennes [24]	31
Figure 1-12 : Les éléments orbitaux dans le référentiel géocentrique [24].....	32
Figure 1-13 : Emprunte du nanosatellite MASAT1 au sol	35
Figure 1-14 : Orbite inertielle 3D du nanosatellite MASAT1.....	35
Figure 1-15 : Les accélérations perturbatrices en fonction de l'altitude [26].....	36
Figure 1-16 : Les vecteurs positions concernant la lune comme 3 ^{ème} corps.....	40
Figure 1-17 : Éléments orbitaux sous la perturbation J2 [MATLAB]	42
Figure 1-18 : Orbite inertielle 3D du nanosatellite MASAT1.....	42
Figure 1-19 : Éléments orbitaux sous l'effet de la traînée [MATLAB]	43
Figure 1-20 : Variation du périhélie et de l'apogée sous l'effet de la traînée [MATLAB].....	44
Figure 1-21 : Durée de vie orbitale de MASAT1 pour une altitude de 650 km [STK].....	44
Figure 1-22 : Durée de vie orbitale de MASAT1 pour une altitude de 400 km [STK].....	44
Figure 1-23 : Éléments orbitaux sous l'effet lunisolaire [MATLAB].....	45
Figure 1-24 : Représentation de la zone du champ de vision (Access Area) [8]	46
Figure 1-25 : Relations angulaires entre satellite, cible et centre de la Terre [8].....	46
Figure 1-26 : Simulation de la couverture de la terre [STK]	48
Figure 1-27 : Géométrie d'une période de visibilité [14].....	49
Figure 1-28 : Orbite instantanée d'un satellite	50
Figure 1-29 : Représentation géométrique de l'amplitude en azimut et la trace au sol	51
Figure 1-30 : Timeline pour une altitude de 400 km et une inclinaison de 90° [STK].....	53
Figure 1-31 : Timeline pour une altitude de 650 km et une inclinaison de 40° [STK].....	53
Figure 1-32 : Position de la Terre dans la sphère céleste centrée autour du satellite [14]	54
Figure 1-33 : Géométrie de l'éclipse sur la sphère céleste centrée autour du satellite [14]	55
Figure 1-34 : Les ceintures de radiations de Van Allen [Internet].....	56
Figure 1-35 : Ejection de masse coronale [142]	57
Figure 1-36 : Eruption solaire [143].....	57
Figure 1-37 : Densité et énergie des particules dans l'environnement spatial [37]	58
Figure 1-38 : La dose ionisante totale pour les différents scénarios [SPENVIS].....	59
Figure 1-39 : Les doses de dommage de déplacement [SPENVIS]	60
Figure 1-40 : Mécanismes des ions lourds et des protons causant les effets SEU [40]	61

Figure 1-41 : L'occurrence des évènements uniques [SPENVIS]	62
Figure 1-42 : Vibrations longitudinales sinusoïdales des différents lanceurs [41-47]	64
Figure 1-43 : Vibrations latérales sinusoïdales des différents lanceurs [41-47]	65
Figure 1-44 : Spectre de réponse aux chocs pour les différents lanceurs [41-47].....	66
Figure 1-45 : Pression statique des différents lanceurs [41-47]	66
Figure 1-46 : Flux aérothermique typique après le largage du carénage [36].....	68
Figure 2-1 : Interrelations entre les processus de conception du système	71
Figure 2-2 : Diagramme bloc fonctionnel de la mission MASAT1	72
Figure 2-3 : Diagramme N ² de la mission MASAT1	73
Figure 2-4 : Architecture centralisée (à droite) et distribuée (à gauche).....	74
Figure 2-5 : les bus de communication série utilisés dans les missions Cubesat.....	76
Figure 2-6 : Dispositif de connexion standard pour le bus I2C.....	76
Figure 2-7 : Dispositif de connexion pour le bus CAN [ISO11898 High Speed].....	77
Figure 2-8 : Circuit protecteur de bus I2C à base du P82B96 [56]	79
Figure 2-9 : Cadre de référence du Cubesat	80
Figure 2-10 : Exemple de configuration en "slots" [57]	81
Figure 2-11 : Exemple de configuration en couches [62].....	81
Figure 2-12 : configuration en pile selon la norme PC-104 [58]	82
Figure 2-13 : Configuration de MASAT1 [SOLIDWORKS].....	83
Figure 2-14 : Spectre radio utilisé pour la communication des engins spatiaux [3]	83
Figure 2-15 : Diagramme d'états et de transitions entre mode opérationnelles	87
Figure 2-16 : Diagramme bloc matériel et électrique	91
Figure 2-17 : Diagramme bloc représentant le flux des données	92
Figure 3-1 : Diagramme bloc du sous-système EdB	94
Figure 3-2 : Courbes I-V et P-V d'une cellule solaire type GaAs [65]	95
Figure 3-3 : Effet de la température sur les courbes I-V et P-V [66]	96
Figure 3-4 : Spectre solaire [147]	96
Figure 3-5 : Points de fonctionnement résultants de l'association directe d'un GPV avec une charge résistive (R1, R2) ou une batterie (V _{Bat}).....	97
Figure 3-6 : Solution MPPT pour un système PV [66]	98
Figure 3-7 : Transfert direct d'énergie avec un bus d'alimentation régulé [70]	100
Figure 3-8 : Architecture adoptée pour le sous-système EdB de MASAT1	101
Figure 3-9 : Interface électrique du sous-système EdB.....	103
Figure 3-10 : Circuit équivalent simplifié d'une cellule PV [74]	104
Figure 3-11 : Configuration des panneaux PV de MASAT1	105
Figure 3-12 : Panneau solaire de GOMSPACE équipé de 2 cellules PV type 3G30A	105
Figure 3-13 : Courbe I-V d'un panneau PV pour différentes températures [Matlab].....	106
Figure 3-14 : Courbe P-V d'un panneau PV pour différentes températures [Matlab].....	106
Figure 3-15 : Courbe I-V d'un panneau PV pour différents niveaux d'éclairement	107
Figure 3-16 : Courbe P-V d'un panneau PV pour différents niveaux d'éclairement	107
Figure 3-17 : Rendement des convertisseurs DC/DC des différents modules EdB [76].....	108
Figure 3-18: Module EdB GOMSpace Nanopower P31U [79].....	109
Figure 3-19 : Diagramme fonctionnel du module Nanopower P31U [79]	109
Figure 3-20 : Connecteurs de la carte NanoPower P31u [79].....	110
Figure 3-21 : Modèle 3D du Cubesat exporté vers STK.....	114
Figure 3-22 : Puissance instantanée générée par les cellules PV [STK]	114
Figure 4-1 : Diagramme bloc fonctionnel d'un sous-système COM.....	119
Figure 4-2 : Interfaces électriques du sous-système COM.....	121
Figure 4-3 : Architecture du sous-système COM	121
Figure 4-4 : Débits de communication utilisés dans les missions Cubesat [83].....	123
Figure 4-5 : Techniques de modulation utilisés dans les missions Cubesat [83]	124
Figure 4-6 : Taux d'erreur BER pour les modulations BPSK, MSK et FSK [84]	124
Figure 4-7 : Densité spectrale (à gauche : Signal FSK et à droite : signal GMSK)	125
Figure 4-8 : Modélisation de la communication entre segment spatial et station au sol.....	126

Figure 4-9 : Exemples de polarisation d'une onde électromagnétique	128
Figure 4-10 : Pertes à l'air libre pour les différentes bandes de fréquences [84].....	129
Figure 4-11 : La composante radiale de la vitesse radiale d'un satellite [84]	130
Figure 4-12 : Déviation Doppler en fonction de la fréquence de transmission [84].....	130
Figure 4-13 : Atténuations atmosphériques en fonction de l'angle d'élévation [84]	131
Figure 4-14 : Performances BER de différents techniques de modulation [84]	134
Figure 4-15 : Module GOMSpace NanoCom AX100 [89]	137
Figure 4-16 : Diagramme fonctionnel du module NanoCom AX100 [89]	138
Figure 4-17 : Connecteurs de la carte NanoCom AX100 [89].....	139
Figure 5-1 : Architectures des microcontrôleurs utilisées dans les missions Cubesat.....	148
Figure 5-2 : Fabricants des microcontrôleurs utilisées dans les missions Cubesat.....	148
Figure 5-3 : Bus de communication utilisées dans les missions Cubesat.....	148
Figure 5-4 : Interfaces électriques du sous-système OdB.....	149
Figure 5-5 : Schéma bloc fonctionnel de l'OdB de MASAT1.....	150
Figure 5-6 : Cubesat Kit MSP430 de la société Pumpkin [90]	151
Figure 5-7 : Carte NanoMind A712D de GOMSpace [91].....	152
Figure 5-8 : Carte NanoMind A3200 de GOMSpace [92]	153
Figure 5-9 : Schéma bloc fonctionnel de la Carte NanoMind A3200 de GOMSpace [92].....	155
Figure 5-10 : Connecteurs de la carte NanoMind A3200 de GOMSpace [92].....	155
Figure 5-11 : Carte mère NanoDock DMC-3 de GOMSpace [93]	156
Figure 5-12 : Diagramme bloc de le carte mère Nano Dock DMC-3 [93].....	157
Figure 5-13 : Emplacement des connecteurs dans le carte mère Nano Dock DMC-3 [93].....	158
Figure 5-14 : Vue 3D des cartes OdB et COM montée sur la carte mère [SOLIDWORKS]	159
Figure 5-15 : Diagramme bloc électrique de la plateforme MASAT1	160
Figure 5-16 : Vue perspective du P-POD et des ressorts de séparation [10].....	161
Figure 5-17 : Diagramme fonctionnel de la structure d'un Cubesat [62]	162
Figure 5-18 : Structure ISIS CubeSat (à gauche) et Structure Pumpkin Cubesat (à droite)	163
Figure 5-19 : Vue perspective de la structure Cubesat 1U Pumpkin [145].....	164
Figure 5-20 : Vue 3D de la configuration du segment spatial MASAT1 [SOLIDWORKS]	164
Figure 5-21 : Centre de masse du segment spatial MASAT1 [SOLIDWORKS]	166
Figure 5-22 : Méthodes de contrôle d'attitude en fonction de la taille du Cubesat [34]	167
Figure 5-23 : Alignement d'un cubesat sur le champ magnétique terrestre [Internet].....	167
Figure 5-24 : Cycle d'hystérésis typique [95]	171
Figure 5-25 : Modèle approximatif du cycle d'hystérésis (en vert) [96]	172
Figure 6-1 : Diagramme préliminaire des transitions entre les modes opérationnels	177
Figure 6-2 : Algorithme des opérations en mode séparation	178
Figure 6-3 : Algorithme des opérations en mode Initialisation.....	179
Figure 6-4 : Algorithme des opérations en mode Sans Echec.....	180
Figure 6-5 : Algorithme des opérations en mode Nominal.....	180
Figure 6-6 : Topologie réseau utilisée pour la mission MASAT1	182
Figure 6-7 : La pile de protocole CSP [97]	183
Figure 6-8 : Structure de l'entête CSP dans les trames de communication [97]	184
Figure 6-9 : Trame du protocole AX25 [98].....	186
Figure 6-10 : Trame du protocole PUS selon la norme ECSS-E-70-41A [99]	187
Figure 6-11 : En tête principale de la trame du protocole PUS [99]	187
Figure 6-12 : En tête secondaire de la trame du protocole PUS pour les TC [99].....	187
Figure 6-13 : En tête secondaire de la trame du protocole PUS pour la TM [99]	188
Figure 6-14 : Systèmes d'exploitation temps réel adoptées pour les missions Cubesat	189
Figure 6-15 : Fichiers sources constituant le module FreeRTOS [101]	190
Figure 6-16 : Machine à états des tâches pour FreeRTOS [101]	191
Figure 6-17 : 1 ^{ère} implémentation de la gestion mémoire dans FreeRTOS [101]	192
Figure 6-18 : 2 ^{ème} implémentation de la gestion mémoire dans FreeRTOS [101]	193
Figure 6-19 : Organisation du logiciel embarqué dans l'OdB de MASAT1	194
Figure 7-1 : Historique et prédictions des lancements de nano/microsatellites [13].....	197

Figure 7-2 : Architecture générale d'un circuit FPGA [105]	200
Figure 7-3 : Gestionnaire d'horloge d'un FPGA [106]	201
Figure 7-4 : Schéma d'une Cellule SRAM	202
Figure 7-5 : Structure de programmation à base de point mémoire SRAM [106].....	202
Figure 7-6 : Transistor à grille flottante utilisé dans la mémoire flash	203
Figure 7-7 : Classification des événements singuliers SEE	204
Figure 7-8 : Evènements SEU dans la mémoire de configuration d'un FPGA Xilinx.....	205
Figure 7-9 : Duplication avec comparaison	207
Figure 7-10 : Triplification modulaire TMR	208
Figure 7-11 : Voteur de la majorité et sa table de vérité [119].....	208
Figure 7-12 : auto reconfiguration (à gauche) et reconfiguration externe (à droite).....	212
Figure 7-13 : Diagramme fonctionnelle d'une architecture auto-reconfigurable	213
Figure 7-14 : Architecture du contrôleur de reconfiguration Xilinx avec interface ICAP.....	216
Figure 7-15 : Schéma bloc fonctionnel de l'OdB secondaire de MASAT1	216
Figure 7-16 : Schéma bloc fonctionnel du softcore MicroBlaze [121]	219
Figure 7-17 : Diagramme bloc fonctionnel de l'IP core AXI_HWICAP de Xilinx.....	221
Figure 7-18 : Comparaison des mémoires Flash NORT et Flash NAND [127]	223
Figure 7-19 : Circuit électrique du bloc mémoire Flash SPI	224
Figure 7-20 : Circuit électrique du bloc mémoire DDR3 SDRAM	225
Figure 7-21 : Circuit électrique du bloc mémoire EEPROM IIC	226
Figure 7-22 : Circuit électrique du bloc alimentation 3.3V et 1.5V	227
Figure 7-23 : Circuit électrique du bloc alimentation 2.5V et 1.8V.....	227
Figure 7-24 : Circuit électrique du bloc alimentation DDR VTT	228
Figure 7-25 : Circuit imprimé de l'OdB secondaire face composants [Altium Designer]	229
Figure 7-26 : Circuit imprimé de l'OdB secondaire face soudure [Altium Designer]	229
Figure 7-27 : Interface de l'environnement de développement XPS de Xilinx	230
Figure 7-28 : Etapes de la conception d'un système embarqué dans un environnement Xilinx.....	231
Figure 7-29 : Modules constituant XilKernel [80]	232
Figure 7-30 : Etapes pour créer une application XilKernel.....	233
Figure 7-31 : Interface de configuration série de l'OdB secondaire MASAT1.....	234
Figure 7-32 : Plan mémoire de la mémoire Flash SPI.....	235
Figure 7-33 : Diagramme bloc fonctionnel du système de détection d'erreur Lockstep.....	237
Figure 7-34 : Les ports de l'IP core MicroBlaze Comparator	238
Figure 7-35 : Système référence de la plateforme matérielle implémentée sur le FPGA	239
Figure 7-36 : Sous-programme d'injection et de récupération d'erreurs	241
Figure 7-37 : Bitstream pour une commande IPROG via ICAP	241
Figure 7-38 : Paramétrage du mode de configuration et LED indicatrice d'erreur.....	242
Figure 7-39 : Démarrage de l'application logicielle pour le test d'injection d'erreur	242
Figure 7-40 : Test du comparateur et effacement du registre indicateur d'erreurs.....	243
Figure 7-41 : Injection et récupération d'erreur par reconfiguration dynamique complète.....	243
Figure 7-42 : Modèle logique de processus d'implémentation du multiboot/fallback	244
Figure 7-43 : Flux des opérations logicielles pour créer le fichier .mcs	245
Figure 7-44 : Script exécutable pour générer le fichier MCS final	246
Figure 7-45 : Script exécutable pour générer le fichier MCS final	247
Figure 7-46 : L'état du registre BOOTSTS après une reconfiguration réussie	247
Figure 7-47 : L'état du registre BOOTSTS après une opération fallback	247
Figure 7-48 : Description fonctionnelle de la méthode de mise à jour distante	248
Figure 7-49 : Flux logiciel pour la préparation des fichiers images mis à jour	249
Figure 7-50 : Flux logiciel du téléchargement du fichier image via le serveur TFTP	251
Figure 7-51 : Plan mémoire de la mémoire Flash SPI pour la mise à jour à distance	252
Figure 7-52 : Script exécutable pour générer le fichier MCS final	252
Figure 7-53 : Fenêtre terminal Windows affichant l'exécution de l'application initiale	254

LISTE DES TABLEAUX

Tableau 1 : Classification des satellites en fonction de leur masse	3
Tableau 2 : Certains fournisseurs spécialisés pour les petits satellites.....	7
Tableau 3 : Description des niveaux de réussite de la mission	13
Tableau 1-1 : Identificateurs et types d'exigences définies dans le DES.....	25
Tableau 1-2 : Aperçu des exigences générales du segment spatial de la mission [71].....	26
Tableau 1-3 : scénarios orbitaux de la mission MASAT1.....	34
Tableau 1-4 : Paramètres pour la simulation des effets de la trainée atmosphérique	43
Tableau 1-5 : Paramètres pour la simulation des effets de la trainée atmosphérique	47
Tableau 1-6 : Temps de communication estimés par calcul	51
Tableau 1-7 : Temps de communication pour la station au sol d'Ifrane [STK]	52
Tableau 1-8 : Durée d'éclipse maximale calculée et simulée durant 6 mois de mission.....	55
Tableau 1-9 : Récapitulatif des effets radiatifs avec un blindage de 2mm [SPENVIS]	63
Tableau 1-10 : Facteurs de charges quasi-statiques des différents lanceurs [41-47]	64
Tableau 1-11 : Niveau de pression acoustique à l'intérieur du carénage (en dB) [41-47].....	65
Tableau 1-12 : Niveaux d'acceptation et de qualification [41-47]	67
Tableau 2-1 : Avantages et inconvénients des architectures distribuée et centralisée	74
Tableau 2-2 : Comparaison des bus de communication série I2C et CAN	78
Tableau 2-3 : Bandes de fréquences disponibles pour les CubeSat [61]	84
Tableau 2-4 : Etats possibles en mode opérationnel nominal.....	89
Tableau 3-1 : Caractéristiques électriques des cellules PV type 3G30A [146].....	104
Tableau 3-2 : Comparaison des modules EdB COTS des différents fournisseurs [90 - 93]	108
Tableau 3-3 : Description des broches des connecteurs H1 et H2 [79]	110
Tableau 3-4 : Exigences de stockage d'énergie pour MASAT1.....	111
Tableau 3-5 : Contraintes et hypothèses pour le bilan de puissance	113
Tableau 3-6 : Cas de consommation d'énergie possibles en mode nominal.....	115
Tableau 3-7 : Consommation mode EL-TCU.....	115
Tableau 3-8 : Consommation mode EL-nTCU	115
Tableau 3-9 : Consommation mode EL-TnCU	115
Tableau 3-10 : Consommation mode EL-nTnCU	115
Tableau 3-11 : Consommation mode EP-TCU	116
Tableau 3-12 : Consommation mode EP-nTCU	116
Tableau 3-13 : Consommation mode EP-TnCU	116
Tableau 3-14: Consommation mode EP-nTnCU.....	116
Tableau 3-15 : Combinaisons de consommation d'énergie en mode nominal.....	116
Tableau 3-16 : Scénario de consommation d'énergie en mode nominal	117
Tableau 4-1 : Pertes de polarisation des antennes [84].....	128
Tableau 4.2 : Bilan de la liaison montante [84].....	135
Tableau 4-3 : Bilan de la liaison descendante	136
Tableau 4-4 : Caractéristiques des modules COM des différents fournisseurs	137
Tableau 5-1 : Estimation de la taille du logiciel embarqué et du temps de traitement [8]	145
Tableau 5-2 : Estimation de la taille mémoire statique pour les données de TM	147
Tableau 5-3 : Broches du bus PC-104 permettant d'accéder au sous-système OdB	158
Tableau 5-4 : Broches du bus PC-104 permettant d'accéder au sous-système COM.....	159
Tableau 5-5 : Bilan de masse préliminaire de la mission MASAT1.....	165
Tableau 5-6 : Ordre de grandeurs des couples perturbateurs pour les missions CubeSat 1U MCubed-2, AAU Sat et CalPoly PolySat [94]	169
Tableau 5-7 : Caractéristiques des aimants utilisés dans les mission Cubesat [94].....	170
Tableau 5-8 : Matériaux hystérésis utilisés dans les mission Cubesat [94]	173
Tableau 6-1 : Description des modes opérationnels de MASAT1	176
Tableau 6-2 : Description des RTOS potentiellement utilisables pour MASAT1	189
Tableau 6-3 : Tâches réalisées par les modules de la couche application	195

Tableau 7-1 : Niveau d'autonomie de la DIRD selon la norme ECSS-E-ST-70C [103].....	198
Tableau 7-2 : Méthodes de correction et de détection des erreurs.....	209
Tableau 7-3 : Fonctionnalités de reconfiguration des FPGA Xilinx [120].....	215
Tableau 7-4 : Caractéristiques des FPGA type SRAM de Xilinx [107 - 109].....	218
Tableau 7-5 : Résumé des caractéristiques du FPGA Spartan-6 LX-45T [107 - 109].....	218
Tableau 7-6 : Entête d'un paquet de configuration type 1 [123].....	221
Tableau 7-7 : Description de certains registres de configuration [123].....	221
Tableau 7-8 : Fonctionnalités et avantages d'un bloc MCB du Spartan 6 [129].....	224
Tableau 7-9 : Tension générée par le bloc d'alimentation de l'OdB secondaire.....	226
Tableau 7-10 : Mode de configuration du FPGA Spartan-6 [123].....	234
Tableau 7-11 : Fichier entête pour une commande IPROG via ICAP.....	236
Tableau 7-12 : Signaux de l'interface "Lockstep" de MicroBlaze [121].....	237
Tableau 7-13 : Registres de l'Ip Core MicoBlaze Comparator [117].....	238
Tableau 7-14 : Principales fonctions utilisées pour implémenter le serveur TFTP [80].....	250

Liste des abréviations

ANRT	Agence Nationale de Réglementation des Télécommunication
ATT	Sous-système contrôle d'attitude
BDR	Battery Discharge Regulator
BER	Bit Error rate
BOL	Begin Of Life
BSB	Base System Builder
BSTI	Basic Space Technology initiative
CdG	Centre de Gravité
CIDR	Classless InterDomain Routing
CIRA72	COSPAR International Reference Atmosphere 1972
CEM	Compatibilité Electromagnétique
CME	Coronal Mass Ejection
COM	Sous-système Communication
ConOps	Concept des Opérations
COTS	Commercial Off-The-Shelf
CQS	Charges Quasi-Statiques
CRERS	Centre Royal d'Etudes et de Recherches Spatiales
CSP	Cubesat Space Protocol
CTR	Compile Time Reconfiguration
DDD	Dose de Dommage de Déplacement
DES	Documents des Exigences Système
DIRD	Détection, Isolation et récupération des Défaillances
DMA	Direct Memory Access
DOD	Depth Of Decharge
ECI	Earth Centred Inertial
ECSS	European Cooperation for Space Standardization
EDAC	Error Detection and Correction Codes
EdB	Sous-système Energie de Bord
ElaNa	Educational Launch of Nanosatellites
EOL	End Of life
ESA	European Space Agency
FPGA	Field-Programmable Gate Array
GCR	Rayons Cosmiques Galactiques
GPV	Génératrice Photovoltaïque
HDLC	High-Level Data Link Control
I2C	Inter-Integrated Circuit
IARU	International Amateur Radio Union
ICAP	Internal Configuration Access Port
IP	Intellectual Property
ITU	International Telecommunication Union
JAXA	Japan Aerospace Exploration Agency
JD	Julian Date
JPL	Jet Propulsion Laboratory
JTAG	Joint Test Action Group
KS	Kill Switch
LEO	Low earth Orbit
LET	Linear Energy Transfert
LEOP	Launch and Early Orbit Phase
LUT	Look Up table
LwIP	lightweight IP
MAC	Medium Access control

MI	Mode Initialisation
MIPS	Million of Instruction Per Second
ML	Mode Lancement
MN	Mode nominal
MN_nTCU	Mode Nominal sans Transmission et avec Charge Utile active
MN_nTnCU	Mode Nominal sans Transmission et avec Charge Utile inactive
MN_TCU	Mode Nominal avec Transmission et Charge Utile active
MN_TnCU	Mode Nominal avec Transmission et avec Charge Utile inactive
MPPT	Maximum Power Point Tracking
MS	Mode Séparation
MSE	Mado Sans Echec
NASA	National Aeronautics and Space Administration
NIEL	Non Ionizing Energy Loss
OEM	Onde Electromagnétique
OdB	Sous-système Ordinateur de Bord
PPM	Point de Puissance Maximale
P-POD	Poly-Picosatellite Orbital Deployer
PRD	Project Requirements Document
PLD	Programmable Logic Device
PV	Photovoltaïque
RAAN	Right Ascension of the Ascending Node
RBF	Remove Before Fligh
RDP	Reliable Datagram Protocole
RF	Radio Fréquence
RTC	Real Time Clock
RTOS	Real Time Operating System
RTR	Run Time Reconfiguration
S3R	Sequential Switching Shunt Regulator
SCL	Serial Clock Line
SDA	Serial Data Line
SEE	Single Event Effects
SEFI	Single Event Functional Interrupt
SEGR/SEB	Single Event Gate Rupture / Burnout
SEL	Single Event Latchup
SEU	Single Event Upset
SET	Signle Event Transient
SLOC	Source Ligne Of Code
SNR	Signal-to-Noise Ratio
SoC	System on Chip
SPI	Serial Peripheral Interface
SPP	Space Packet Protocol
SSETI	Student Space Exploration and Technology Initiative
SSP	Sub-Satellite Point
STR	Sous-système Structure
TC	Télécommandes
TCOM	Temps de Communication
THER	Sous-système Thermique
TID	Total Ionizing Dose
TM	Téléométrie
TMR	Triple Modular Redundancy
TWI	Two Wire Interface
UNOOSA	United Nations Office for Outer Space Affairs
WBS	Work Breakdown Structure

INTRODUCTION GENERALE

1. CONTEXTE DE LA RECHERCHE

L'espace est probablement le principal symbole du progrès technologique dans notre société moderne, avec tout ce qu'il peut offrir comme perspectives de recherche et de développement du capital intellectuel, technologique et économique d'une nation. C'est un domaine qui est longtemps resté hors de portée de pays ou d'organisations ayant des budgets limités ou une faible expérience dans les technologies spatiales. Cependant, l'avènement du standard Cubesat et l'essor qu'a connu la microélectronique et les composants COTS (Commercial Off-The-Shelf), ont permis de développer des missions spatiales moins coûteuses en termes de budget, de durée de conception et de ressources humaines et technologiques [1].

Les nanosatellites, en tant que solution technologique standardisée et économiquement viable, ont permis aux universités d'un nombre croissant de pays émergents, dont le Brésil, la Colombie, l'Argentine, l'Afrique du sud et la Turquie [2], de développer une large gamme de missions spatiales en orbite terrestre basse LEO (Low Earth Orbit). Mais, bien qu'initiée par les universités pour leur valeur ajoutée dans l'enseignement et la recherche, le développement de la technologie des nanosatellites a été considérablement accéléré par l'implication croissante des agences gouvernementales de recherche spatiale. Ainsi :

- L'agence spatiale américaine NASA (National Aeronautics and Space Administration) a initié depuis 2008, le programme éducatif de lancement de nanosatellites ELaNa (Educational Launch of Nanosatellites) afin de faciliter l'accès à l'espace pour les universités et les organisations à but non lucratif [135-4].
- Le département de défense américain, en collaboration avec la NASA, a créé en 2007 l'ORS (Operationally Responsive Space Office) pour permettre à l'armée américaine de disposer de solutions en matière de conception et de lancement de petits satellites. Une des études réalisées a permis d'évaluer le modèle architectural DoDAF permettant d'intégrer un processus d'ingénierie des systèmes satellitaires pour accroître la réactivité et réduire les coûts [4].
- L'agence spatiale européenne ESA (European Space Agency) a initié deux programmes visant à encourager la recherche spatiale en permettant aux étudiants européens d'avoir une expérience pratique des différentes phases de conception, de lancement et d'exploitation des nanosatellites : le programme "Fly your satellite!" en 2008 [136] et le programme SSETI (Student Space Exploration and Technology Initiative) en 2010 [137].

- L'Organisation des Nations Unies (ONU), à travers son bureau des affaires spatiales UNOOSA (United Nations Office for Outer Space Affairs), promeut la coopération internationale et le renforcement des capacités dans le domaine des techniques spatiales, en particulier dans les pays en développement. Elle a lancé en 2009 l'Initiative des Nations Unies sur les technologies spatiales fondamentales BSTI (Basic Space Technology Initiative) visant à améliorer l'accès aux outils des applications spatiales pour le développement durable, en renforçant les capacités dans les technologies spatiales de base [138].

Dans ce contexte et vu que le secteur de l'aéronautique et du spatial a été identifié par le Maroc comme un moteur de croissance pour l'économie nationale, le Centre National de Recherche Scientifique et Technique (CNRST) a lancé, au début de l'année 2012, une série d'ateliers de formation au profit des universitaires marocains dans le but de réaliser un microsatellite universitaire de télécommunication spatiale. C'était le point de départ de cette thèse de Doctorat.

En 2015, le CNRST a lancé un appel à projets (Projets prioritaires de la recherche), auquel on a participé en tant que consortium regroupant la Faculté des Sciences Dhar El Mahraz de l'Université Sidi Mohamed Ben Abdellah de Fès, l'université Al Akhawayn d'Ifrane et l'Ecole Supérieure de Technologie (EST) d'Oujda. Notre projet intitulé MASAT1, a été accepté et nous avons reçu une subvention du Ministère de tutelle de 900000 DH. Il s'agit d'un projet de recherche appliquée qui vise à réaliser le premier nanosatellite universitaire marocain. L'ambition principale du projet est de promouvoir la recherche et le développement dans le domaine de l'ingénierie spatiale, ainsi que de rendre accessible l'exploration scientifique spatiale aux universitaires marocains.

2. ETAT DE L'ART DES NANOSATELLITES

2.1. Classification des petits satellites

Pour commencer l'ère spatiale, trois petits satellites ont été les premiers à être lancés en orbite terrestre basse (LEO) pendant l'Année Géophysique Internationale (IGY) de 1957 à 1958 : Sputnik I, Explorer I et Vanguard 1 (voir *figure 1*).

Mais bien que depuis, plusieurs petits satellites ont été construits et lancés, on n'a commencé à les classer en fonction de leur masse que depuis 1992 [6]. L'utilité de ce type de classification réside dans l'incidence directe de la masse sur les coûts de conception, le lancement et le type de charge utile du satellite.

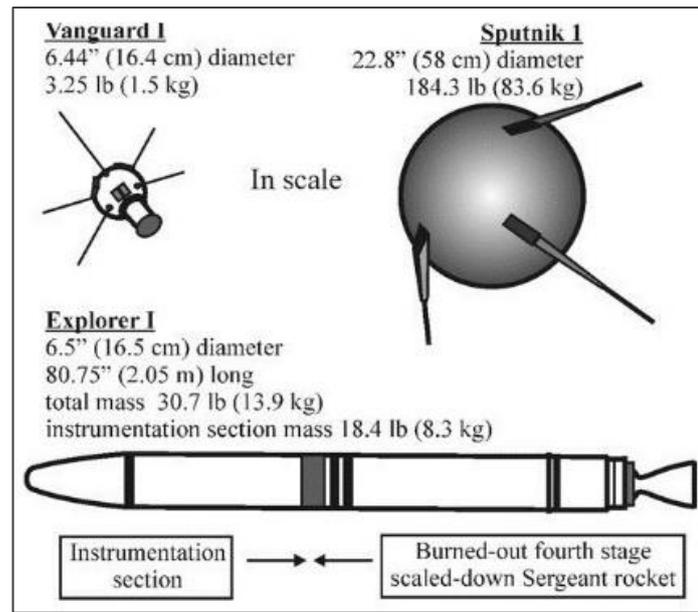


Figure 1 : Vues des segments spatiaux des missions Sputnik I, Explorer I et Vanguard 1 [5]

Un système de classification a d'abord été proposé par le centre de recherche en ingénierie satellitaire de l'Université de Surrey en Angleterre, et il a été facilement adopté par toute la communauté satellitaire [7]. Le **tableau 1** résume cette classification, où chaque classe s'étend sur une plage de masse déployée qui est définie comme la masse du satellite plus celle du combustible :

Catégorie	Masse déployée (Kg)	Description
Large satellite	> 1000	
Moyen satellite	500 – 1000	
Mini-satellite	100 – 500	Seuil limite pour les petits satellites
Microsatellite	10 – 100	
Nanosatellite	1 – 10	Cubesat appartient à cette classe
Picosatellite	0.1 – 1	
Femtosatellite	< 0.1	

Tableau 1 : Classification des satellites en fonction de leur masse

2.2. Description système d'un nanosatellite

Du point de vue système, il est pratique de subdiviser un engin spatial ou satellite en plusieurs éléments fonctionnels comme le montre la **figure 2**. On en distingue ainsi :

- Un segment spatial communément appelé satellite.
- Un segment au sol constitué d'une ou plusieurs stations au sol dont le rôle est d'envoyer les commandes au segment spatial et de recevoir les données de télémétrie et celles de la charge utile embarquée.

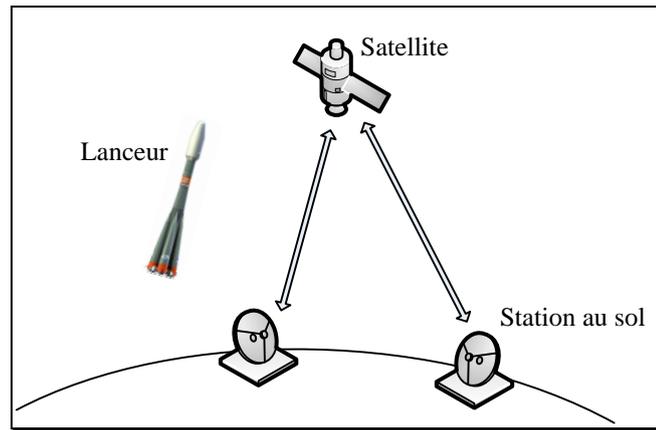


Figure 2 : Vue globale du système englobant segment spatial et segment au sol

Chacun des éléments du système spatial peut être considéré comme un sous-système exécutant des fonctions qui lui seront associées, et interagissant avec les autres en fonction des objectifs et exigences fonctionnelles de la mission.

Dans le cas d'un nanosatellite, le segment spatial est généralement constitué de deux parties (*Figure 3*) :

- **La charge utile** qui dépend de la mission. Vu la contrainte de masse des nanosatellites, cette charge utile consiste généralement à faire de l'expérimentation scientifique dans l'espace (évaluer, vérifier et valider certaines technologies et protocoles de communication).

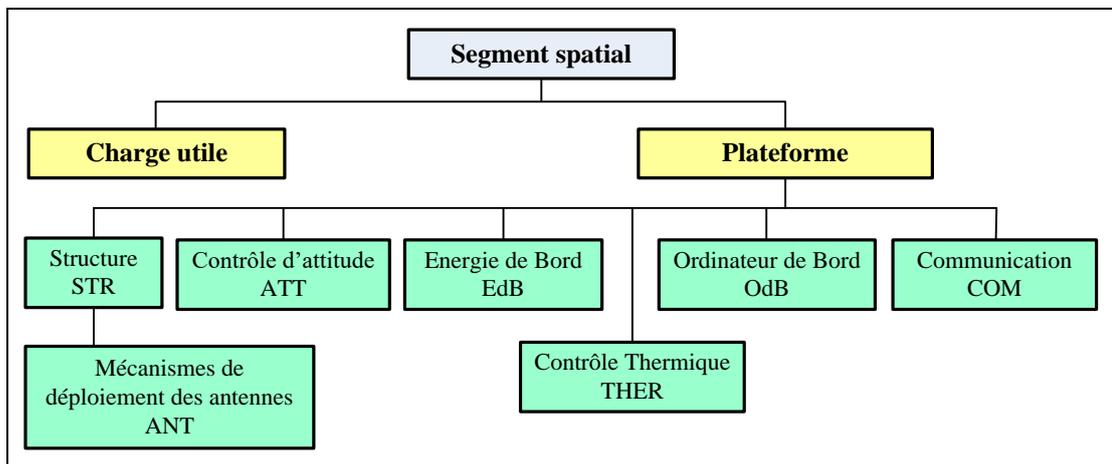


Figure 3 : Sous-systèmes du segment spatial

- La plateforme du segment spatiale est constituée de plusieurs sous-systèmes qui assurent le bon déroulement de la mission. Pour les nanosatellites, il s'agit généralement des sous-systèmes suivants :

- **Structure (STR)** qui assure, d'une part un interfaçage mécanique entre le segment spatial et le lanceur, et d'autre part un environnement de fonctionnement sécurisé contre les effets des radiations spatiales.
- **Contrôle d'attitude (ATT)** permet d'assurer une orientation optimale du segment spatial dans l'espace en fonction des objectifs de la mission.
- **Energie de Bord (EdB)** qui a pour tâches principales la production de l'énergie électrique provenant des cellules solaires, son stockage dans des batteries, et sa régulation/distribution sécurisée aux différents sous-systèmes embarqués dans le segment spatial en fonction de leurs besoins en termes de consommation.
- **Contrôle Thermique (THER)** qui doit garantir au segment spatial un environnement thermique permettant son fonctionnement nominal, sachant que la température dans l'environnement spatiale peut varier de -130°C à $+100^{\circ}\text{C}$ en quelque dizaines de secondes ou minutes [8].
- **Communication (COM)** qui est composé d'un émetteur/récepteur radio permettant d'assurer un lien fiable entre le segment spatial et celui au sol.
- **Ordinateur de Bord (OdB)** qui est le système principal de traitement des données et des télécommandes. Il doit ainsi garantir un état opérationnel stable du segment spatial en assurant toute les tâches de supervision et de contrôle.
- **Mécanisme de déploiement des Antennes (ANT)** qui est un système spécifique pour les nanosatellites, avec comme rôle principal le maintien des antennes en place lors du lancement, et leur déploiement une fois le segment spatial en orbite.

3. LE CONCEPT CUBESAT

3.1. Configurations

Le programme Cubesat a été lancé à l'université de Stanford au début de l'année 1999 pour répondre au besoin éducatif d'avoir un satellite qui pourrait être développé dans un délai de 2 ans, à très faible coût et avec un faible poids pour réduire les coûts de lancement. L'ambition était d'avoir un outil standard pour enseigner aux étudiants les processus de développement, de lancement et d'exploitation d'un satellite [9]. L'idée a officiellement été proposée par le Professor Robert Twiggs au 2^{ème} Symposium universitaire des systèmes spatiaux (USSS) qui s'est déroulé à Hawaï en Novembre 1999.

Dans le but de fournir une plateforme standard pour le développement, la spécification Cubesat [10] a été introduite suite à une collaboration entre l'Université Stanford et l'Université Polytechnique de Californie (Cal Poly) [11].

Cette norme définit la configuration du segment spatial, la masse, le volume, les dimensions extérieures, les procédures de test et les contraintes énergétiques et opérationnelles. Les configurations les plus connues sont les Cubesats 1U, 2U et 3U, où chaque "U" représente un cube de 10 cm. Par exemple, un Cubesat 1U est défini comme un nanosatellite avec une dimension nominale de 10 x 10 x 10 cm et une masse allant jusqu'à 1,33kg, comme le montre la *figure 4* :

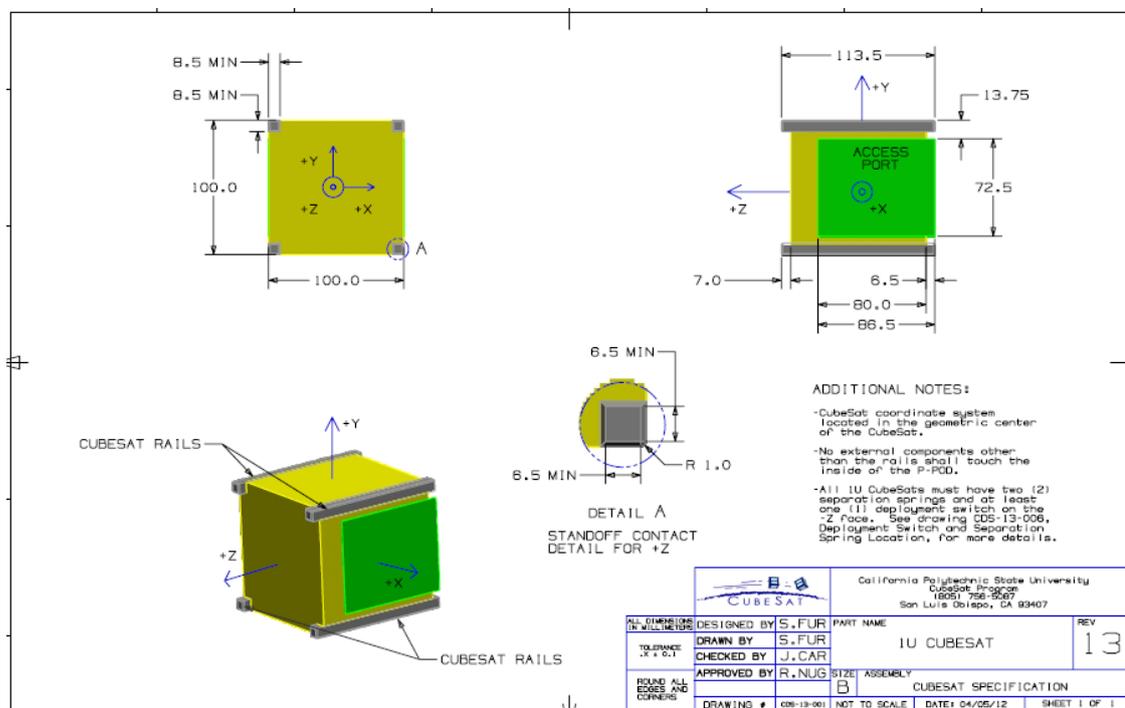


Figure 4 : Spécifications de conception d'un CubeSat Format 1U [10]

Depuis la standardisation des Cubesat, le coût comparativement faible des projets de recherche spatiale et des activités de développement technique qui répondent aux contraintes de taille, de masse et de puissance des satellites de moins de 10 kg, a attiré quelques 80 établissements universitaires dans le monde [1]. Les sous-systèmes des plateformes Cubesat (qualifiés espace) sont devenus disponibles auprès de plusieurs fournisseurs spécialisés, ce qui devrait favoriser encore plus la croissance des missions spatiales dans le monde universitaire.

Le **tableau 2** résume certains des fournisseurs leaders sur le marché des Cubesat :

Compagnie	Produits et/ou services	Date création/pays
Pumpkin, Inc.	Kits Cubesat et services d'intégration	1995, Etats Unis
Clyde Space	Sous-systèmes pour Cubesat	2005, Angleterre
ISIS	Petits satellites et services d'intégration/lancement	2006, Hollande
GOMSpace	Petits satellites et logiciels	2007, Danemark

Tableau 2 : Certains fournisseurs spécialisés pour les petits satellites

3.2. Technologies

Les Cubesats sont actuellement conçus pour les orbites terrestres basses (LEO) et l'étendue de leurs applications a considérablement augmenté au cours de la dernière décennie, en grande partie grâce aux progrès technologiques dans des domaines comme la microélectronique et les composants COTS, les modules de communication faible puissance, les cellules solaires photovoltaïques à rendement élevé, les batteries à haute densité d'énergie, les microcontrôleurs à faible consommation et les moteurs/actionneurs miniatures à rendement élevé. Les missions Cubesats ont permis de démontrer l'utilisation avec succès de composants standards COTS pour le développement des différents sous-systèmes du segment spatial [3].

Néanmoins, la petite masse et les dimensions propres aux Cubesats limitent considérablement la quantité d'énergie disponible en orbite, ce qui reste une contrainte à la fois pour la capacité de traitement des processeurs, pour l'utilisation de systèmes redondants, pour le système de communication qui ne dispose pas de suffisamment de puissance pour émettre en continu, et pour le choix de la mission utile.

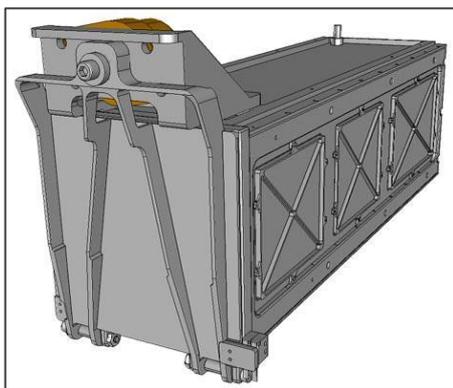
Ces possibilités limitées au niveau de la communication, mais aussi au niveau des bandes passantes disponibles, imposent des contraintes à la manipulation des données, à la détection et à la correction des défaillances, et au contrôle du segment spatial depuis le segment au sol. Pour cette raison, le segment spatial doit implémenter un certain degré d'autonomie pour gérer la plupart des défaillances, ainsi que son mode de fonctionnement selon l'énergie disponible.

3.3. Systèmes de déploiement

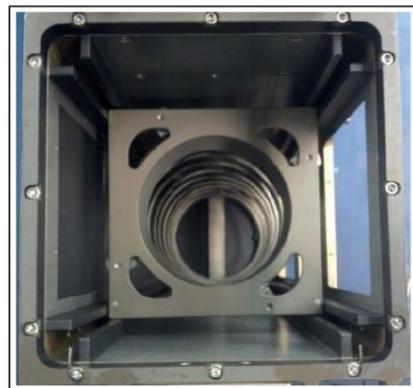
La clé de réussite du concept Cubesat est de pouvoir utiliser un système de déploiement en orbite qui répond à un certain nombre d'exigences, à savoir [10] [11] :

- Le système de déploiement doit protéger le lanceur et la charge primaire contre toute interférence mécanique, électrique ou magnétique provenant des Cubesats qui sont considérés comme des charges secondaires.
- Les Cubesats doivent pouvoir être libérés du système de déploiement en utilisant le minimum de ressorts pour minimiser les probabilités de collision avec le lanceur et avec les autres Cubesats.
- Le système de déploiement doit pouvoir s'interfacer avec le maximum de lanceurs disponibles et sans modification de la norme Cubesat.
- Le système de déploiement doit avoir une masse réduite tout en ayant une conception modulaire lui permettant de lancer plusieurs Cubesats par mission.
- Tous les constituants du Cubesat doivent rester attachés pendant le processus de lancement et de déploiement. Aucun débris spatial supplémentaire ne doit être créé.

La solution la plus utilisée est un "conteneur" nommé P-POD (Poly Picosatellite Orbital Deployer), développé par l'Université Polytechnique de l'état de Californie (Cal Poly). Certifié par un grand nombre de lanceurs [12], le P-POD Mark III fournit une enceinte suffisamment solide pour supporter une défaillance structurelle des trois Cubesats qu'il peut contenir, tout en jouant le rôle d'une cage de Faraday pour protéger la charge utile primaire du lanceur. Le déploiement est déclenché par la commande électrique d'ouverture de la porte du P-POD, un mécanisme standard avec un ressort à piston fournissant une force de poussée qui éjecte les Cubesats à une vitesse linéaire d'environ 0,3 m/s [11], limitant la rotation des Cubesats après le déploiement (Voir *figure 5*) :



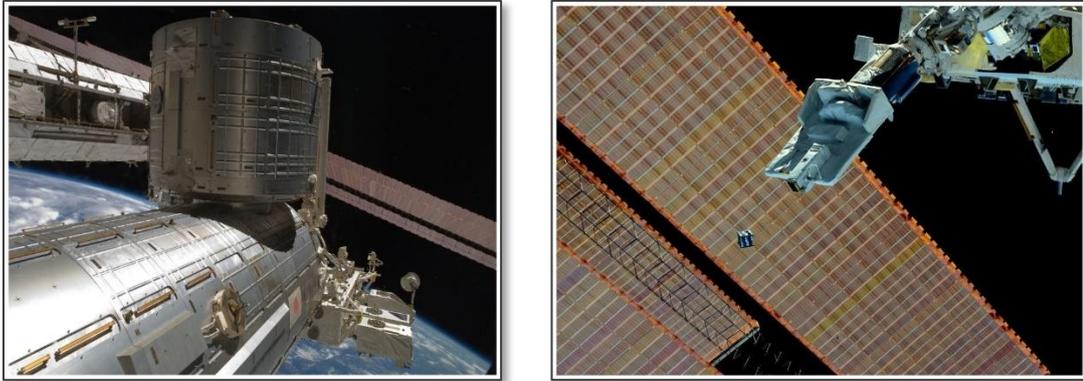
(a) Conteneur P-POD / Mark III



(b) Mécanisme de déploiement avec rails

Figure 5 : Système de déploiement P-POD [10]

Il est à noter que depuis l'année 2008, le déploiement des Cubesats depuis la Station Spatiale Internationale (ISS), qui est située à une altitude d'environ 400 km, est possible grâce au module "KiboCUBE" développé par l'agence japonaise d'exploration aérospatiale (JAXA) [139]. Ce système, unique en son genre, est composé d'un sas et d'un bras robotique comme le montre la *figure 6* :



(a) Module Kibo monté sur l'ISS

(b) Déploiement d'un CubeSat depuis l'ISS

Figure 6 : Système de déploiement KiboCUBE [139]

Le projet est mené en collaboration avec l'UNOOSA et le premier déploiement orbital de Cubesats avec le module "KiboCube" a été mené avec succès en octobre 2012. Ce type de déploiement a l'avantage de générer moins de vibrations que le déploiement direct via un lanceur, ce qui entraîne des contraintes d'interfaçage relativement moins exigeantes pour les Cubesats.

4. HISTORIQUE DES MISSIONS CUBESAT

E. Kulu [140] a réalisé une base de données regroupant les missions cubesat, communément appelé nanosatellite, lancés depuis 2002 jusqu'à aujourd'hui. Il y a eu approximativement 580 nanosatellites lancés à partir de 18 lanceurs différents avec des échecs portant sur 70 cubesats comme le montre la *figure 7* qui illustre l'état de ces nanosatellites.

Le secteur des compagnies commerciales et les universités représentent les principaux acteurs comme le montre la *figure 8*, avec une plus grande croissance prévue pour le secteur commercial dans le futur.

La *figure 9* montre un classement des cubesats lancés par type d'organisation avec une prédominance pour les Cubesats 3U, tandis que la *figure 10* montre la répartition géographique des développeurs de nanosatellites.

On voit clairement que les États-Unis sont les plus grands développeurs dans ce domaine alors que le continent africain n'atteint même pas le seuil des 1% de nanosatellites lancés.

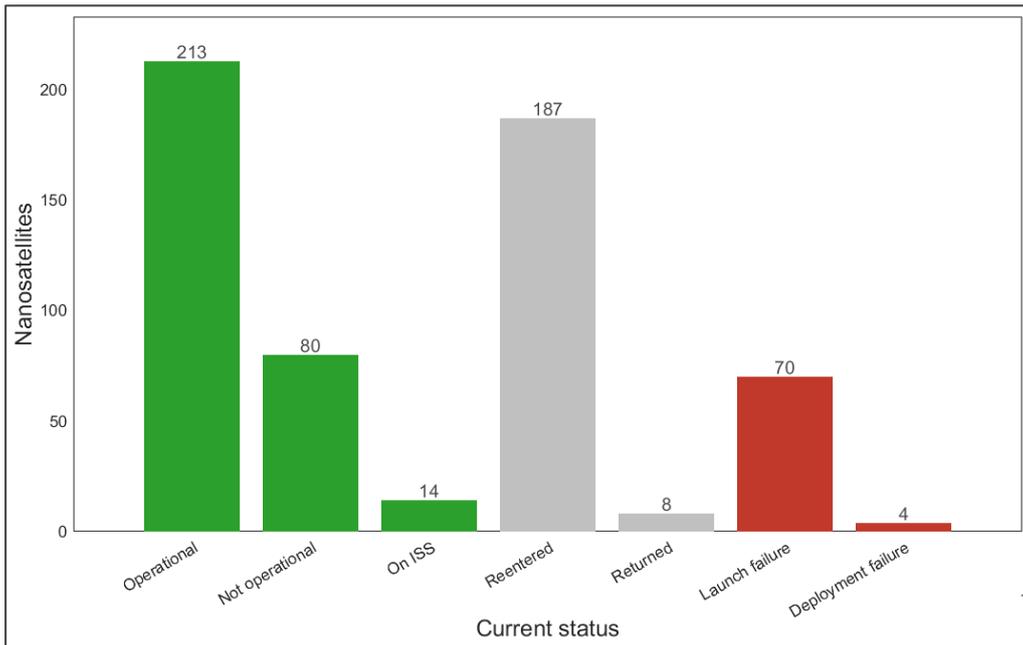


Figure 7 : Etat des nanosatellites lancés à la date du 08/01/2017 [140]

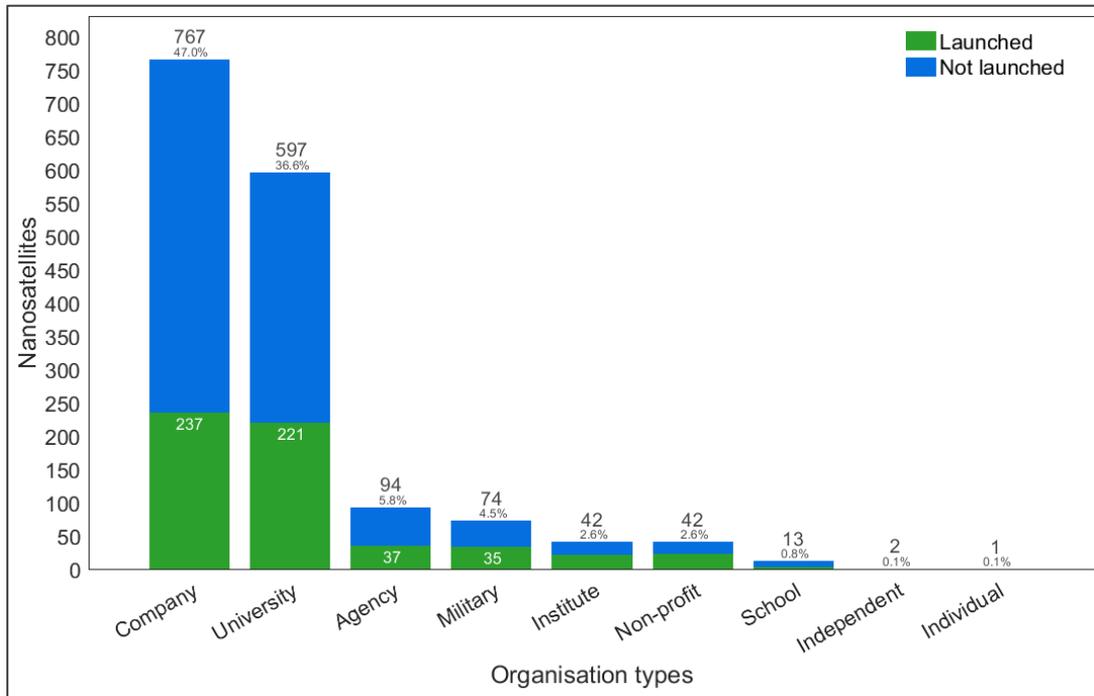


Figure 8 : Types d'organisations qui participent aux lancements des nanosatellites [140]

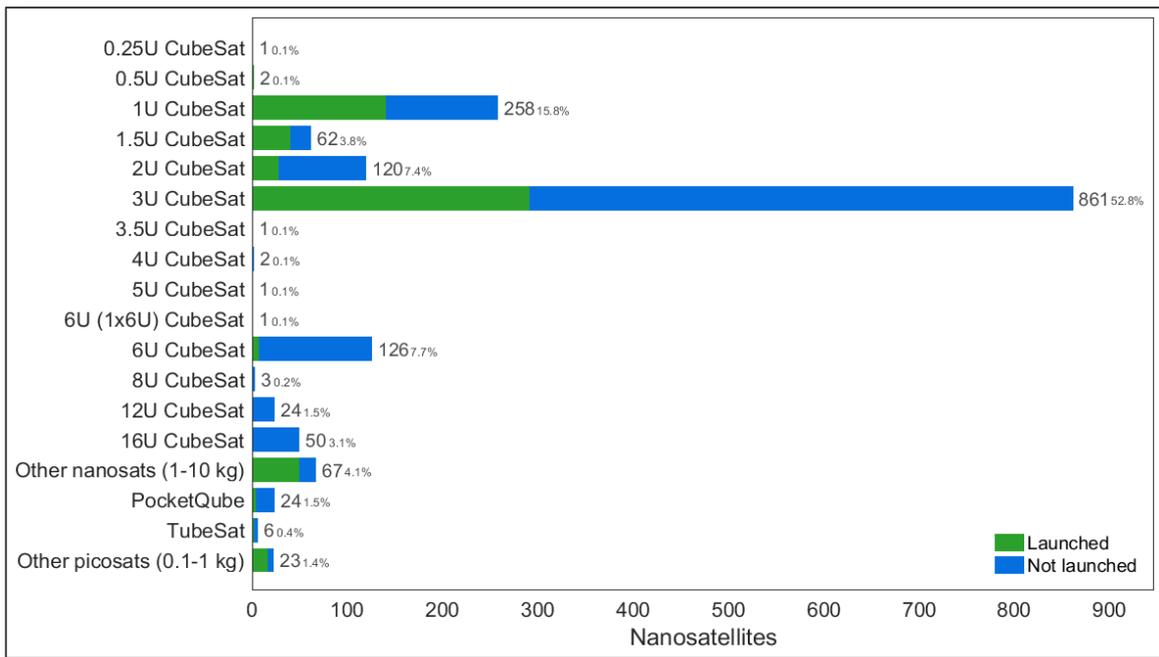


Figure 9 : Classement des nanosatellites lancés par type de configuration [140]

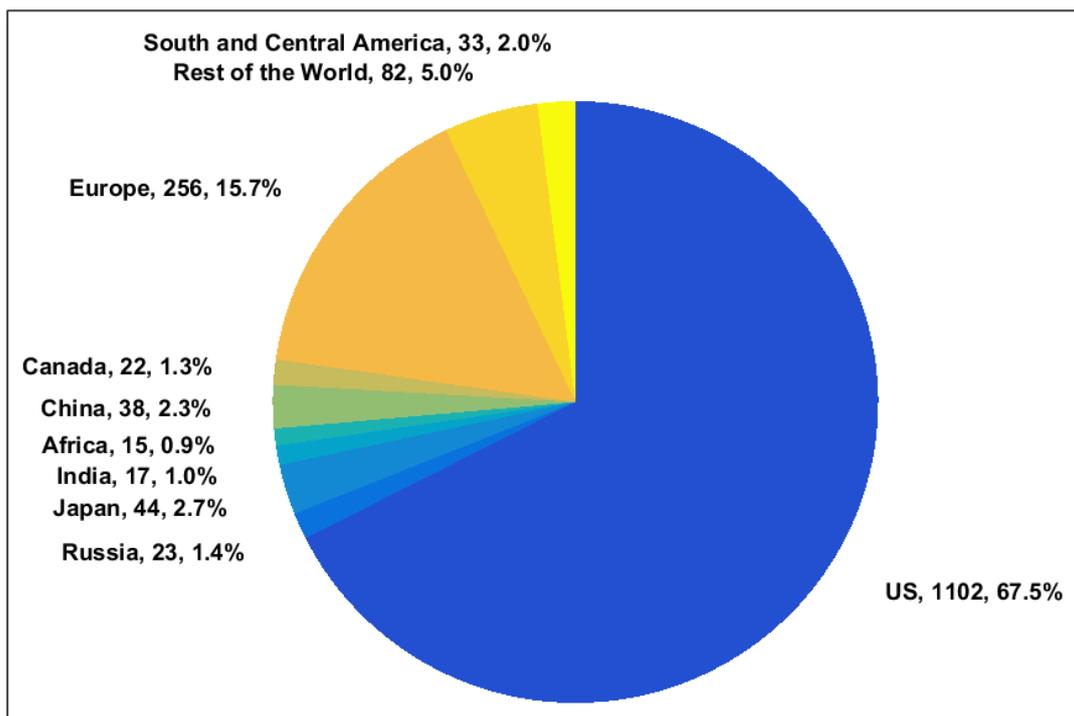


Figure 10 : Répartition géographique des développeurs de nanosatellites [140]

5. PROJET DU NANOSATELLITE UNIVERSITAIRE MASAT1

Il s'agit d'un projet de recherche appliquée intra-universitaire, porté par le laboratoire LISTA de la Faculté des Sciences Dhar El Mehraz de Fès, sous la tutelle du CNRST. Le projet est financé par le Fonds National de Soutien à la Recherche Scientifique et au Développement Technologique (FNSRSDT) sur une durée de 3 ans.

Le projet a pour ambition d'impliquer plusieurs structures de recherches universitaires dans la réalisation des sous-systèmes constituant le segment spatial du premier nanosatellite universitaire marocain, ainsi que les moyens au sol l'accompagnant. La finalité étant de :

- Maîtriser les différentes étapes technologiques de conception et de réalisation des nanosatellites (acquisition de l'expertise dans le domaine).
- Réaliser un modèle ingénieur et un modèle de vol du nanosatellite.
- Mettre en place la ou les stations au sol associées.
- Créer un réseau de laboratoires et de compétences permettant de soutenir par la recherche à forte valeur ajoutée, le tissu innovant dans le domaine du spatial au Maroc.

Le système permettant la mise en œuvre de la mission du projet sera composé d'un segment spatial évoluant en orbite LEO et d'un segment au sol. La charge utile du segment spatial aura pour mission, en plus des télécommunications, le test d'un panneau solaire expérimental et d'un calculateur de bord secondaire à base du circuit émergent qui est le FPGA (Field-Programmable Gate Array).

Dans cette optique, les différentes structures universitaires impliquées dans le projet sont amenées à concevoir, réaliser et tester les sous-systèmes de la plateforme du segment, ainsi que la charge utile, comme le montre la *figure 11* :

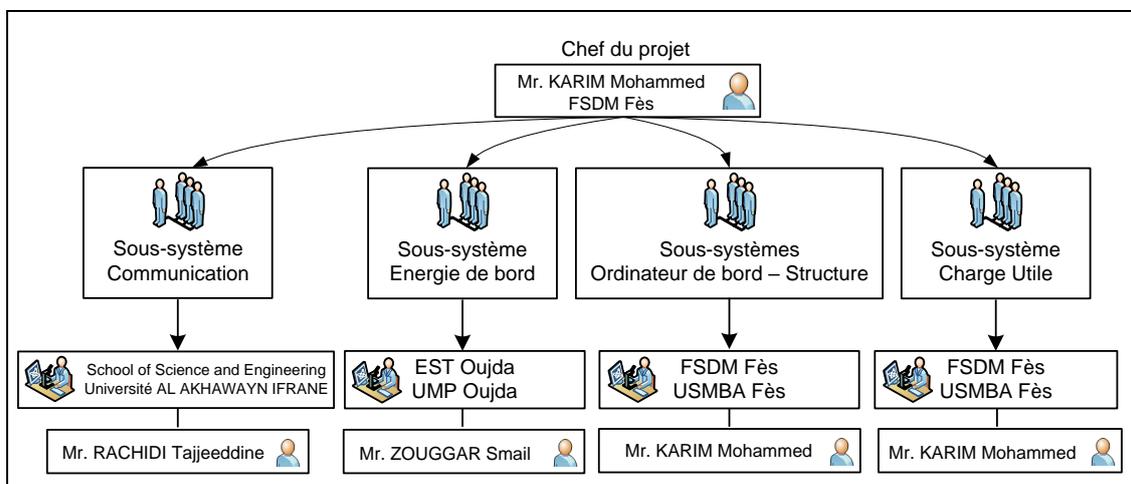


Figure 11 : Les structures universitaires participantes au projet

Pour le segment au sol, on envisage la possibilité de créer plusieurs stations au sol tout en privilégiant le principe de réutilisation et d'adaptation efficace des installations existantes, sans pour autant compromettre le retour scientifique de la mission avec la possibilité d'exploitation du réseau Internet pour faire participer les différentes structures scientifiques participantes au projet.

Puisque le projet constitue la première tentative des universitaires marocains pour développer et lancer un nanosatellite en orbite, la mission aura comme objectif la vérification des exigences fonctionnelles de base nécessaires à tous engin spatial, à savoir :

- Assurer et contrôler l'échange de données entre les sous-systèmes.
- Gérer l'alimentation des différents sous-systèmes.
- Permettre le stockage de données et l'exécution des télécommandes (TC).
- Créer un logiciel de vol qui contrôle le segment spatial.

Afin de clarifier ces objectifs, nous avons fixé quatre niveaux de réussite pour la mission :

Niveau	Description
Minimal	<ul style="list-style-type: none"> ▪ Concevoir et réaliser le segment spatial et le segment au sol de la mission.
Réduit	<ul style="list-style-type: none"> ▪ Réception des signaux provenant du segment spatial de la mission.
Complet	<ul style="list-style-type: none"> ▪ Transfert de données fiables entre le segment spatial et le segment sol dans les deux sens. ▪ Tous les sous-systèmes entièrement opérationnels.
Avancé	<ul style="list-style-type: none"> ▪ Ordinateur de bord secondaire à base de FPGA et panneau solaire expérimental opérationnels

Tableau 3 : Description des niveaux de réussite de la mission

6. MOTIVATION ET OBJECTIFS DE LA RECHERCHE

Selon SpaceWorks Enterprises, le segment des nano/microsatellites à faible coût (1-50kg) devrait connaître une augmentation annuelle de 23,8% sur la période 2014-2020 [13]. Cette tendance est favorisée, comme on l'a vu, par la standardisation de la norme Cubesat qui permet le développement rapide, peu coûteux et rentable de plateformes appropriées aussi bien pour l'apprentissage des connaissances de base sur l'ingénierie spatiale, que pour l'évaluation et le test rapide de nouvelles technologies.

D'autre part, la diversité et la complexité croissante des charges utiles entraînent le besoin de systèmes de contrôle d'attitude plus complexes, ainsi que des exigences élevées en termes de capacité de calcul des plateformes.

Les scientifiques se trouvent alors confrontés, en raison des limites en termes de taille, de masse et de puissance disponible à bord des nanosatellites, devant le défi de développer des solutions embarquées innovantes.

Sur la base de ce contexte, les motivations et les objectifs de cette thèse de recherche résident dans les points suivants :

- Acquérir la technologie et les connaissances de base sur l'ingénierie spatiale, à travers le développement complet des sous-systèmes constituant la plateforme du segment spatial d'un Cubesat au format 1U.
- Fournir, pour les futurs chercheurs marocains dans le domaine spatial, une littérature complète et détaillée concernant l'état de l'art et les étapes de conception des nanosatellites, pour combler le manque qu'il y a dans ce domaine au Maroc.
- Proposer, comme charge utile, une alternative à faible coût, fiable et reconfigurable pour remplacer les architectures matérielles traditionnelles basées sur les microcontrôleurs et les processeurs. Notre approche va consister à concevoir un ordinateur de bord secondaire hautement intégré et reconfigurable selon le principe du System-on-Chip (SoC), en mettant en œuvre la technologie des FPGA type SRAM.

La finalité étant de :

- Démontrer l'utilisation de la technologie FPGA reprogrammable comme un moyen viable permettant de diminuer les temps de conception, d'augmenter la capacité de calcul, et de permettre une reconfiguration flexible en orbite des fonctions matérielles et/ou logicielles de la plateforme embarquée pour, par exemple, prendre en charge des exigences de mission dynamiques ou mettre à jour la configuration du FPGA afin de corriger des erreurs de conception.
- Tester et évaluer, en orbite, des techniques d'atténuation pour assurer une bonne tolérance aux pannes des FPGA type SRAM. En effet, les radiations spatiales et leur effet sur ce type de FPGA constituent le principal défi au déploiement des systèmes reconfigurables dans l'espace, et ils nécessitent la mise en œuvre de techniques de détection, d'isolation et de récupération de défaillance.

7. PLAN DE LA THESE

Notre thèse vise l'étude et la conception de la plateforme et d'un ordinateur de bord secondaire à base de FPGA pour le nanosatellite universitaire MASAT1. Le manuscrit est organisé en sept chapitres :

- Une introduction générale qui permet de décrire le contexte de la recherche, et de donner l'état de l'art du concept Cubesat.
- Le chapitre 1 décrit la phase préliminaire de la recherche dont le but est de réaliser une analyse de la mission du nanosatellite pour déduire les exigences et les paramètres déterminants dans la conception des différents sous-systèmes de segment spatial. Cette analyse s'est basée sur certains standards ECSS (European Cooperation for Space Standardization) de l'ESA, et elle comprend aussi une analyse orbitale suivie d'une étude et d'une évaluation des contraintes spatiales.
- Dans les chapitres 2 à 6, on présente les étapes qui ont menées à la conception de la plateforme du segment spatial. Les travaux ont commencé par choisir une architecture générale sur laquelle s'est basée la conception détaillée des sous-systèmes de la plateforme. Cette partie du rapport de thèse présente aussi les budgets de masse, de puissance et de liaison, ainsi que l'organisation du logiciel de vol qui sera embarqué.
- Le chapitre 7 concerne l'étude et la conception l'ordinateur de bord secondaire. Il commence par la présentation des architectures reconfigurables à base de FPGA, et des considérations de la conception à base de FPGA type SRAM dans les applications spatiales. On décrit ensuite l'architecture matérielle adoptée pour notre ordinateur de bord secondaire reconfigurable à base de FPGA, suivie de l'organisation logicielle qui est basée sur un système d'exploitation temps réel embarqué.

Enfin, une conclusion où on fait un bilan du travail réalisé et on présente les perspectives d'avenir comprenant les étapes qui restent à faire pour mener à bien la mission du nanosatellite universitaire marocain MASAT1.

Chapitre 1

ANALYSE DE LA MISSION DU NANOSATELLITE MASAT1

1. PHASE DE DEVELOPPEMENT D'UNE MISSION SPATIALE

Le cycle de vie d'une mission spatiale progresse typiquement à travers 4 phases [8] :

- **Phase 0 d'étude initiale** : Elle se traduit par une définition large de la mission spatiale, des exigences fonctionnelles et opérationnelles, ainsi que des contraintes.
- **Phase 1 de conception formelle** : Elle se traduit par le développement d'un modèle ingénieur du nanosatellite avec la conception des différents sous-systèmes qui le composent.
- **Phase 2 de production et de déploiement** : Elle se traduit par la réalisation d'un modèle de vol du nanosatellite (développement du matériel et du logiciel de vol), et la mise en place d'une station au sol opérationnelle.
- **Phase 3 des opérations** : Elle se traduit par la mise en orbite du nanosatellite, son suivi, son exploitation et sa maintenance en cours de mission.

Comme le montre la **figure 1-1**, ces phases peuvent être divisées et nommées différemment selon l'agence spatiale choisie comme référence : ESA [15] ou NASA [16]. Chaque phase a une durée dépendante de la portée et des objectifs de la mission ; et elle doit aboutir sur un document livrable (revue) qui évaluera et validera le travail réalisé, et permettra de progresser tout au long du cycle de vie de la mission.

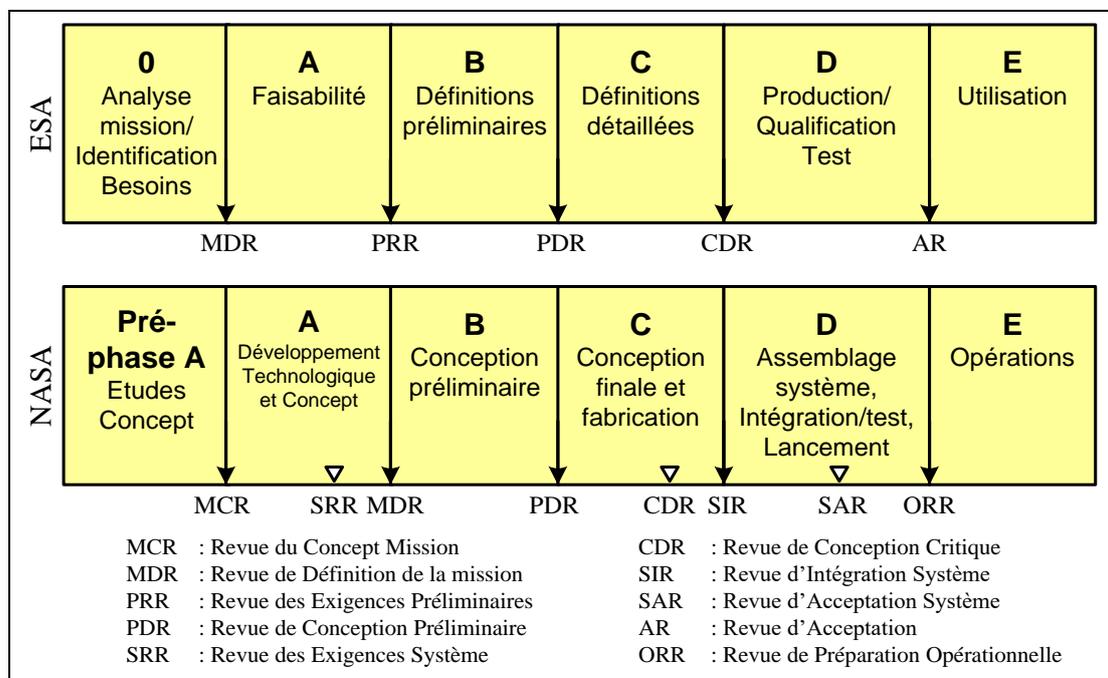


Figure 1-1 : Les phases de développement d'une mission spatiale

Ce chapitre va traiter de la phase initiale d'analyse de la mission dont les activités permettront la définition préliminaire et basique du système en tenant compte de l'objectif, des exigences et des contraintes de la mission.

Comme cadre référentiel pour nos travaux, nous avons partiellement utilisé les standards ECSS de l'ESA, qui fournissent gratuitement [141] les documents références et outils nécessaires pour la gestion de la mission et son ingénierie système.

2. APPLICATION DES STANDARDS ECSS SUR LA MISSION

La norme ECSS constitue une initiative soutenue en Juin 1994 par l'ESA, et conçue pour développer un ensemble unique de normes spatiales cohérentes pour être utilisées par l'ensemble de la communauté spatiale européenne [17].

2.1. Architecture et applicabilité du système ECSS

Pour décrire brièvement l'architecture, la mise en œuvre et les exigences générales des standards ECSS, on va se baser sur le document référence ECSS-S-ST-00C [18].

Les normes ECSS sont organisées sous la forme de trois branches principales correspondantes à la gestion et organisation (Standards ECSS-M), l'ingénierie (Standards ECSS-E), l'assurance produit, qualité (Standards ECSS-Q) et auxquelles s'ajoutent les aspects généraux et systèmes (Standards ECSS-S).

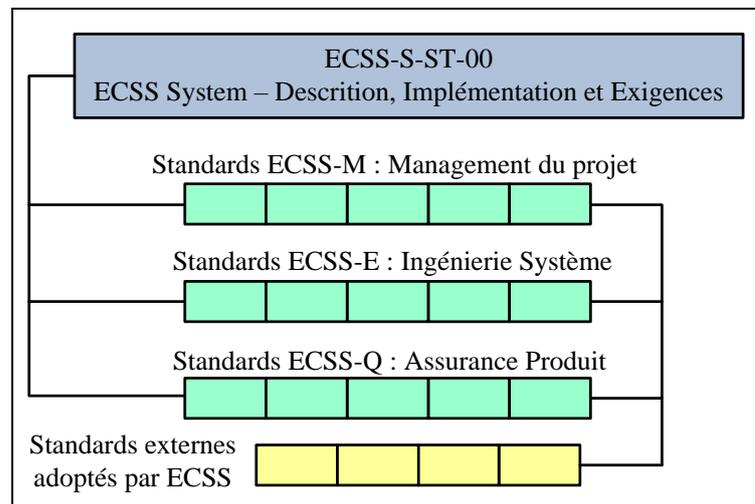


Figure 1-2 : Normes d'utilisation ECSS structurées en Branches [18]

Chaque branche est constituée par un premier niveau de normes, et le cas échéant par des documents complémentaires plus détaillés, des guides facilitant la mise en œuvre des normes de premier niveau, ou des manuels fournissant des informations additionnelles, non normatives comme le montre la *figure 1-3* :

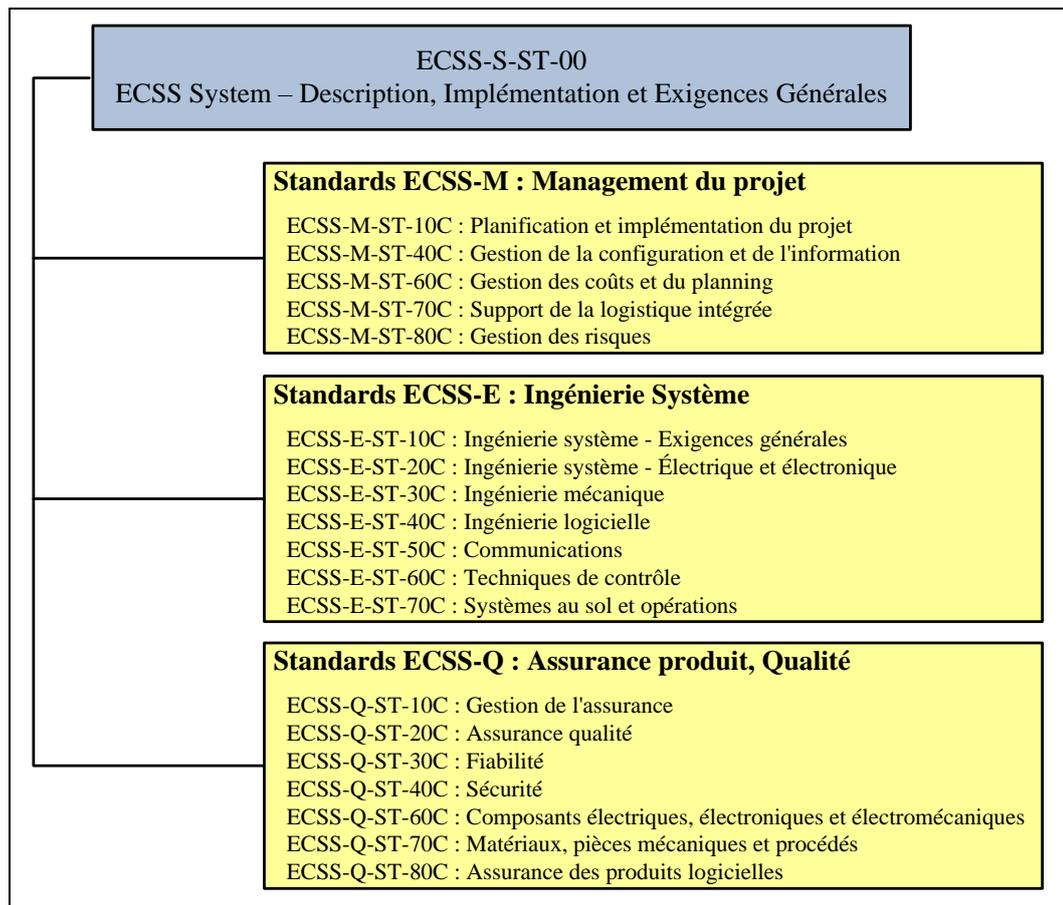


Figure 1-3 : Disciplines du système des normes ECSS [18]

Mener à bien une mission spatiale nécessite la coopération de plusieurs acteurs qui partagent l'objectif commun de fournir un produit répondant aux besoins et exigences de la mission. Tous les acteurs du projet spatial sont, soit un client ou un fournisseur, soit les deux ; et ils sont liés par des contrats commerciaux ou des accords.

Les normes ECSS n'ont pas de statut juridique et ne constituent pas des accords commerciaux ou des contrats, mais elles fournissent les documents utiles pour régler et gérer les relations entre tous les acteurs du projet. L'applicabilité des normes ECSS est précisée dans les documents relatifs aux exigences du projet (PRD : Project Requirements Documents), qui sont inclus dans les contrats et acceptés par les différentes parties.

Un acteur, à tous les niveaux, est responsable de démontrer la conformité des travaux réalisés aux exigences du projet contenues dans les PRD. La conformité au PRD est présentée dans un document d'implémentation (ID). La structure hiérarchique de la *figure 1-4* constitue la chaîne client-fournisseur adoptée au sein du projet MASAT1 :

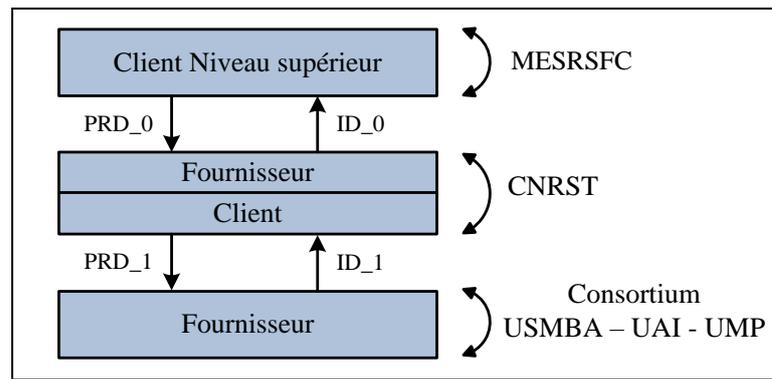


Figure 1-4 : Chaîne Client-Fournisseur du projet MASAT1

2.2. Gestion du projet à l'aide du standard ECSS-M

L'objectif global de la branche management du projet des standards ECSS est de mettre en œuvre un processus pour parvenir à la réussite du projet en termes de coût, de planning et de performances techniques [18].

Dans la phase d'analyse de la mission, nous avons appliqué le standard planification et implémentation du projet ECSS-M-ST-10C qui englobe tout le processus à réaliser pour planifier et exécuter un projet spatial d'une manière coordonnée, efficace et structurée. Ce processus comprend en particulier [19] :

- La définition des phases du projet et le planning permettant de contrôler le progrès des travaux en termes de coût, d'échéances et d'objectifs techniques. Le planning du cycle de vie de la mission MASAT1 et la description des tâches à réaliser et des documents livrables sont donnés à la *figure 1-5*.
- Les structures de décomposition du projet qui constituent un référentiel commun et unique permettant d'identifier les tâches et les responsabilités de chaque acteur. On en distingue en particulier :
 - L'arborescence "Produit", déduite de l'analyse des fonctionnalités du système spatial, et qui permet de le diviser en niveaux successifs de produits matériels et logiciels.
 - La structure de répartition du travail WBS (Work Breakdown Structure) qui est la principale structure pour gérer un projet spatial et fournir un cadre de travail pour les différents acteurs. Cette structure est dérivée de l'arborescence "Produit", et elle divise le projet en plusieurs tâches organisées en fonctions des travaux à réaliser. Chaque bloc de travail peut à son tour être divisé en sous-tâches et ainsi de suite.

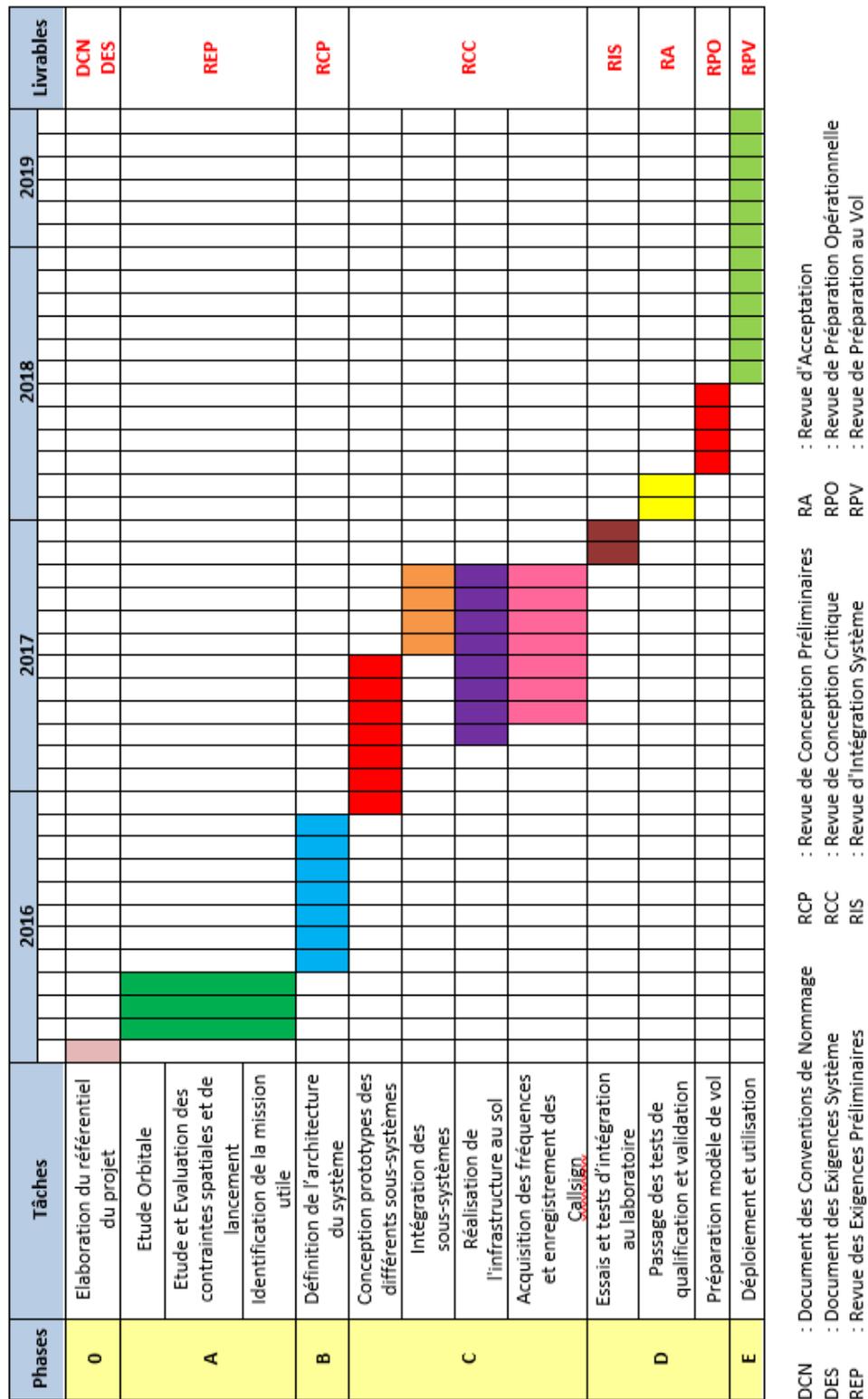


Figure 1-5 : Tâches à effectuées et Planning de la mission

La *figure 1-6* et la *figure 1-7* donnent respectivement l'arborescence "Produit" et la structure de répartition du travail relatives à notre projet :

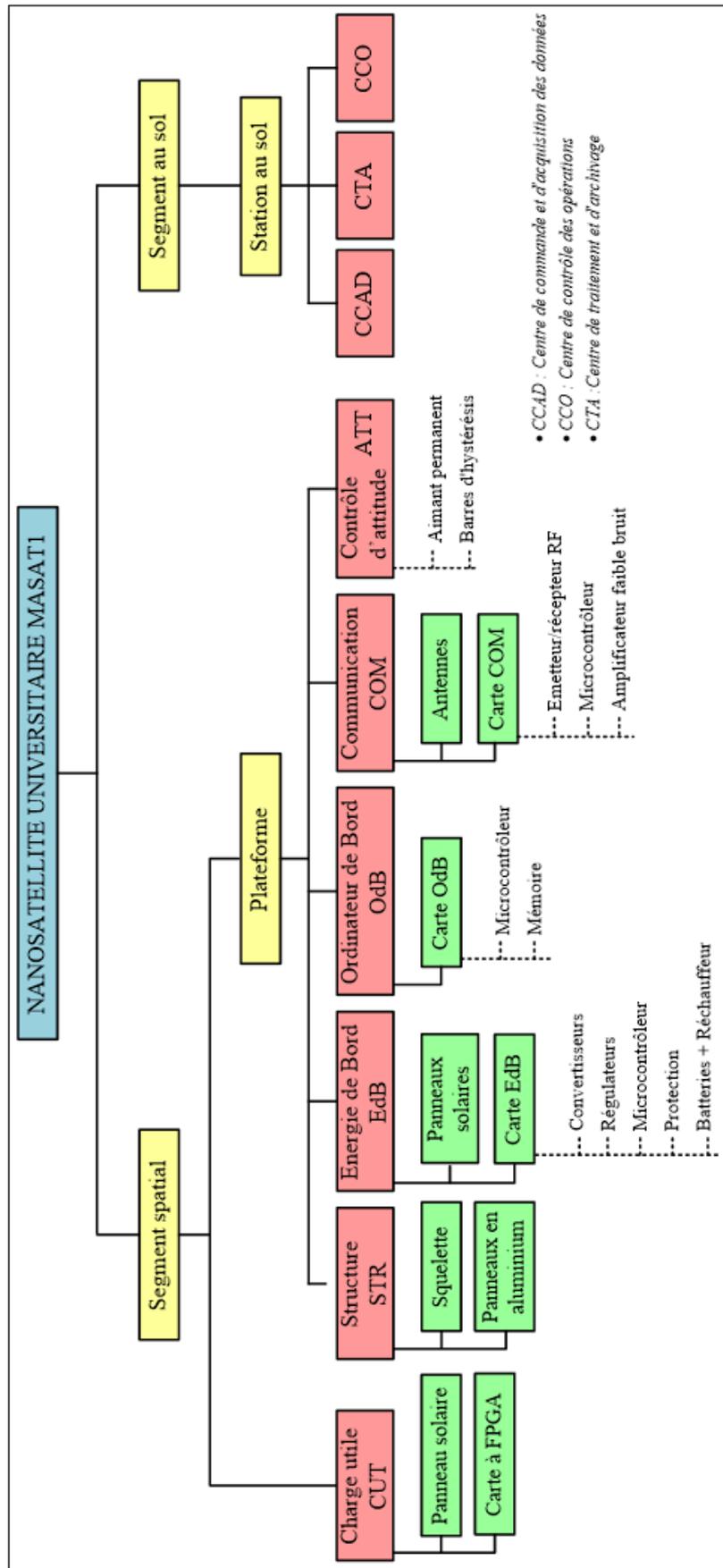


Figure 1-6 : Arborescence "Produit" du projet MASATI

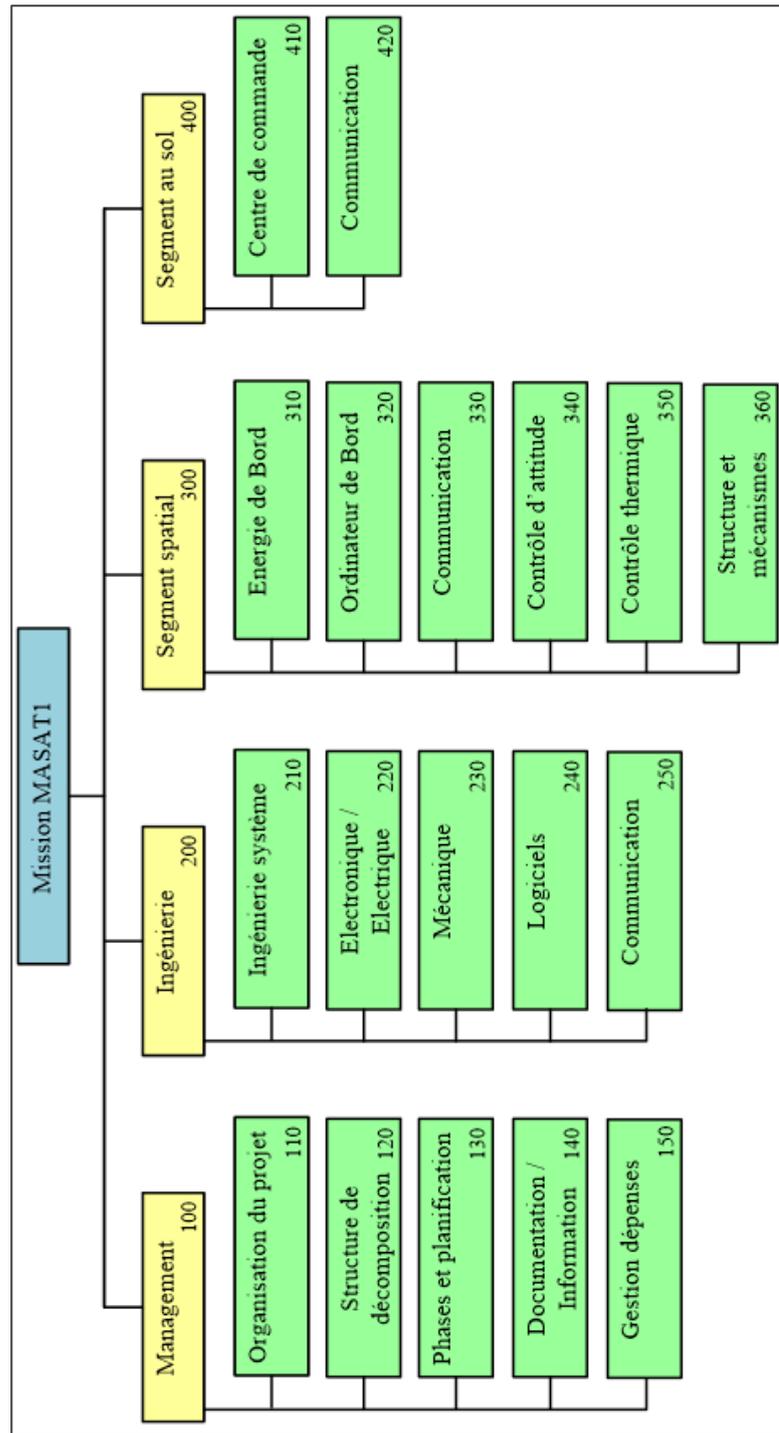


Figure 1-7 : Structure de répartition du travail (WBS)

La structure de répartition du processus WBS est la base du processus d'organisation et de documentation du projet en fournissant aux documents, une numérotation correspondante directement à celle donnée aux différents blocs de travaux de la structure. Enfin, la définition détaillée des différents blocs de travaux du WBS permet de faciliter, d'une part l'intégration des sous-systèmes, et d'autre part la contribution apportée par des partenaires internationaux.

2.3. Détermination des exigences de la mission à l'aide du standard ECSS-E

La branche d'ingénierie système ECSS-E fournit un processus d'ingénierie applicable aux systèmes spatiaux. Elle est divisée en disciplines, couvrant les différents aspects techniques des systèmes et produits spatiaux, y compris [18] :

- ECSS-E-10 : Ingénierie système
- ECSS-E-20 : Génie électrique et électronique
- ECSS-E-30 : Ingénierie mécanique
- ECSS-E-40 : Génie logiciel
- ECSS-E-50 : Systèmes de Communications
- ECSS-E-60 : Technique de contrôle
- ECSS-E-70 : Stations au sol et opérations

Plusieurs standards ECSS-E ont été étudiés au cours du processus de conception du nanosatellite universitaire, et ils seront cités au fur et à mesure de l'avancement de ce rapport de thèse.

Dans ce qui suit, on va présenter le document des exigences système (DES) qui est l'un des documents référence qui sera la base sur laquelle la conception finale du système sera validée, et le projet accepté. C'est un document permettant d'exprimer "l'énoncé de la mission", ainsi que les contraintes et l'environnement correspondants ; ceci en termes d'exigences qui ne doivent référer à aucune solution technique spécifique mais plutôt permettre par la suite, dans la phase A du projet, de proposer les meilleures solutions techniques et logiciels [21].

Parmi les standards d'ingénierie système que nous avons utilisé pour élaborer le DES de la mission MASAT1, on peut citer :

- ECSS-E-ST-10C : Ingénierie Système – Exigences générales [20]
- ECSS-E-ST-10-06C : Spécifications des exigences techniques [21]
- ECSS-E-ST-20C : Ingénierie système - Électrique et électronique [22]

Les exigences et fonctionnalités du DES sont référencées et étiquetées pour identifier leur type ou catégorie. Les identificateurs utilisés répondent à la syntaxe suivante :

DDDD – nnnn

Avec : DDDD : représente la catégorie ou le type d'exigence
nnnn : représente le numéro unique de l'exigence.

Le *tableau 1-1* résume les catégories d'identificateurs qui ont été utilisées pour les exigences, en respectant le standard ECSS-E-ST-10-06C relatif aux spécifications des exigences techniques [21] :

Identificateur	Type d'exigence	Description
CONC	Exigences de conception	Exigences liées à la conception imposée et aux normes d'implémentation (interchangeabilité, sécurité ou marges).
CONF	Exigence de configuration	Exigences indiquant la composition ou l'organisation du système (différents modules et sous-systèmes).
FONC	Exigences fonctionnelles	Exigences qui indiquent ce que doit faire le système afin de se conformer aux exigences de l'énoncé de la mission
INTF	Exigences d'interfaçage	Exigences relatives aux caractéristiques d'interconnexion entre les sous-systèmes (Interfaces physiques, thermiques, électriques ou protocoles)
OPER	Exigences opérationnelles	Exigences relatives à la capacité de fonctionnement du système et les événements auxquels il doit répondre (autonomie, contrôle de la contingence).
PHYS	Exigences physiques	Exigences qui établissent les conditions aux limites pour assurer la compatibilité physique : caractéristiques mécaniques, poids, limites dimensionnelles, etc.
CONT	Exigences des contraintes de l'environnement	Exigences relatives à l'environnement du système au cours de son cycle de vie.
FMST	Exigences de fiabilité, maintenance, sécurité et test	Exigences relatives aux activités qui couvrent l'assurance du produit.

Tableau 1-1 : Identificateurs et types d'exigences définies dans le DES

Le *tableau 1-2* donne un aperçu des principales exigences générales du segment spatial de la mission MASAT1 [71] :

Exigences opérationnelles	
[DES] OPER-0010	Un profil des opérations doit être défini pour couvrir toutes les phases de la mission du lancement jusqu'à la fin de la mission.
[DES] OPER-0030	Les opérations de contrôle d'attitude doivent être réduites au minimum, ou utiliser un contrôle d'attitude passif dans les limites des exigences de la mission.
[DES] OPER-0040	La supervision du segment spatial doit permettre d'évaluer son état par le segment au sol à chaque fois que la communication peut être établie. Cette supervision doit être autonome et ne dépendre d'aucun mode de fonctionnement spécifique.
[DES] OPER-0070	Le segment spatial doit retarder le déploiement des antennes jusqu'à ce que le segment spatial atteigne des conditions d'alimentation optimales en énergie.
Exigences fonctionnelles	
[DES] FONC-0100	Le segment spatial doit fonctionner pour une durée de 12 mois au moins.
[DES] FONC-0105	Le segment spatial doit continuellement envoyer un signal de balise.
[DES] FONC-0110	Le segment spatial doit accepter, stocker et traiter les commandes envoyées par le segment au sol.
[DES] FONC-0160	Il doit être possible de déclencher une réinitialisation partielle ou complète du segment spatial depuis le segment au sol.
Exigences de configuration	
[DES] CONF-0200	La configuration du segment spatial doit être conçue pour être compatibles avec les contraintes du lanceur dédié avec au moins deux référence de lanceurs.
[DES] CONF-0210	Le nombre d'éléments déployables du segment spatial doit se limiter à l'antenne
[DES] CONF-0230	La conception de la configuration du segment spatial doit prendre en compte les effets de la comptabilité Radio fréquence et électromagnétique (CRF / CEM).
[DES] CONF-0240	La configuration du segment spatial doit faciliter le contrôle d'attitude en minimisant les moments d'inertie et en assouplissant la structure.
Autres types d'exigences	
[DES] PHYS-0250	La masse totale du segment spatial, y compris la charge utile et les marges, doit être égale à 1,33Kg et compatible avec les performances du lanceur [10].
[DES] FMST-0280	Le segment spatial doit être conçu pour éviter qu'un point singulier de défaillance provoque l'arrêt définitif des fonctionnalités. La redondance ou une architecture distribuée peuvent être une solution pour réduire l'exposition aux défaillances singulières.
[DES] FMST-0350	L'OdB et la COM doivent être capables de stocker une trace de l'exécution du programme dans une mémoire non-volatile, et d'envoyer cette trace via l'interface de débogage et/ou de communication.
[DES] FMST-0360	Seules les commandes non corrompues et envoyées par le segment au sol, doivent être exécutées.
[DES] FMST-0370	Un programme défectueux ne doit pas affecter la fonctionnalité de base du système.

Tableau 1-2 : Aperçu des exigences générales du segment spatial de la mission [71]

3. ETUDE ORBITALE

Le choix de l'orbite est une étape importante dans chaque mission spatiale car il influence fortement la conception des sous-systèmes ATT, EdB, COM et THER. Il dépend habituellement des exigences de missions et donc il est spécifique pour chaque satellite.

Pour le projet MASAT1, ni le mode ni la date du lancement ne sont encore définis. Et même si les paramètres orbitaux sont fixés pour un lanceur donné, ils pourraient changer car les Cubesats ne constituent qu'une charge secondaire. Notre mission doit alors prévoir les changements potentiels dans les éléments orbitaux et préparer une analyse de la mission qui prend en compte plusieurs scénarios possibles.

En résumé, la première tâche de la phase A de la mission (voir *figure 1-5*) consiste à réaliser une étude orbitale permettant de :

- Maitriser la dynamique orbitale et évaluer les effets des perturbations orbitales sur les spécifications d'orbite du nanosatellite.
- Sélectionner les orbites optimales pour que le nanosatellite accomplisse sa mission.
- Fournir les paramètres déterminants dans la conception des différents sous-systèmes du segment spatial, comme :
 - La couverture terrestre et le temps de communication pour le sous-système COM.
 - Les périodes d'éclipse, ainsi que le pourcentage de profondeur de déchargement des batteries pour le sous-système EdB.
 - L'énergie potentiellement fournit par les panneaux solaires pour valider les scénarios opérationnels de la mission.

3.1. Etat de l'art de la dynamique orbitale

3.1.1. Référentiels utilisés

a. Référentiel géocentrique

Ce système a pour origine le centre de la terre et il est généralement noté IJK :

- Le plan fondamental est le plan équatorial de la Terre.
- L'axe I est orienté vers l'équinoxe vernal, l'axe J est 90° à l'est dans le plan équatorial et l'axe K pointe vers le pôle Nord.
- Ce système de coordonnées n'est pas en rotation et il est supposé être fixe dans l'espace. La terre tourne autour de l'axe Z (axe Nord/Sud).

En anglais, il est appelé le système de coordonnées ECI (Earth-centered inertial).

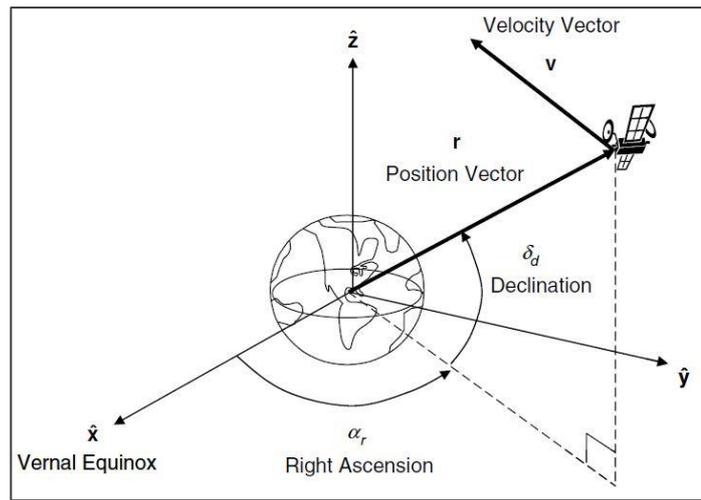


Figure 1-8 : Système de coordonnées du référentiel géocentrique [25]

b. Référentiel périfocales

Dans ce système, le plan fondamental est le plan orbital du satellite, et l'origine est le centre de la terre. Il est généralement repéré par PQW :

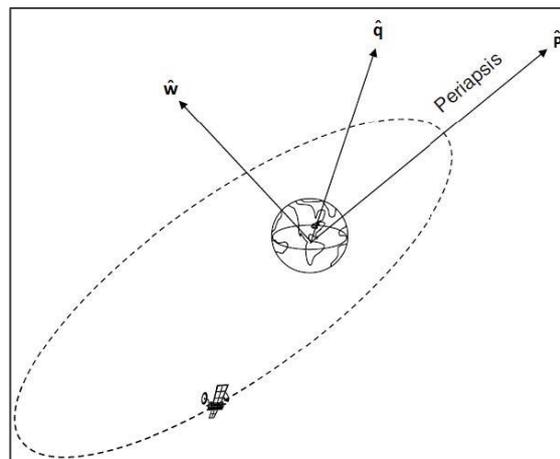


Figure 1-9 : Système de coordonnées du référentiel périfocal [25]

- Le plan fondamental est le plan orbital du satellite.
- L'axe P est orienté vers le périégée, l'axe W est perpendiculaire au plan fondamental dans la direction du vecteur moment cinétique orbital et l'axe Q complète la configuration orthogonale dans la direction du mouvement du satellite.
- Ce système de coordonnées ne tourne pas avec le satellite, mais il maintient toujours une orientation vers le périégée et il est donc mieux adapté pour la description de l'orbite avec un centre bien défini.

3.1.2. Le temps

Le temps est une dimension fondamentale et critique en astrodynamique car les objets se déplacent à des vitesses très élevées. Le système de temps conventionnel (année / mois / jour) n'est pas adapté à l'implémentation des simulations numériques sur ordinateur.

On utilise une autre référence temporelle qui est le calendrier Julien (JD : Julian Date) qui consiste à compter le nombre de jours et fraction de jours écoulés depuis une date conventionnelle fixée au 1^{er} janvier -4712 à 12 heures (4713 av. J.C). Pour trouver la date JD à partir d'une date et d'une heure connues entre le 1^{er} mars 1900 et le 28 février 2100, on peut utiliser l'équation suivante [23] :

$$JD = 367y - \text{tronc} \left(\frac{7}{4} \left(y + \text{tronc} \left(\frac{m+9}{12} \right) \right) \right) + \text{tronc} \left(\frac{275m}{9} \right) + d + 1721013.5 + \frac{1}{24} \left(\frac{1}{60} \left(\frac{sec}{60} + min \right) + hr \right) \quad (2.1)$$

Le JD permet de préserver dans une seule variable, toutes les informations concernant l'année, le mois, le jour, l'heure, les minutes et les secondes d'une date ; ce qui est particulièrement intéressant pour les applications informatiques.

3.1.3. Les lois de Kepler

Johannes Kepler (1571-1630) était un mathématicien qui a utilisé les données d'observation de Tycho Brahe (1546- 1601) pour décrire les orbites planétaires elliptiques autour du soleil. Il publia ses deux premières lois du mouvement planétaire en 1609 et la troisième loi en 1619 [23].

Ces trois lois du mouvement planétaire de Kepler sont également applicables aux satellites orbitant autour de la terre :

- Première loi : les planètes décrivent une ellipse dont le Soleil occupe l'un des foyers.
- Deuxième loi : le rayon Soleil-planète balaie des aires égales pendant des intervalles de temps égaux.
- Troisième loi : le carré de la période de révolution est proportionnel au cube du demi grand-axe de l'orbite.

3.1.4. Problème de deux corps

La 2^{ème} loi du mouvement de Newton, appliquée à un système de masse constant et, combinée à la loi de gravitation, fournit la base mathématique pour l'analyse de la dynamique orbitale, spécialement quand c'est combiné avec les lois de Kepler [23].

Pour deux corps T (Terre) et S (Satellite) de masse respectives m_T et m_S en interaction gravitationnelle (mutuelle), le mouvement est rapporté à un référentiel galiléen (voir *figure 1-10*). Le principe fondamental de la dynamique (deuxième loi de Newton) de chacun des corps, combiné à la loi de gravitation, donne [23] :

$$m_T \frac{d^2 \vec{R}_1}{dt^2} = +Gm_T m_S \frac{\vec{r}}{|\vec{r}|^3} \quad (2.2)$$

$$m_S \frac{d^2 \vec{R}_2}{dt^2} = -Gm_T m_S \frac{\vec{r}}{|\vec{r}|^3} \quad (2.3)$$

Avec G la constante universelle de gravitation égale à $6,670 \cdot 10^{-11}$ ($N \cdot m^2 kg^{-2}$) et r la position du satellite par rapport à la terre.

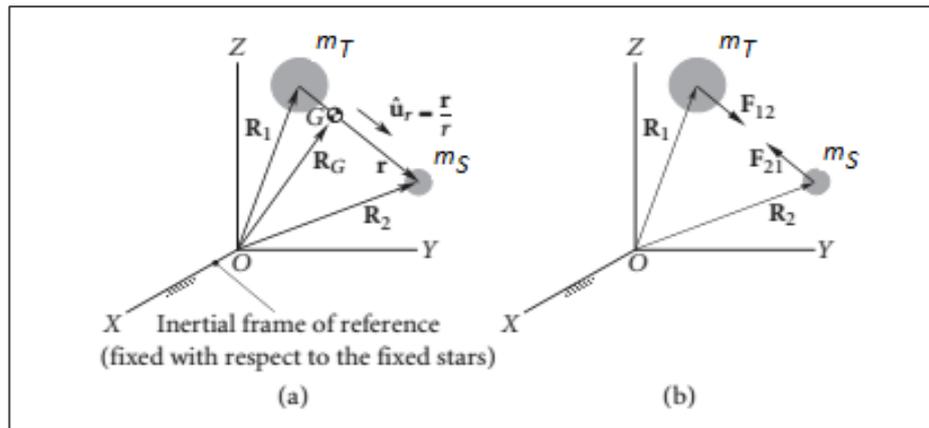


Figure 1-10 : (a) 2 masses situées dans un référentiel galiléen (b) Corps libre [24]

En éliminant m_T de l'équation (2.2) et m_S de l'équation (2.3) et en soustrayant les deux résultats, on obtient l'équation différentielle qui décrit le mouvement du satellite par rapport à la terre :

$$\ddot{\vec{r}} = -\mu \frac{\vec{r}}{r^3} \quad (2.4)$$

$\mu = Gm_T$: Paramètre gravitationnel

Cette équation, appelée équation du mouvement de deux corps, est l'équation relative du mouvement d'un vecteur de position du satellite lorsqu'il est en orbite autour de la Terre. Les hypothèses faites lors de sa dérivation sont [23] :

- La masse du satellite est négligeable par rapport à celle du corps attirant, la Terre.
- Les deux corps sont supposés sphériques ce qui permet de les étudier comme si leur masse est concentrée dans leur centre de masse.
- Seule la force gravitationnelle (mutuelle) qui agit sur le système le long de la ligne reliant les centres de masse des 2 corps.

3.1.5. Equation de la trajectoire (1ère loi de Kepler)

La résolution de l'équation (2.4) du mouvement d'un satellite par rapport à la terre donne l'équation de la trajectoire du satellite qui représente l'amplitude du vecteur position \vec{r} en fonction de l'emplacement en orbite [23] :

$$r = \frac{a \cdot (1 - e^2)}{1 + e \cos \theta} \quad (2.5)$$

C'est l'équation d'une section conique (*figure 1-11*) dont l'un des foyers est la terre, avec a le demi-grand axe, e l'excentricité qui détermine la taille et le type de l'orbite et θ l'anomalie vraie (l'angle entre le vecteur excentricité \vec{e} et le vecteur position \vec{r}) :

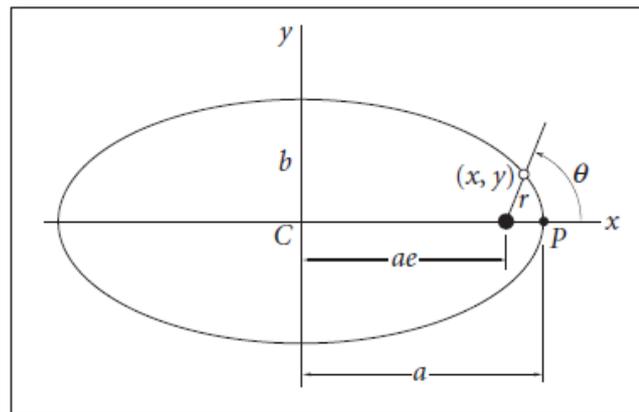


Figure 1-11 : Paramètres de l'orbite Képlérien en coordonnées cartésiennes [24]

3.1.6. Détermination de l'orbite à l'aide des paramètres orbitaux

Lors de la résolution de l'équation différentielle (2.4) du mouvement d'un satellite, six constantes d'intégration (conditions initiales de position et de vitesse relatives à l'injection du satellite dans son orbite) sont nécessaires pour la solution. Ces six paramètres, appelés éléments orbitaux classiques, permettent de caractériser l'orbite du satellite dans un référentiel fixe (la taille, la forme et l'orientation de l'orbite), et de donner, en plus, l'emplacement du satellite à tout moment dans l'orbite (voir *figure 1-12*).

Les deux paramètres qui définissent la trajectoire dans le plan orbital sont :

- **Demi-grand axe "a"** qui décrit la taille de l'orbite. Il représente la moitié de la distance entre l'apogée et le périhélie. Pour une orbite circulaire, le demi-grand axe est le rayon de l'orbite (altitude).
- **Excentricité "e"** qui décrit la forme de l'orbite. C'est le quotient de la distance entre les foyers et la longueur de demi-grand axe. Une excentricité nulle indique une orbite circulaire.

Les trois paramètres qui définissent l'orientation du plan orbital dans l'espace sont :

- **Inclinaison "i"** qui indique l'angle d'inclinaison du plan orbital par rapport au plan équatorial. Cet angle est mesuré dans le sens antihoraire au point dans lequel le satellite traverse le plan équatorial vers le nord dans son orbite (nœud ascendant) tout en regardant la terre :
 - $0^\circ < i < 180^\circ$: *une orbite équatoriale* qui réside dans le plan équatorial.
 - $i = 90^\circ$: *une orbite polaire* qui passe au-dessus des pôles.
 - $0^\circ < i < 90^\circ$: *une orbite prograde* qui orbite dans le même sens que la terre.
 - $90^\circ < i < 180^\circ$: *une orbite rétrograde* qui orbite dans sens inverse de la terre.
- **Ascension droite du nœud ascendant (RAAN) " Ω "** qui mesure la rotation du plan orbital par rapport à la terre. C'est l'angle entre la direction vernale et la direction du nœud ascendant mesurée dans un plan de référence.
- **L'argument du périhélie " ω "** qui mesure l'angle dans le plan orbital, entre le nœud ascendant et le périhélie (point de référence des orbites elliptique), dans la direction de la rotation du satellite.

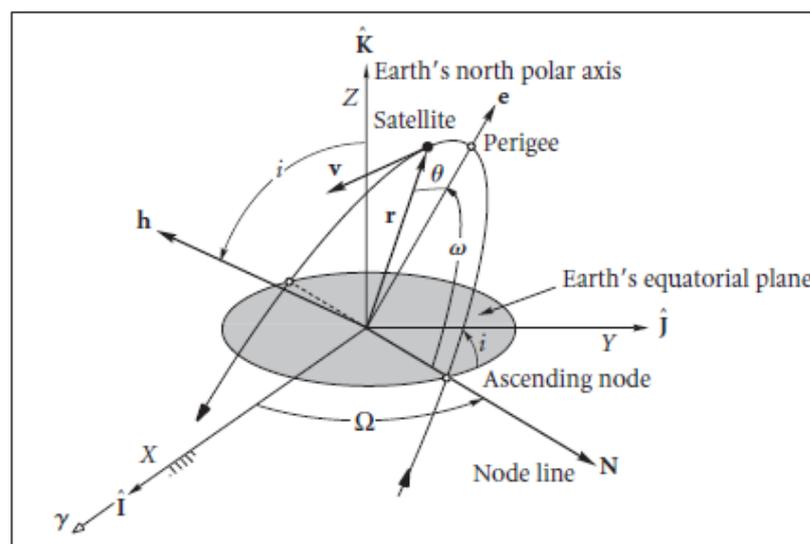


Figure 1-12 : Les éléments orbitaux dans le référentiel géocentrique [24]

Le dernier paramètre est l'**anomalie vraie "** θ **"** qui permet de localiser le satellite dans son orbite. C'est l'angle décrivant le satellite dans son orbite par rapport au périégée et c'est le seul paramètre qui varie dans le temps.

A tout instant, les éléments orbitaux (a , i , e , ω , Ω , θ) contiennent les mêmes informations que les vecteurs de position \vec{r} et de vitesse \vec{v} (voir **figure 1-12**). Il est donc possible de convertir un ensemble de constante vers l'autre.

a. Conversion du vecteur d'état \vec{r} et \vec{v} vers les éléments orbitaux

La conversion du vecteur d'état du satellite dans un référentiel géocentrique vers les éléments orbitaux est possible en connaissant les vecteurs \vec{e} , \vec{h} et \vec{N} qui caractérisent entièrement le mouvement du satellite [23] [24], comme le montre la **figure 1-12** :

- $\vec{h} = \vec{r} \wedge \vec{v}$: le moment angulaire du satellite.
- $\vec{N} = \vec{K} \wedge \vec{h}$: le vecteur de la ligne des nœuds.
- $\vec{e} = \frac{1}{\mu} \left[(|\vec{v}|^2 - \frac{\mu}{r}) \cdot \vec{r} - (\vec{r} \cdot \vec{v}) \cdot \vec{v} \right]$: le vecteur d'excentricité.

Par conséquent, tous les éléments orbitaux se dérivent de ces vecteurs avec des corrections simples des quadrants :

$$a = \frac{h^2}{\mu} \frac{1}{1 - e^2} \quad (2.6) \quad \cos i = \frac{h_z}{h} \quad (2.7) \quad \cos \omega = \frac{\vec{N} \cdot \vec{e}}{N \cdot e} \quad (2.8)$$

$$\cos \Omega = \frac{N_x}{N} \quad (2.9) \quad \cos \theta = \frac{\vec{e} \cdot \vec{r}}{e \cdot r} \quad (2.10)$$

b. Conversion des éléments orbitaux vers le vecteur d'état \vec{r} et \vec{v}

L'approche pour trouver le vecteur d'état dans un référentiel géocentrique inertiel à partir des éléments orbitaux consiste à trouver une représentation du vecteur d'état dans le référentiel local. Après quoi, la transformation vers le référentiel géocentrique inertiel se fait par l'intermédiaire du référentiel péri-focale [23].

Les vecteurs d'état \vec{r} et \vec{v} peuvent être déterminés par les équations suivantes :

$$\mathbf{r}(a, e, i, \Omega, \omega, \theta) = \frac{a(1-e^2)}{1+e \cdot \cos \theta} \begin{bmatrix} c_{\theta+\omega} c_{\Omega} - c_i s_{\theta+\omega} s_{\Omega} \\ c_i c_{\Omega} s_{\theta+\omega} + c_{\theta+\omega} s_{\Omega} \\ s_i s_{\theta+\omega} \end{bmatrix} \quad (2-11)$$

$$\mathbf{v} = \dot{\mathbf{r}}(a, e, i, \Omega, \omega, \theta) = \sqrt{\frac{\mu}{a(1-e^2)}} \begin{bmatrix} -c_{\Omega} s_{\theta+\omega} - s_{\Omega} c_i c_{\theta+\omega} - e(c_{\Omega} s_{\omega} + s_{\Omega} c_{\omega} c_i) \\ c_{\Omega} c_i c_{\theta+\omega} - s_{\Omega} s_{\theta+\omega} - e(s_{\Omega} s_{\omega} - c_{\Omega} c_{\omega} c_i) \\ s_i (c_{\theta+\omega} + e c_{\omega}) \end{bmatrix} \quad (2.12)$$

Avec : $s = \sin$ et $c = \cos$

3.2. L'orbite du nanosatellite MASAT1

L'identification des orbites potentielles pour la mission MASAT1 a été réalisée sur la base des paramètres déterminants suivants :

- **Altitude** : Le choix s'est porté sur une orbite LEO située entre 400km et 700km pour les raisons suivantes :
 - La durée fonctionnelle de la mission qui est au minimum de 12 mois et qui doit être au maximum de 25 ans pour être conforme au code de conduite pour la réduction des débris spatiaux [35].
 - Les perturbations orbitales matérialisées par la trainée atmosphérique pour les altitudes inférieures à 400km, et les radiations causées par la ceinture intérieure de van Allen pour les altitudes supérieures à 800km.
 - L'orbite de la Station Spatiale Internationale (ISS) qui se situe entre 330 km en périégée et 400 km en apogée.
 - 58% des missions cubesat avec un contrôle d'attitude passif évoluent à une altitude comprise entre 400 km et 700 km [34].
- **Inclinaison** : Le choix s'est porté sur une inclinaison supérieure aux latitudes des stations au sol pour avoir des opportunités de communication entre le segment sol et le segment spatial [8]. Les latitudes des trois stations au sol de la mission sont : Ifrane (31,4°), Fès (34,2°), et Rabat (34°).
- **Excentricité** : La plupart des opportunités du lancement des Cubesats, comme charge secondaire des lanceurs, sont des orbites circulaires [34], c'est-à-dire avec une excentricité $e \cong 0$.

Les paramètres orbitaux des différents scénarios possibles identifiés pour la mission MASAT1 sont résumés dans le *tableau 1-3* :

Paramètre orbital	Valeur
Altitude (h)	400 - 500 - 650
Inclinaison (i)	40° - 65° - 90°
Excentricité (e)	0 - 0.013
RAAN (Ω)	A déterminer
Argument du périégée (ω)	A déterminer
Anomalie vraie (θ)	A déterminer

Tableau 1-3 : scénarios orbitaux de la mission MASAT1

Pour avoir une idée de l'orbite képlérienne du nanosatellite MASAT1, nous avons utilisé le logiciel Satellite Tool Kit (STK) pour représenter, sur une journée, l'empreinte au sol du satellite (*figure 1-13*) et une vue tridimensionnelle de l'orbite (*figure 1-14*) :

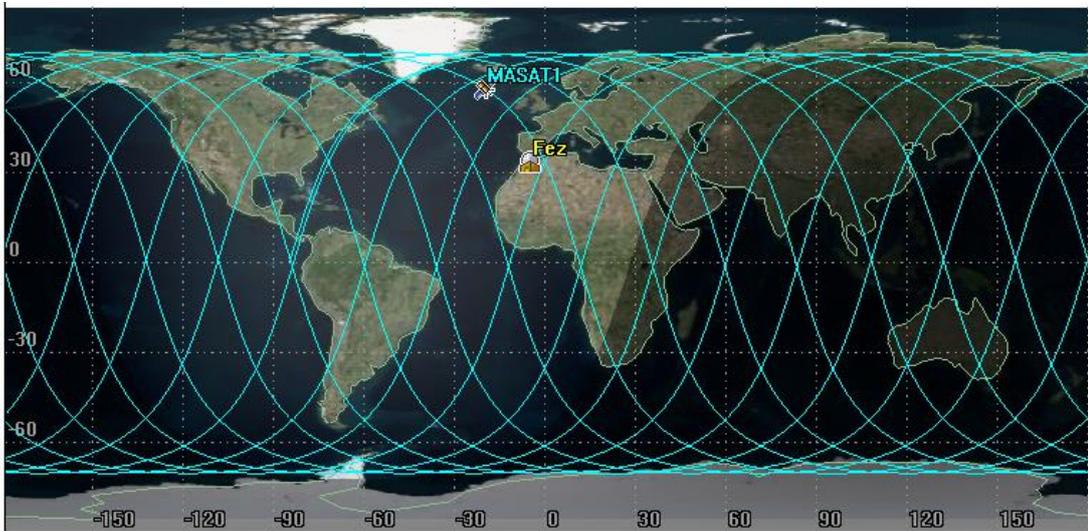


Figure 1-13 : Empreinte du nanosatellite MASAT1 au sol
[STK : altitude 650km – inclinaison 70° - excentricité 0.013]

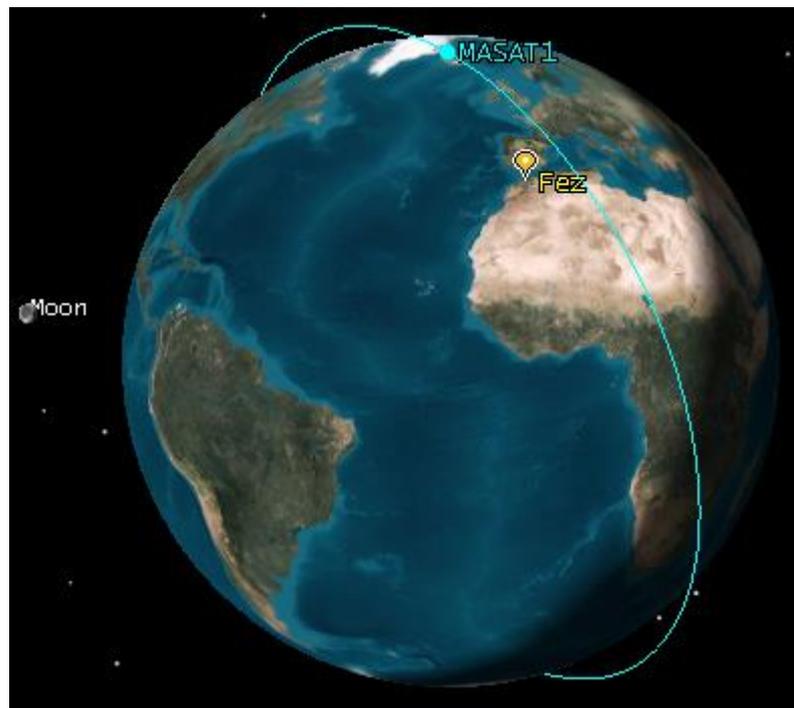


Figure 1-14 : Orbite inertielle 3D du nanosatellite MASAT1
[STK : altitude 650km – inclinaison 70° - excentricité 0.013]

3.3. Perturbations orbitales

L'étude précédemment réalisée, à l'égard des hypothèses considérées (§ 3.1.4), était une approximation du mouvement d'un satellite orbitant autour de la terre. Dans la réalité, la masse de la terre n'est pas distribuée sphériquement et le satellite subit plusieurs forces autres que la force gravitationnelle.

En conséquence, le satellite dévie de son orbite Képlérienne idéale et le plan orbital se déplace dans le référentiel galiléen, et cette perturbation orbitale augmente régulièrement avec le temps [26].

Les perturbations dans les éléments képlériens peuvent changer avec l'altitude, et vu que notre mission sera localisée dans une orbite LEO, les quatre forces perturbatrices dominantes agissant sur le satellite seront : l'aplatissement de la terre, la traînée atmosphérique, les perturbations d'un troisième corps matérialisé par le soleil et la lune, et enfin la pression du rayonnement solaire [24] [26].

La *figure 1-15* montre les effets des différentes forces perturbatrices (exprimées en termes d'accélération), en fonction de l'altitude h :

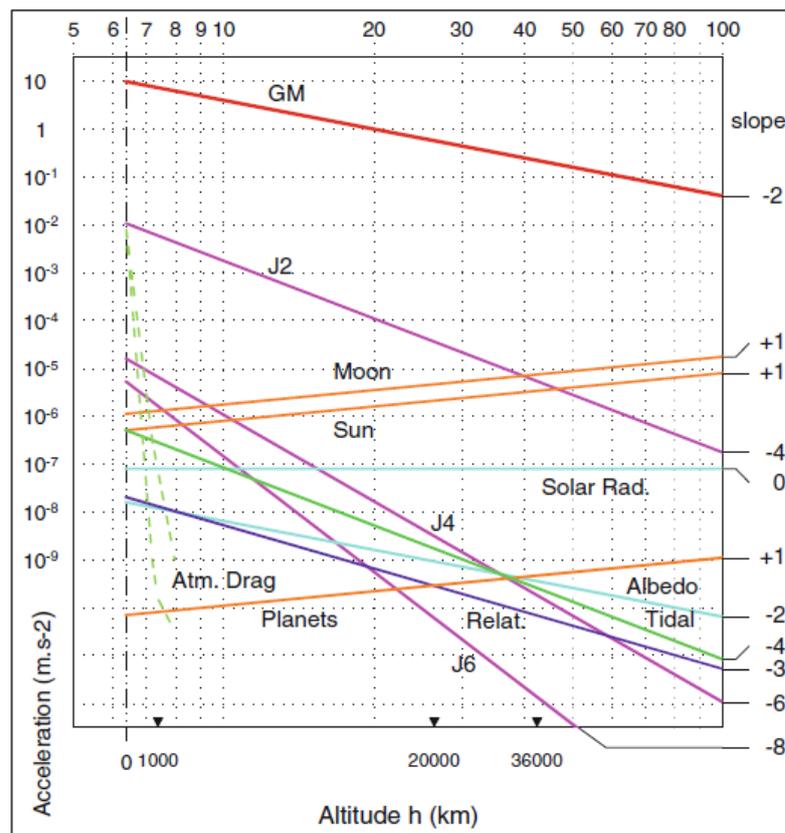


Figure 1-15 : Les accélérations perturbatrices en fonction de l'altitude [26]

Pour la suite de l'étude, on a négligé les perturbations dues aux radiations solaires à cause des contraintes de surfaces imposées par la norme Cubesat [10], hypothèse confirmée par la courbe "Solar Rad" de la *figure 1-15*.

Avec la prise en compte des perturbations orbitales, et selon la formulation de Cowell, l'équation du mouvement du satellite (2.4) permettant de déterminer son orbite, devient [23] [26] [27] :

$$\begin{cases} \ddot{\vec{r}} = \overline{\text{grad}}\left(\frac{\mu}{r}\right) + \vec{f}_{ap} + \vec{f}_{s,l} + \vec{f}_{ad} & (2.13) \\ \vec{r}(t=0) = \vec{r}_0 ; \dot{\vec{r}}(t=0) = \dot{\vec{r}}_0 \end{cases}$$

Où \vec{f}_{ap} : Accélération perturbatrice causée par l'aplatissement de la terre.

$\vec{f}_{s,l}$: Accélération perturbatrice causée par le soleil et la lune.

\vec{f}_{ta} : Accélération perturbatrice causée par la traînée atmosphérique.

3.3.1. Perturbations géopotential dues à l'aplatissement de la terre

Dans la réalité, la terre est un peu asymétrique aux pôles et renflée à l'équateur et sa forme la plus appropriée est celle d'un ellipsoïde d'une distribution de masse non uniforme. Pour prendre en compte la distribution de masse non uniforme et le champ gravitationnel non uniforme résultant, le potentiel gravitationnel ou géopotential U de la terre est donné généralement par une série des harmoniques sphériques [23] [27] [28] :

$$U(r, \Phi, \lambda) = U_0 + U_p$$

$$U(r, \Phi, \lambda) = \frac{\mu}{r} \left(1 + \sum_{n=2}^{\infty} \sum_{m=0}^n \left(\frac{R_E}{r}\right)^n P_{nm}(\sin \Phi) [C_{nm} \cos m\lambda + S_{nm} \sin m\lambda] \right) \quad (2.14)$$

Où r , Φ et λ sont respectivement la distance géocentrique, la latitude géocentrique et la longitude géographique mesurée vers l'est du méridien Greenwich, $U_0 = \frac{\mu}{r}$ est le potentiel Newtonien, U_p est le potentiel perturbateur, μ est le paramètre gravitationnel de la terre, R_E est le rayon équatorial moyen, $P_{nm}(\sin \Phi)$ est la fonction de Legendre associée normalisée, et (C_{nm}, S_{nm}) sont les coefficients harmoniques sphériques normalisés. Ces coefficients peuvent être classés en trois groupes et c'est uniquement avec les *coefficients harmoniques zonaux* dépendants uniquement de la latitude ($m = 0$ et $C_{n0} = -J_n$) que le potentiel $U(r, \Phi, \lambda)$ prend en compte l'aplatissement de la terre [23] [27].

En éliminant donc les termes indépendants de la latitude, le potentiel perturbateur de la formule (2.14) devient :

$$U_p = \frac{\mu}{r} \left\{ - \sum_{n=2}^{\infty} J_n \left(\frac{R_E}{r} \right)^n P_n(\sin \Phi) \right\} \quad (2.15)$$

$$\text{avec : } P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n$$

Le terme J_2 constitue, de loin, la plus forte perturbation due à la forme de la Terre (1000 fois plus importante que le coefficient suivant J_3) [23]. En ignorant les termes d'ordre supérieur, le potentiel gravitationnel perturbateur dû au terme J_2 est donné en coordonnées cartésiennes par :

$$U_p = \frac{\mu J_2 R_E^2}{2r^3} \left(1 - 3 \left(\frac{z}{r} \right)^2 \right) \quad (2.16)$$

En notant que $\vec{r} = x.\vec{i} + y.\vec{j} + z.\vec{k}$, on peut exprimer l'accélération perturbatrice causée par l'aplatissement de la terre par [23] [29] :

$$\vec{f}_{ap} = \overrightarrow{grad}(U_p) = \frac{\partial U_p}{\partial x} \vec{i} + \frac{\partial U_p}{\partial y} \vec{j} + \frac{\partial U_p}{\partial z} \vec{k} = f_{ap}(x).\vec{i} + f_{ap}(y).\vec{j} + f_{ap}(z).\vec{k}$$

$$\begin{cases} f_{ap}(x) = -\frac{3\mu J_2 R_E^2}{2r^5} \cdot x \cdot \left(1 - \frac{5z^2}{r^2} \right) \\ f_{ap}(y) = -\frac{3\mu J_2 R_E^2}{2r^5} \cdot y \cdot \left(1 - \frac{5z^2}{r^2} \right) \\ f_{ap}(z) = -\frac{3\mu J_2 R_E^2}{2r^5} \cdot z \cdot \left(3 - \frac{5z^2}{r^2} \right) \end{cases} \quad (2.17)$$

3.3.2. Perturbations dues à la traînée atmosphérique

Lorsque l'altitude de l'orbite est inférieure à environ 1200 km, la haute atmosphère est présente et le satellite rencontre les particules atmosphériques dans la direction de son mouvement, ce qui entraîne une force de traînée dans le sens opposé à la vitesse. L'amplitude de la force de traînée dépend de la densité locale de l'atmosphère, de la vitesse, de l'orientation et de la forme de l'engin spatial [23] [28].

La traînée atmosphérique constitue la principale force non gravitationnelle agissant sur des satellites en orbite LEO. Elle agit dans la direction opposée du vecteur vitesse et réduit progressivement l'énergie cinétique de l'orbite. Cette réduction d'énergie provoque une diminution de l'altitude entraînant le retour du satellite dans l'atmosphère [10].

La trainée atmosphérique est exprimée par [10] [23] [30] [31] [32] :

$$\vec{f}_{ta} = -\frac{1}{2} C_d \frac{A}{m} \rho |\vec{v}_r| \vec{v}_r \quad (2.18)$$

$$A = \sum_{i=1}^n A_i \begin{cases} A_i = 0 & \text{si } \vec{s}_i \cdot \vec{v}_r < 0 \\ A_i = \vec{s}_i \cdot \vec{v}_r & \text{autre} \end{cases}$$

Où m est la masse du satellite, C_d est le coefficient de la trainée, ρ est la densité atmosphérique, \vec{v}_r est la vitesse relative du satellite par rapport à l'atmosphère et A est la surface totale projetée perpendiculairement sur \vec{v}_r avec \vec{A}_n le vecteur surface de la $n^{\text{ème}}$ face du satellite dans un référentiel de corps.

Le coefficient de la trainée C_d est un paramètre sans dimension qui reflète la susceptibilité du satellite à la trainée atmosphérique. Il dépend de la forme du satellite et de la manière dont les particules interagissent avec lui. La valeur fréquemment utilisée pour les Cubesats est obtenue pour un écoulement sur une face plane, à savoir $C_d = 2.2$ [33].

La vitesse relative du satellite par rapport à l'atmosphère, en supposant que ce dernier tourne avec la terre, est [23] [32] :

$$\vec{v}_r = \vec{v} - \vec{\omega}_E \wedge \vec{r} \quad (2.19)$$

Où \vec{v} est la vitesse inertielle du satellite, $\vec{\omega}_E$ est la vitesse angulaire de la terre et \vec{r} est la position inertielle du satellite.

La densité atmosphérique peut être exprimée en utilisant un modèle simple et statique qui suppose que la densité décroît exponentiellement avec l'altitude [23] :

$$\rho = \rho_i \cdot \exp\left[-\frac{h - h_i}{H_i}\right] \quad \text{avec } h_i < h < h_{(i+1)} \quad (2.20)$$

Où h est l'altitude du satellite, h_i est l'altitude initiale pour un intervalle d'altitude donné, H_i est la hauteur d'échelle applicable pour un intervalle d'altitude, et ρ_i est la densité de référence correspondante à l'altitude initiale h_i de l'intervalle d'altitude.

Ce modèle, léger du point de vue calculs numériques, conserve une précision raisonnable pour les missions type Cubesat. Les données d'intervalles d'altitudes associées à (2.20) et calibrées par rapport au modèle CIRA72 (COSPAR International Reference Atmosphere de 1972), sont données en *ANNEXE A*.

3.3.3. Perturbations du 3^{ème} corps

Les perturbations dues aux forces gravitationnelles exercées par un troisième corps, comme le soleil ou la lune, affectent principalement les satellites au-dessus des orbites LEO. Leurs effets deviennent perceptibles quand les effets de la traînée commencent à diminuer [23].

L'accélération perturbatrice exercée par un 3^{ème} corps est facilement descriptible, dans un système de coordonnées cartésien géocentrique, par l'accélération directe exercée sur le satellite par le 3^{ème} corps moins l'accélération indirecte exercée sur la terre par le même corps [23] [32] :

$$\vec{f}_{3C} = \sum_i Gm_i \left(\frac{(\vec{r}_i - \vec{r}_{sat})}{|\vec{r}_i - \vec{r}_{sat}|^3} - \frac{\vec{r}_i}{|\vec{r}_i|^3} \right) \quad (2.21)$$

Où G est la constante universelle de gravitation, m_i est la masse du 3^{ème} corps, \vec{r}_i est le vecteur position de la terre vers le 3^{ème} corps, et \vec{r}_{sat} est le vecteur position de la terre vers le satellite.

On déduit de la formule (2.19) la perturbation luni-solaire exercée sur le satellite :

$$\vec{f}_{s,l} = \mu_l \left(\frac{\vec{r}_{sl}}{|\vec{r}_{sl}|^3} - \frac{\vec{r}_{tl}}{|\vec{r}_{tl}|^3} \right) + \mu_s \left(\frac{\vec{r}_{ss}}{|\vec{r}_{ss}|^3} - \frac{\vec{r}_{ts}}{|\vec{r}_{ts}|^3} \right) \quad (2.22)$$

Où

- $\mu_l = G \cdot m_l$ et $\mu_s = G \cdot m_s$: Le paramètre gravitationnel de la lune et du soleil
- \vec{r}_{sl} : vecteur position du satellite vers la lune (*figure 1-16*)
- \vec{r}_{tl} : Le vecteur position de la terre vers la lune (*figure 1-16*)
- $\vec{r}_{ss}, \vec{r}_{ts}$: Le vecteur position du satellite et de la terre vers le soleil

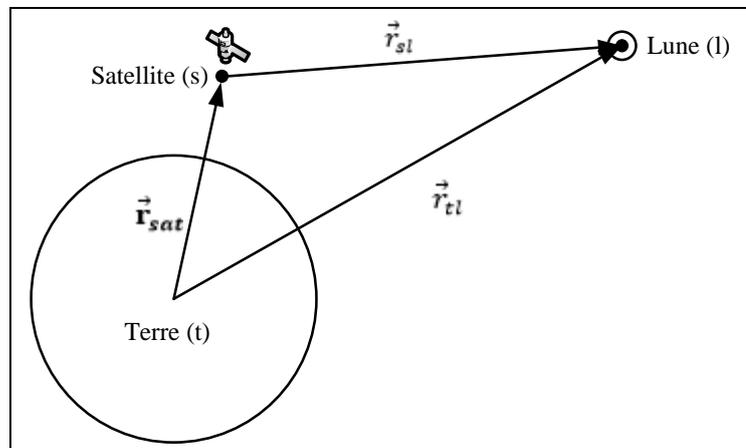


Figure 1-16 : Les vecteurs positions concernant la lune comme 3^{ème} corps

3.3.4. Evaluation des perturbations orbitales avec MATLAB

Pour résoudre, sous MATLAB, les équations différentielles du second ordre des accélérations perturbatrices, il faut les mettre tout d'abord sous la forme d'un système différentiel de premier ordre :

$$\left\{ \begin{array}{l} \dot{\vec{r}} = \vec{v} \\ \dot{\vec{v}} = -\frac{\mu}{r^3} \cdot \vec{r} + \vec{f}_p \end{array} \right\} \quad (2.23)$$

Où \vec{r} et \vec{v} sont les vecteurs d'état du satellite par rapport à la terre, et \vec{f}_p l'accélération des perturbations orbitales. Le système des équations différentielles du mouvement du satellite sous perturbations devient alors :

$$\left\{ \begin{array}{l} \dot{x} = v_x = f_1(x, y, z, \dot{x}, \dot{y}, \dot{z}) \\ \dot{y} = v_y = f_2(x, y, z, \dot{x}, \dot{y}, \dot{z}) \\ \dot{z} = v_z = f_3(x, y, z, \dot{x}, \dot{y}, \dot{z}) \\ \dot{v}_x = -\frac{\mu}{r^3} \cdot x + f_{px} = f_4(x, y, z, \dot{x}, \dot{y}, \dot{z}) \\ \dot{v}_y = -\frac{\mu}{r^3} \cdot y + f_{py} = f_5(x, y, z, \dot{x}, \dot{y}, \dot{z}) \\ \dot{v}_z = -\frac{\mu}{r^3} \cdot z + f_{pz} = f_6(x, y, z, \dot{x}, \dot{y}, \dot{z}) \end{array} \right. \quad (2.24)$$

Ensuite, on va utiliser l'algorithme d'intégration numérique Rung-Kutta (via le solveur ode23) avec les conditions initiales suivantes :

- Le vecteur d'état initial ($\vec{r}(t=0) = \vec{r}_0$; $\vec{v}(t=0) = \vec{v}_0$) obtenue à partir des éléments orbitaux du satellite (§ 3.1.5). L'orbite LEO initiale, choisi pour les simulations, a une altitude au périégée de 650 km, une excentricité de 0.031, une inclinaison de 70° et un RAAN de 30°.
- Les vecteurs position et vitesse initiaux de la lune et du soleil ont été ajustés pour avoir le barycentre à l'origine d'intégration, au moment initial qui est le jour JD = 2440400.5 (28 Juin 1969).

La simulation des différentes perturbations s'est faite le long d'une journée pour évaluer l'évolution de ces effets sur plusieurs révolutions puisque la durée d'une révolution est de 97.10 mn (le code source de certains scripts MATLAB est donné en *ANNEXE B*).

a. Effets de l'aplatissement de la terre (Propagateur J2)

La *figure 1-17* montre les perturbations dues à l'aplatissement de la terre, représentées par le principal coefficient harmonique zonal J2.

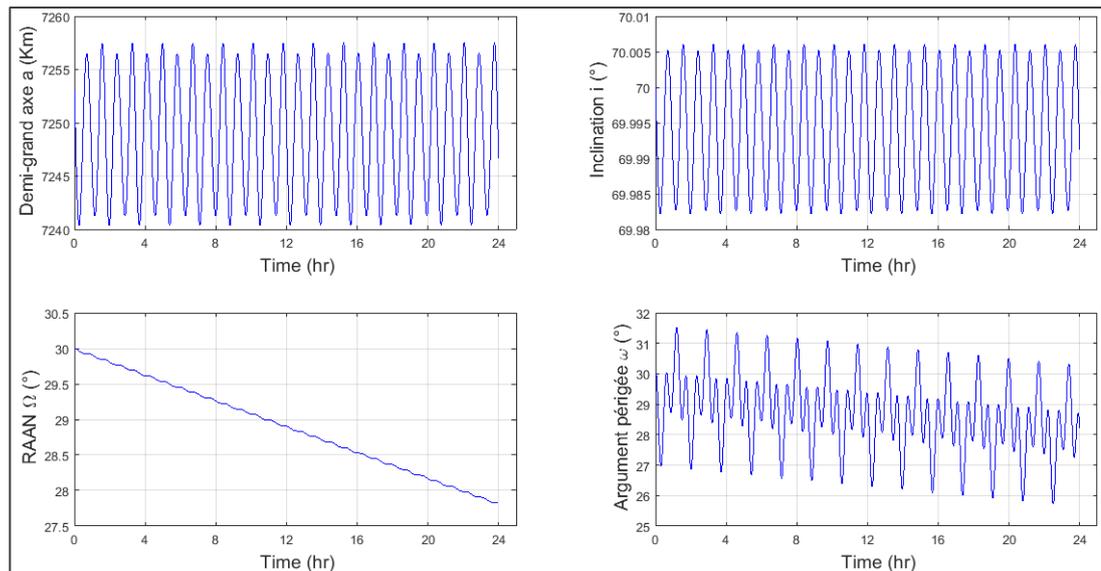


Figure 1-17 : Éléments orbitaux sous la perturbation J2 [MATLAB]

On remarque que les perturbations dominantes sont des variations séculaires dans le RAAN (régression de la ligne des nœuds) et dans l'argument du périégée (une précession du périégée).

Une simulation sous STK pendant une période de sept jours (01 août 2018 au 07 août 2018) a permis de visualiser la régression de la ligne des nœuds sur l'orbite inertielle 3D du nanosatellite MASAT1 comme le montre la *figure 1-18*.



Figure 1-18 : Orbite inertielle 3D du nanosatellite MASAT1

[STK : J2 Propagator – altitude 650km – inclinaison 70° - excentricité 0.013]

b. Effets de la traînée atmosphérique et durée de vie orbitale

Pour simuler les effets de la traînée atmosphérique sous MATLAB, certains paramètres physiques du satellite ont été utilisés :

Paramètre	Valeur
Masse (m)	1.33 kg
Surface exposée (A)	0.014 m ²
Coefficient de la traînée (C_d)	2.2
Densité atmosphérique	Modèle CIRA72

Tableau 1-4 : Paramètres pour la simulation des effets de la traînée atmosphérique

La surface exposée au flux est difficile à estimer puisqu'elle dépend de l'attitude du satellite. Pour notre simulation, nous avons considéré une surface de 0.014 m² qui correspond à un angle d'incidence de 45° avec les faces du Cubesat.

La *figure 1-19* montre les perturbations dues à la traînée atmosphérique :

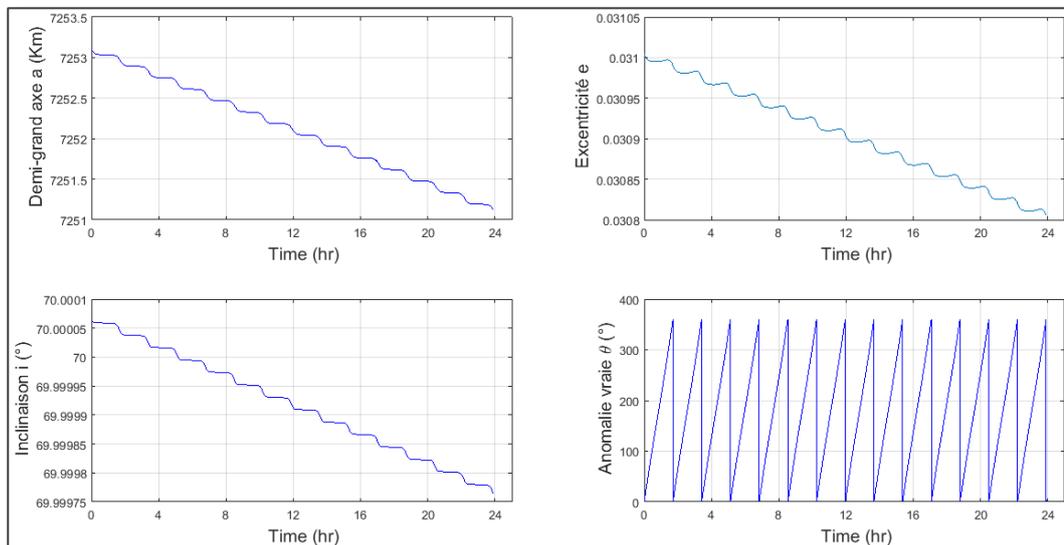


Figure 1-19 : Éléments orbitaux sous l'effet de la traînée [MATLAB]

On remarque que l'effet prédominant de la traînée atmosphérique est la diminution progressive du demi- grand axe et de l'excentricité. L'orbite devient ainsi circulaire à chaque révolution du satellite, avec une diminution plus prononcée de l'altitude de l'apogée par rapport à l'altitude du périégée comme le montre la *figure 1-20*.

Dans la plupart des cas, la traînée atmosphérique est la principale cause du retour du satellite vers l'atmosphère terrestre limitant ainsi la durée de vie orbitale.

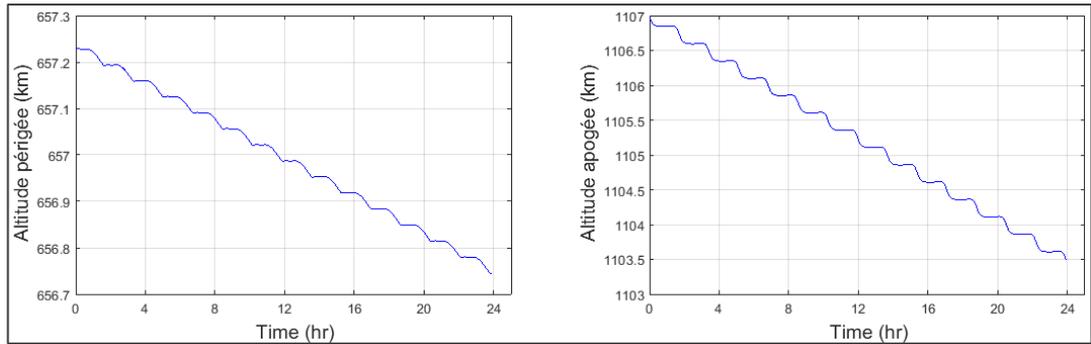


Figure 1-20 : Variation du périgée et de l'apogée sous l'effet de la trainée [MATLAB]

Pour prédire la durée de vie orbitale du nanosatellite MASAT1, nous avons utilisé l'outil "lifetime" du logiciel STK avec les mêmes paramètres que MATLAB, sauf pour la détermination de la densité atmosphérique où on a utilisé le modèle Jacchia 1970 qui est plus précis que le modèle CIRA72. On remarque que pour le scénario orbital correspondant à 650 km (voir *figure 1-21*), la durée de vie est de 5.6 ans contre 1.5 ans pour le scénario avec une altitude de 400 km (voir *figure 1-22*).

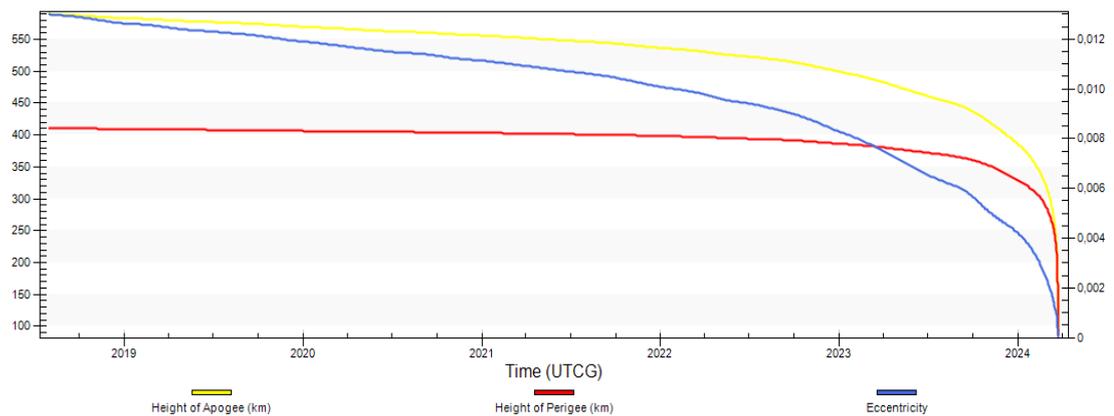


Figure 1-21 : Durée de vie orbitale de MASAT1 pour une altitude de 650 km [STK]

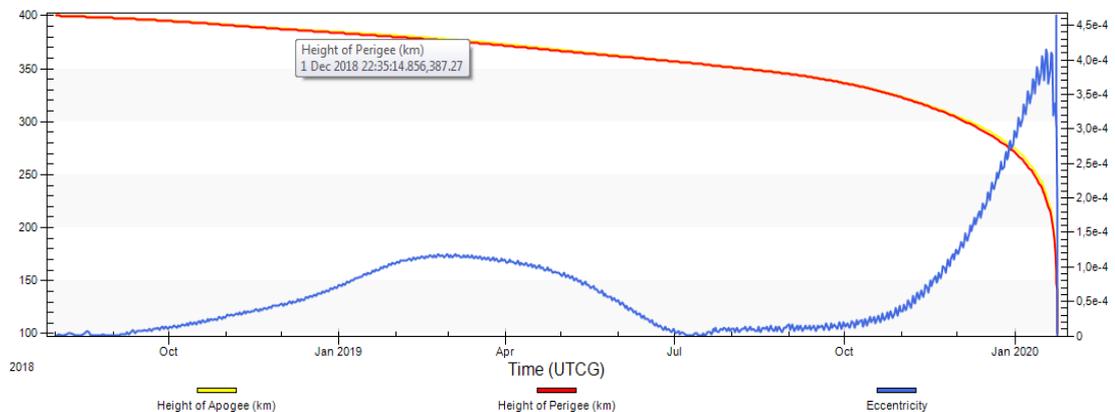


Figure 1-22 : Durée de vie orbitale de MASAT1 pour une altitude de 400 km [STK]

c. Effets des perturbations causées par le soleil et la terre (3^{ème} corps)

La *figure 1-23* montre l'effet des perturbations causées par le soleil et la lune sur certains paramètres orbitaux :

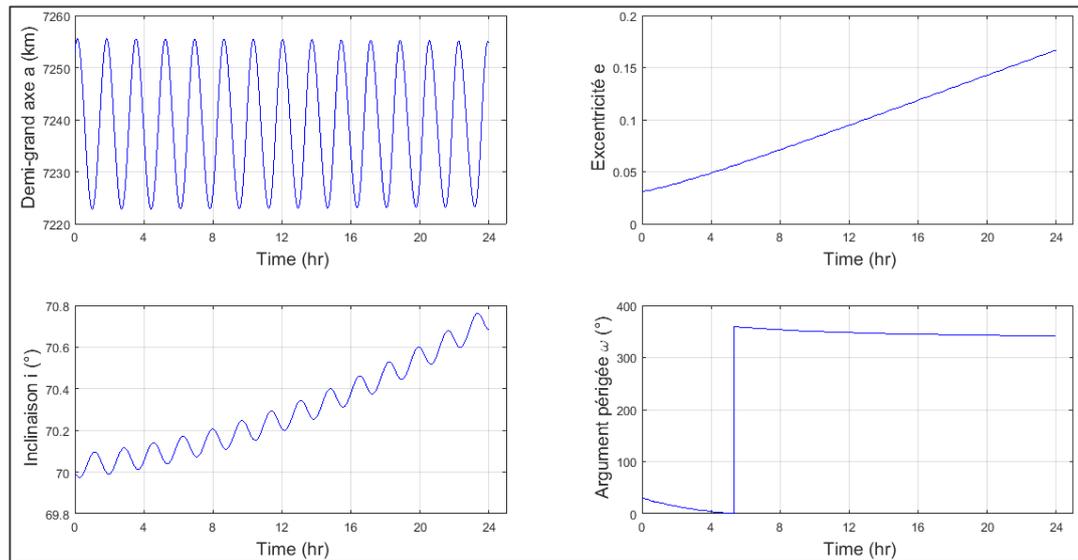


Figure 1-23 : Éléments orbitaux sous l'effet lunisolaire [MATLAB]

On remarque qu'il n'y a pas de variations séculaires au niveau du demi-grand axe, mais il y en a au niveau des autres paramètres orbitaux comme l'inclinaison, l'argument du périégée ou l'excentricité.

3.4. Couverture de la terre

L'étude de la couverture de la terre permet de déterminer les stations au sol susceptibles d'entrer en communication avec le segment spatial. Dans notre cas, cette couverture fait référence à la partie de la Terre que l'antenne omnidirectionnelle du segment spatial peut couvrir à un instant donné ou sur une période prolongée [8]. En effet, vu qu'on prévoit un contrôle d'attitude passif, ce choix de configuration d'antennes permet d'assurer une réception fiable du signal incident quel que soit la position des antennes par rapport à la station au sol.

Pour évaluer la couverture de notre antenne, on détermine deux paramètres clés [8] :

- La zone du champ de vision ou la zone d'accès qui représente la zone que l'antenne peut voir à tout instant (*figure 1-24*).
- Le taux de couverture de zone (Area Coverage Rate) qui représente la vitesse à laquelle l'antenne détecte ou accède à de nouvelles zones.

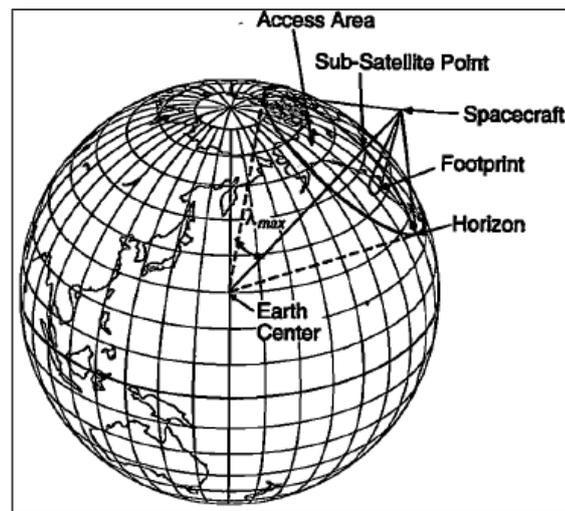
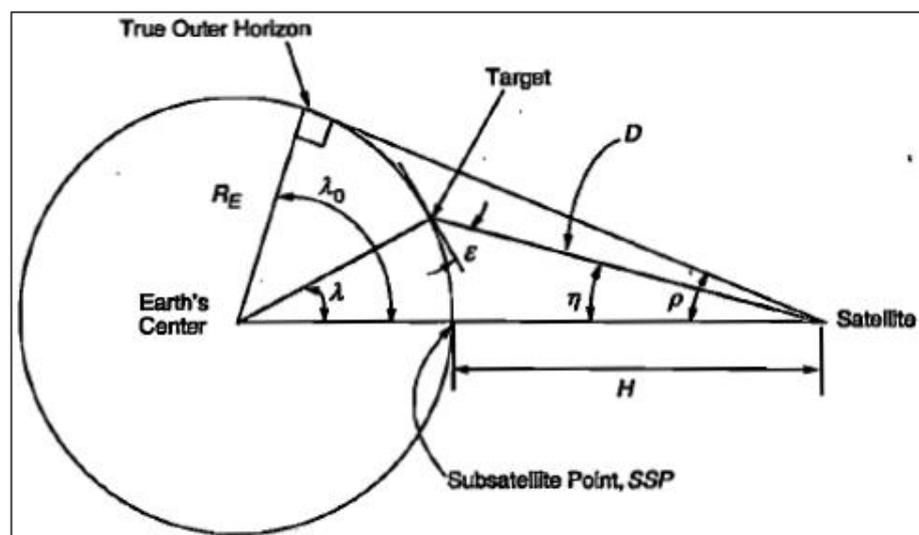


Figure 1-24 : Représentation de la zone du champ de vision (Access Area) [8]

L'évaluation de la couverture se fera en deux étapes : premièrement, une évaluation analytique avec des formules approximatives pour estimer la zone du champ de vision ; deuxièmement, des simulations numériques sous STK pour fournir les données utiles pour l'analyse de la mission.

3.4.1. Approximations analytiques

Toutes les formules utilisées seront déduites de la géométrie satellitaire décrite par la *figure 1-25*, en considérant un angle d'élevation au-dessus de l'horizon vrai $\epsilon = 5^\circ$, car le signal à la limite de l'horizon vrai est atténué à cause de l'atmosphère et des obstacles naturels.



η : angle de nadir

λ : angle au centre de la terre

H : altitude du satellite

ϵ : angle d'élevation au dessus de l'horizon

ρ : rayon angulaire de la terre

R_E : rayon de la terre

Figure 1-25 : Relations angulaires entre satellite, cible et centre de la Terre [8]

La première étape consiste à calculer l'angle de nadir η et l'angle au centre de la terre λ à partir des relations suivantes [8] :

$$\eta = \arcsin\left(\frac{R_E \cos \varepsilon}{R_E + H}\right) \text{ et } \rho = \arcsin\left(\frac{R_E}{R_E + H}\right) \quad (2.25)$$

$$\lambda + \varepsilon + \eta = 90^\circ \quad (2.26)$$

La longueur de la zone d'accès L_{FP} et la zone d'accès A_{FP} seront ensuite déduites à partir des relations suivantes [8] :

$$L_{FP} = 2 \lambda \left(\frac{2\pi R_E}{360}\right) \quad (2.27)$$

$$A_{FP} = 2\pi R_E^2 \int_0^\lambda \sin \varphi d\varphi = 2\pi R_E^2 (1 - \cos \lambda) \quad (2.28)$$

On remarque que la couverture de la terre dépend essentiellement de l'altitude du satellite. Le tableau suivant résume les résultats des calculs pour les différents scénarios orbitaux (§ 3.2) :

Altitude du scénario	ρ	η	L_{FP}	A_{FP}
400 km	70.21°	69.61°	3426 km	$9.14 \cdot 10^6 \text{ km}^2$
500 km	68.01°	67.47°	3902 km	$11.84 \cdot 10^6 \text{ km}^2$
650 km	65.15°	64.69°	4521 km	$15.85 \cdot 10^6 \text{ km}^2$

Tableau 1-5 : Paramètres pour la simulation des effets de la trainée atmosphérique

3.4.2. Estimation de la couverture de la terre de MASAT1

Les formules analytiques fournissent un moyen simple et rapide d'évaluer la couverture de la terre, mais cette approche présente plusieurs limites vu qu'elle ne prend pas en compte les orbites circulaires, et la rotation de la terre [14].

La simulation de la couverture de la terre sous STK permet de prendre en compte les paramètres orbitaux du satellite tout en donnant la possibilité de fixer l'angle d'élévation au-dessus de l'horizon. La *figure 1-26* montre le résultat de la simulation pour les différents scénarios orbitaux, réalisée avec les paramètres suivants :

- Antenne omnidirectionnelle
- Angle d'élévation au-dessus de l'horizon de 5°
- Propagateur J4 (inclut les effets du 1^{er} et second ordre de J2 et du 1^{er} ordre de J4)
- Début de simulation : 01 Août 2018 à 11:00:00.000 UTC
- Fin de simulation : 02 Août 2020 à 11:00:00.000 UTC

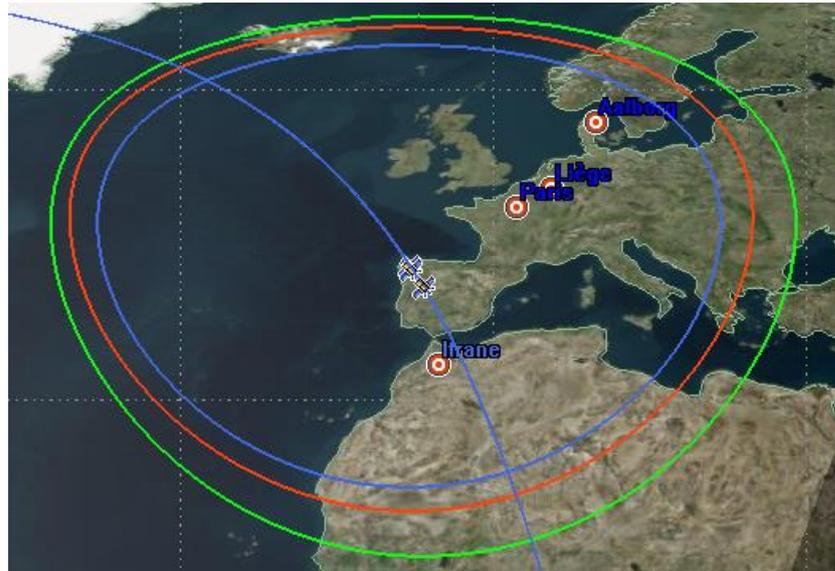


Figure 1-26 : Simulation de la couverture de la terre [STK]

L'analyse des résultats obtenus permet de remarquer que :

- L'augmentation de l'altitude permet d'augmenter la couverture du segment spatial.
- Une orbite par exemple de 650 km (en vert sur la *figure 1-26*) permet d'assurer la communication directe entre le segment spatial et les stations au sol situées par exemple en France, en Belgique ou en Danemark.

3.5. Temps de communication

Le temps de communication (TCOM) du segment spatial avec les stations au sol est lié directement à la couverture de la Terre, et sa détermination permet d'estimer :

- La quantité de données que le segment spatial peut envoyer par passage, en fonction du débit disponible dans le sous-système COM.
- La durée maximale des intervalles entre les passages et donc la taille mémoire pour le stockage des données de télémétrie dans le segment spatial.

Le TCOM d'un passage au-dessus d'une station au sol est défini par :

$$TCOM = AOStime - LOStime$$

Avec : AOStime : Acquisition of the satellite signal time
 LOStime : loss of the satellite signal time

3.5.1. Mouvement apparent d'un satellite pour un observateur sur Terre

Pour déterminer le TCOM, nous devons déterminer les caractéristiques du mouvement apparent du satellite visible depuis la station au sol. Dans les orbites LEO circulaires,

nous pouvons négliger la rotation de la Terre durant une période relativement brève où le satellite passe au-dessus de la station au sol [14]. Tout au long de cette étude, nous utilisons la notation adoptée à la *figure 1-25* du § 3.4.1.

La *figure 1-27* montre la géométrie utilisée avec le petit cercle centré autour de la station au sol qui représente les points de projection du satellite SSP (Subsatellite Point) pour un angle d'élévation au-dessus de l'horizon $\varepsilon_{min} = 5^\circ$ (vu que c'est une mission de communication [14]).

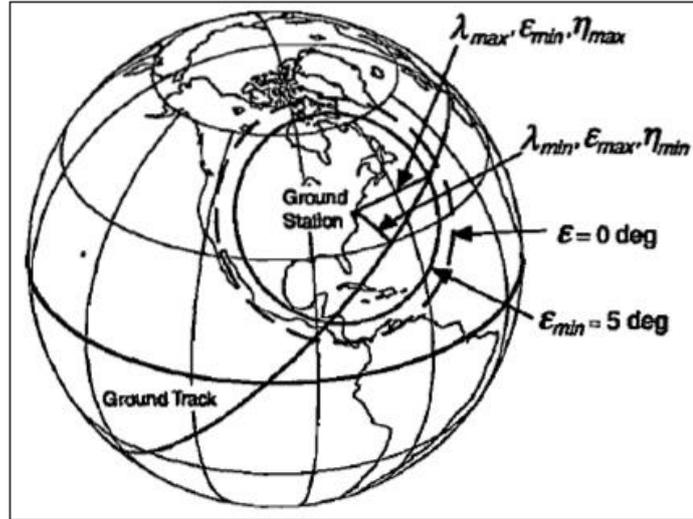


Figure 1-27 : Géométrie d'une période de visibilité [14]

La taille de ce cercle d'accessibilité et, par conséquent, le temps de communication dépend fortement de la valeur de ε . Une fois la valeur de ε_{min} fixée, nous pouvons définir l'angle au centre de la Terre maximal λ_{max} , l'angle nadir maximal η_{max} , et la portée maximale D_{max} à laquelle le satellite sera toujours visible. Ces paramètres sont déterminés par l'application des équations suivantes [14] :

$$\sin \eta_{max} = \sin \rho \cos \varepsilon_{min} \quad (2.29)$$

$$\lambda_{max} = 90^\circ - \varepsilon_{min} - \eta_{max} \quad (2.30)$$

$$D_{max} = R_E \frac{\sin \lambda_{max}}{\sin \eta_{max}} \quad (2.31)$$

Le cercle de rayon λ_{max} correspondant à ε_{min} est considéré comme l'horizon effectif, à ne pas confondre avec l'horizon vrai où $\varepsilon = 0$.

La durée de communication et l'élévation maximum ε_{max} du satellite dépend de la proximité de la station au sol par rapport à la trace au sol du satellite. Cette dernière dépend du plan orbital qui est défini par l'inclinaison i de l'équateur, le RAAN Ω et

la longitude du nœud ascendant L_{node} (§ 3.1). En fait l'orbite reste immobile dans l'espace inertiel quand la terre tourne au-dessous de lui (en excluant les perturbations de Ω qui sont définies par rapport aux étoiles) [14]. D'autre part, L_{node} est défini par rapport à la surface de la Terre, par conséquent, augmente de 360° chaque 1436 min qui est la période de rotation de la Terre par rapport aux étoiles.

À cause de cette rotation de l'orbite par rapport à la Terre, il est commode de parler du nœud ascendant instantané qui est L_{node} évalué au moment de passage au-dessus de la station au sol. À des fins géométriques, il est également approprié de travailler avec le pôle instantané de l'orbite.

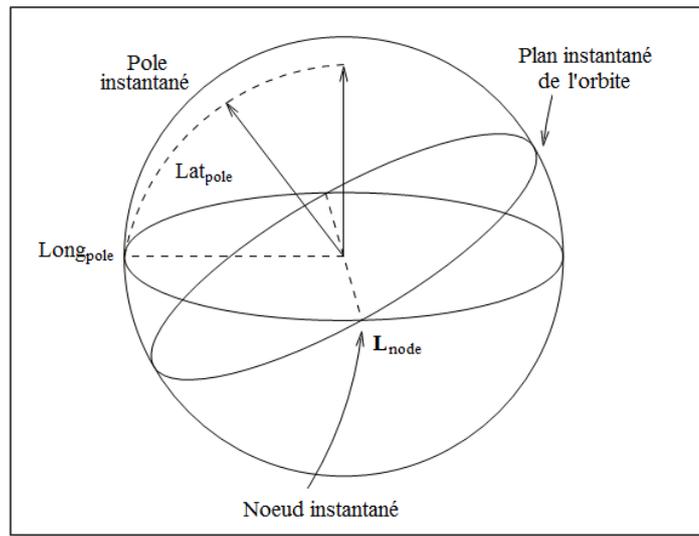


Figure 1-28 : Orbite instantanée d'un satellite

A partir de la **figure 1-28**, on peut facilement déterminer les coordonnées du pôle instantané de l'orbite par :

$$\mathbf{Lat}_{pole} = 90^\circ - i \quad \text{et} \quad \mathbf{Long}_{pole} = L_{node} - 90^\circ \quad (2.32)$$

A partir des coordonnées de la latitude et de la longitude du pôle instantané de l'orbite et de la station au sol, λ_{min} est donné par [14] :

$$\sin \lambda_{min} = \sin Lat_{pole} \sin Lat_{gs} + \cos Lat_{pole} \cos Lat_{gs} \cos \Delta Long \quad (2.33)$$

Où $\Delta Long$ est la différence de longitude entre la station au sol et le pôle instantané de l'orbite.

Une fois que nous savons λ_{min} (la distance angulaire entre la station au sol et la trace au sol), nous pouvons calculer l'angle nadir minimum η_{min} , l'angle d'élévation minimum ε_{min} , et la portée minimum D_{min} , à partir des formules 2.29, 2.30 et 2.31.

L'amplitude en azimut 2Φ et la durée T du contact du satellite avec la station au sol sont obtenues en appliquant à nouveau la trigonométrie du triangle sphérique illustré dans la *figure 1-29* :

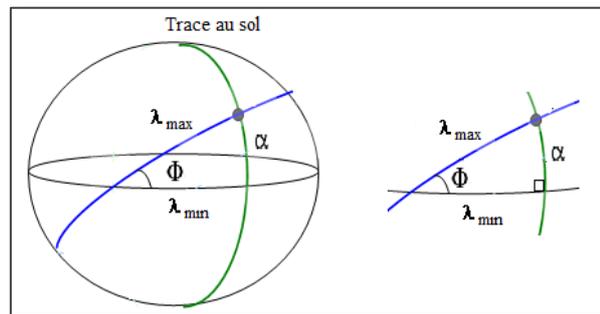


Figure 1-29 : Représentation géométrique de l'amplitude en azimut et la trace au sol

$$\cos \Phi = \frac{\tan \lambda_{\min}}{\tan \lambda_{\max}} \quad \text{et} \quad \cos \alpha = \frac{\cos \lambda_{\max}}{\cos \lambda_{\min}} \quad (2.34)$$

$$T = \left(\frac{P}{180^\circ}\right) \cos^{-1} \left(\frac{\cos \lambda_{\max}}{\cos \lambda_{\min}}\right) \quad (2.35)$$

Où P correspond à la période d'une orbite ($P = 1.658669 \cdot 10^{-4} \cdot (6378.14 + H)^{\frac{3}{2}}$).

La station au sol à un temps de visibilité maximale lorsque le satellite passe exactement au-dessus d'elle ($\lambda_{\min} = 0$). L'expression du TCOM maximal devient :

$$T = \frac{P}{180^\circ} \lambda_{\max} \quad (2.36)$$

3.5.2. Estimation du temps de communication pour MASAT1

En utilisant les formules démontrées au § 3.5.1, on peut estimer le TCOM maximal, en considérant un angle d'élévation au-dessus de l'horizon $\varepsilon_{\min} = 5^\circ$:

Altitude de l'orbite H	400 km	500 km	650 km
Rayon angulaire de la Terre ρ	70.21°	68.01°	65.15°
Période de l'orbite P (min)	92.56	94.61	97.72
Angle nadir η_{\max}	69.61°	67.47°	64.68°
Angle central de la terre λ_{\max}	15.39°	17.53°	20.32°
Période maximale D_{\max}	1806 km	2080 km	2450 km
TCOM maximal (min)	7.91	9.21	11.03

Tableau 1-6 : Temps de communication estimés par calcul

Mais vu que les paramètres orbitaux ne peuvent pas être imposés, et que le chemin apparent d'un satellite vers une station au sol ne présente pas une géométrie simple, on a eu recours au logiciel de simulation STK qui permet de générer des données plus précises sur de longues périodes.

Le **tableau 1-7** résume les résultats obtenus sur une durée de 6 mois à partir de 1 Août 2018, pour la station au sol principale d'Ifrane :

Altitude	Inclinaison	Nombre de révolution par jour	Durée d'une révolution (min)	TCOM max (min)	TCOM moyen par passage (min)	Passages par jour	TCOM moyen par jour (min)	Durée maximale de non contact
400 km	40°	15.45	92.54	8.7	7.7	6	46.2	16 h 26 min
	65°				6.6	4	26.4	
	90°				6.5	3	19.5	
500 km	40°	15.12	94.61	10	8.9	6	53.4	15 h 11 min
	65°				7.7	4	30.8	
	90°				7.5	4	30.0	
650 km	40°	14.64	97.72	12	10.6	6	63.6	15 h 39 min
	65°				9.1	5	45.5	
	90°				8.9	4	35.6	
Moyenne		14.74	94.95	10.2	8.2	4 à 5	39.0	15 h 45 min

Tableau 1-7 : Temps de communication pour la station au sol d'Ifrane [STK]

Dans le pire des cas (400 km – 90°), nous avons un temps d'accès de 19.5 min / jour avec un temps de communication continue maximal d'environ 6.5 min. Dans le meilleur des cas (650 km – 40°), nous avons un temps d'accès de 63 min / jour avec un temps de communication continue d'environ 10 minutes.

Concernant la répartition temporelle (Timeline), nous avons réalisé une simulation sur une période fixe typique de 2 jours (du 11 Août 2018 au 13 Août 2018) en prenant en compte le meilleur et le pire cas possible. Ce "timeline" permettra d'établir un ordonnancement des différentes tâches à réaliser telles que la supervision, les mesures scientifique, l'échange de données, etc.

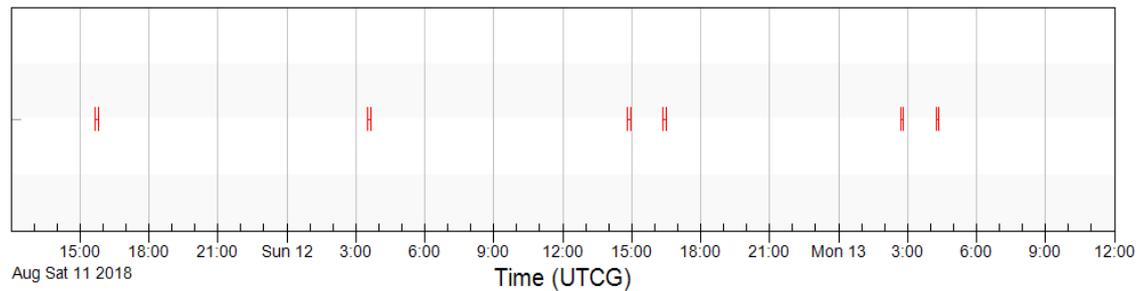


Figure 1-30 : Timeline pour une altitude de 400 km et une inclinaison de 90° [STK]

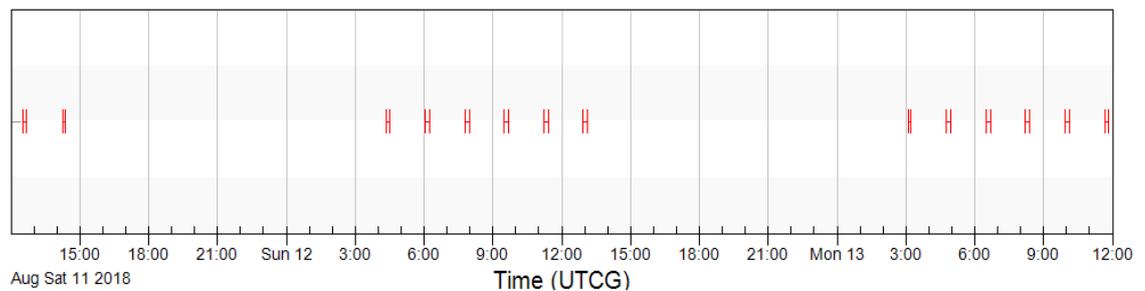


Figure 1-31 : Timeline pour une altitude de 650 km et une inclinaison de 40° [STK]

3.6. Durée d'éclipse

Un nanosatellite en orbite terrestre LEO traverse généralement plusieurs fois le cône d'ombre projeté par la Terre dans la direction opposée au soleil et reste dans l'obscurité pendant une certaine durée. La fréquence et la durée de cette éclipse dépendent de l'inclinaison et de l'altitude orbitale.

La durée de l'éclipse est un paramètre important pour la conception du système spatial puisqu'il permet de déterminer :

- Le flux thermique reçu par le segment spatial et par conséquent la distribution de la température au sein de ce dernier.
- La puissance électrique reçue par le satellite.
- Le nombre de cycles charge/décharge que doivent supporter les batteries du sous-système EdB.
- La Profondeur de Déchargement des Batteries (DOD : Depth of discharge).

3.6.1. Approximations analytiques

Pour évaluer la durée d'éclipse d'un satellite nous devons représenter le système (satellite, Terre, Soleil) dans une sphère céleste centrée sur le satellite et dans laquelle [14] :

- Le plan orbital constitue l'équateur de la sphère.
- L'axe X du référentiel fait constamment face au point nadir, c.-à-d. pointe vers le centre de la Terre.
- L'axe Z pointe vers le pôle de la sphère.

Dans ce référentiel, la Terre est représentée, du point de vue du satellite, par un cercle ombragé fixe de rayon angulaire ρ comme le montre la *figure 1-32* :

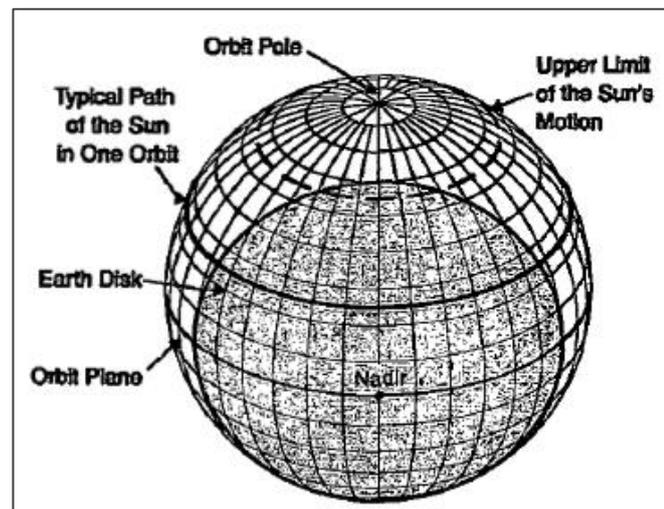
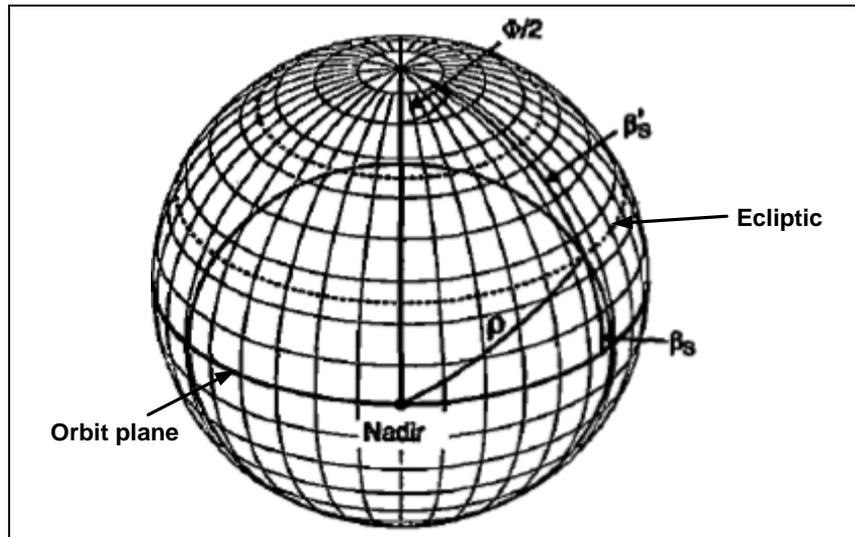


Figure 1-32 : Position de la Terre dans la sphère céleste centrée autour du satellite [14]

Puisqu'un axe du référentiel est toujours confronté à la Terre, ce système de coordonnées tourne une fois par orbite dans l'espace inertiel autour du pôle orbital. Ainsi, le Soleil considéré comme approximativement fixe dans l'espace inertiel, apparaîtra en train de tourner une fois par orbite autour du pôle orbital [14].

En conclusion, la trajectoire typique du soleil sur une orbite va décrire une circonférence appelé écliptique, qui est inférieure et parallèle au plan orbital. Le satellite subira donc une éclipse lorsque le soleil passe derrière la Terre, et la durée de l'éclipse est donnée par la fraction de circonférence qui coupe le disque de la Terre.

Ceci implique que la durée de l'éclipse dépend de la distance angulaire β_S entre le Soleil et l'équateur du système de référence centré autour du satellite comme le montre la géométrie de la *figure 2-33* :



ρ : rayon angulaire de la terre
 β_s : angle du soleil au dessus du plan orbital
 Φ : angle de rotation correspondant à la durée de l'éclipse

Figure 1-33 : Géométrie de l'éclipse sur la sphère céleste centrée autour du satellite [14]

L'éclipse maximale se produit si le Soleil est dans le plan de l'orbite, c.-à-d. $\beta_s = 0$, et diminue avec l'augmentation de β_s jusqu'à ne plus se produire pour $\beta_s \geq \rho$.

A partir des règles de trigonométrie du triangle sphérique de la **figure 2-33**, on peut facilement déterminer l'angle $\Phi/2$ et déduire la durée d'éclipse T_E :

$$\cos(\Phi/2) = \frac{\cos\rho}{\cos\beta_s} \quad (2.37)$$

$$T_E = P \left(\frac{\Phi}{360} \right) \quad (2.38)$$

Où P correspond à la période d'une orbite ($P = 1.658669 \cdot 10^{-4} \cdot (6378.14 + H)^{\frac{3}{2}}$).

3.6.2. Estimation de la durée d'éclipse pour MASAT1

Une analyse a été effectuée pour calculer la durée maximale et moyenne de l'éclipse en fonction de l'angle d'inclinaison et de l'altitude (en utilisant les formules et les données du logiciel de simulation STK durant 6 mois à partir de 1 Août 2018) :

Altitude H	400 km	500 km	650 km
Rayon angulaire de la terre ρ	70.21°	68.01°	65,15°
Période de l'orbite (min)	92.56	94.61	97.72
Angle de rotation d'éclipse Φ ($\beta_s = 0$)	140.42°	136.02°	130.30°
Durée d'éclipse maximale calculée (min)	36.10	35,75	35,37
Durée d'éclipse maximale STK (min)	36.14	35,80	35,44
Nombre maximal d'éclipses	2383	2331	2257

Tableau 1-8 : Durée d'éclipse maximale calculée et simulée durant 6 mois de mission

En résumé, les données les plus pertinentes à prendre en compte durant la conception préliminaire des sous-systèmes THER et EdB du segment spatial sont :

- Durée d'éclipse maximale : 36.14 min.
- Le nombre maximal d'éclipses pendant 6 mois (charge/décharge batterie) : 2383.

4. ETUDE ET EVALUATION DES CONTRAINTES SPATIALES

4.1. Environnement et sources radiatives

4.1.1. Ceintures de radiations de Van Allen

Les ceintures de radiations de Van Allen sont constituées d'électrons et d'ions énergétiques émises par le soleil, et piégés magnétiquement autour de la terre et distribués d'une façon non uniforme dans la magnétosphère.

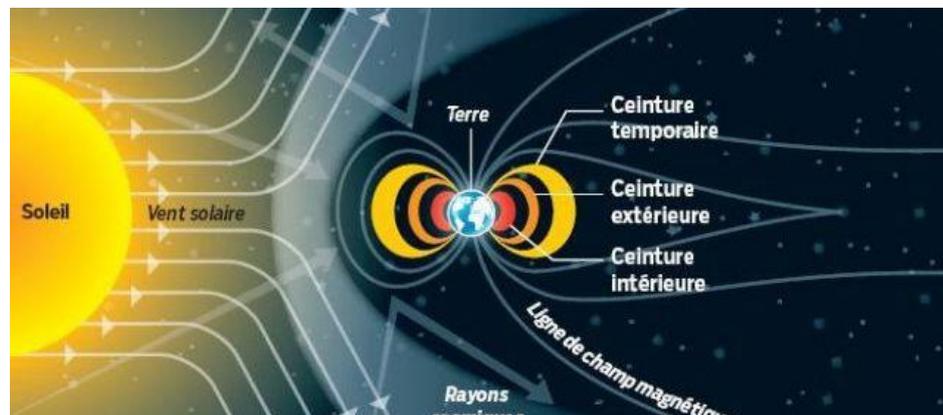


Figure 1-34 : Les ceintures de radiations de Van Allen [Internet]

Les particules piégées forment alors deux ceintures qui tirent leur nom du physicien américain James Alfred Van Allen qui les a découvertes en 1958, grâce au premier satellite américain Explorer 1 :

- Une ceinture intérieure qui commence à une altitude comprise entre 700 km et 1000 km environ pour atteindre sa limite supérieure à environ 10000 km. Elle se compose principalement de protons à haute énergie (jusqu'à plusieurs centaines de MeV, sachant que l'électron-volt (eV) est l'énergie cinétique d'un électron accéléré depuis le repos par une différence de potentiel d'un volt).
- Une Ceinture extérieure située entre 10000 km et 40000 km riche en électrons.

Puisque notre mission est conçue pour une orbite LEO, le segment spatial sera en contact uniquement avec la ceinture intérieure.

4.1.2. Les radiations solaires

Le Soleil domine l'environnement radiatif de l'orbite terrestre en raison de sa proximité et de son énergie extrêmement élevée. En plus d'être la principale source de protons, d'ions lourds et de particules piégées dans la magnétosphère de la Terre, elle permet aussi de limiter l'effet des rayons cosmiques galactiques (GCR) [36].

On distingue deux types de phénomènes en relation avec les radiations solaires [36] :

- L'éjection de masse coronale (CME : Coronal Mass Ejection) sous forme de bulles de plasma pouvant atteindre plusieurs dizaines de rayons solaires. Ces phénomènes peuvent durer plusieurs jours et ils émettent principalement des protons à haute énergie (*Figure 1-35*).
- L'éruption solaire qui est un jet de matière ionisée s'élevant à des centaines de milliers de kilomètres d'altitude (*Figure 1-36*). Il s'agit ici d'ions lourds hautement énergétiques.

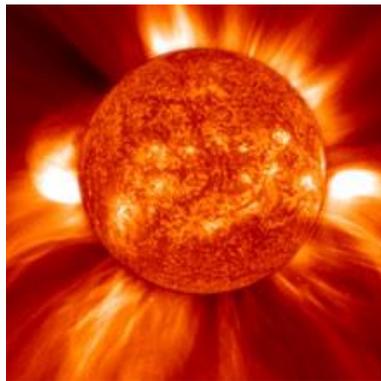


Figure 1-35 : Ejection de masse coronale [142]

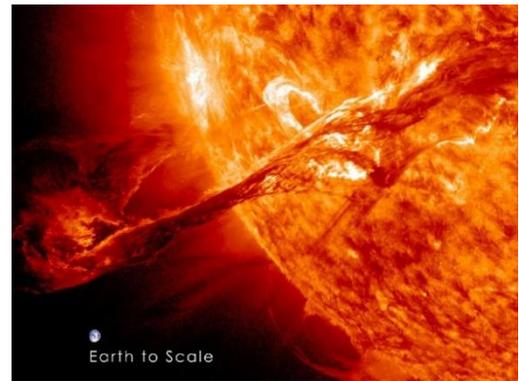


Figure 1-36 : Eruption solaire [143]

4.1.3. Rayons cosmiques galactiques (GCR)

Les rayons cosmiques sont des particules chargées de haute énergie dont l'origine peut être galactique ou extragalactiques. Ces particules sont très énergétiques ($\gg MeV$) parce qu'elles subissent de nombreuses accélérations durant leur long périple dans le cosmos. Le flux de ces particules cosmiques varie le long du cycle solaire :

- Lorsque le soleil est calme pendant le minima solaire, son champ magnétique se rétrécit et par conséquent le segment spatial va recevoir plus de particules cosmiques.
- Tandis que dans le maxima solaire le champ magnétique s'agrandit et le satellite va recevoir moins des particules cosmiques.

Il est donc souhaitable de réaliser notre mission durant une période où le soleil est actif afin de réduire le flux de ces particules très énergétiques et dangereuses.

Le *figure 1-37* est un récapitulatif des sources de particules rencontrées dans l'espace et dans l'atmosphère terrestre, ainsi que leur nature et leur énergie [37].

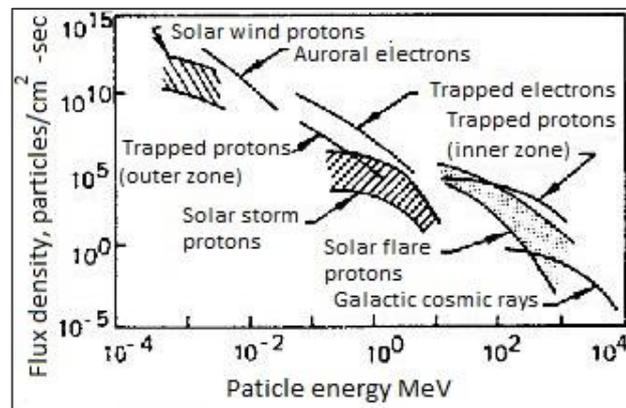


Figure 1-37 : Densité et énergie des particules dans l'environnement spatial [37]

4.2. Les effets des radiations sur les circuits électroniques

Les particules énergétiques des radiations produisent des effets sur les composants à base de semi-conducteurs du segment spatial et par conséquent limitent leur capacité à remplir les objectifs désirés. Ces effets radiatifs peuvent être divisés en deux catégories [64] :

- Les effets de dose cumulée (TID : Total Ionizing Dose) qui produisent des changements progressifs dans les paramètres de fonctionnement des dispositifs.
- Les effets d'évènement unique (SEE : Single Event Effects) qui provoquent des changements brusques ou des comportements transitoires des dispositifs.

4.2.1. Dose cumulée

La dose cumulée TID est définie par la quantité d'énergie déposée par ionisation ou excitation dans un matériau par unité de masse. Elle est presque exclusivement imputable aux particules piégées dans les ceintures de radiations et aux protons issus de l'activité solaire [38]. Leur effet cumulatif sur les circuits se traduit par une dérive des paramètres électriques (décalage des niveaux de seuils, l'augmentation des courants de fuite), qui conduit à une perte progressive puis totale de la fonctionnalité du composant [39]. Cependant l'effet TID peut être atténué en protégeant les composants par un blindage en aluminium par exemple [112].

On a utilisé le logiciel SPENVIS [144] pour évaluer la dose ionisante totale pour les différents scénarios en utilisant les paramètres d'entrée suivants (voir *figure 1-38*) :

- Modèle d'électrons : AE-8 MAX au maxima solaire
- Modèle de protons : AP-8 MAX au maxima solaire
- Durée : 2 ans à partir de 01 Août 2018
- Blindage en aluminium : 1 mm et 2 mm (modèle sphérique)

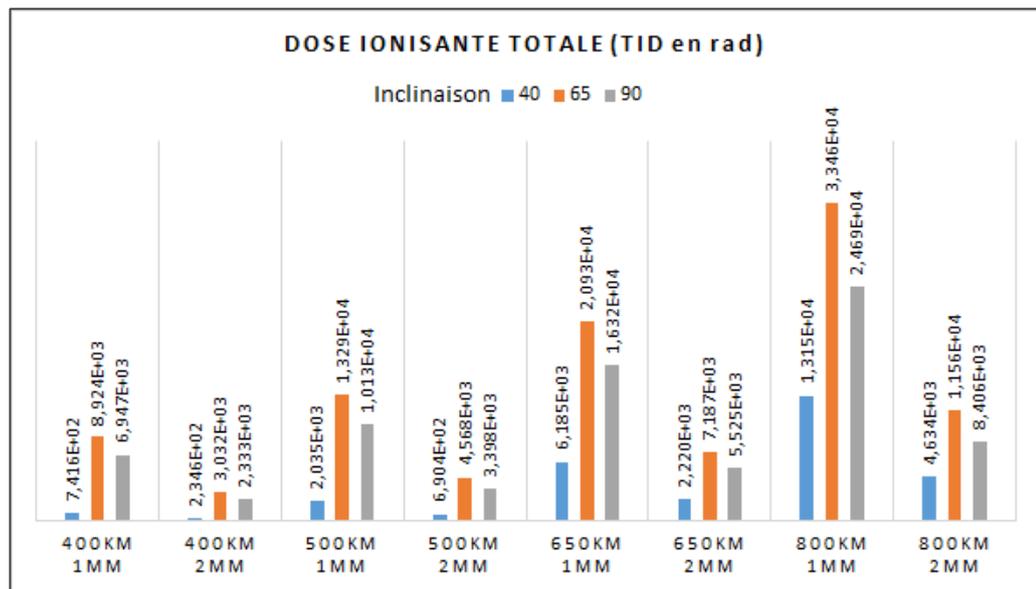


Figure 1-38 : La dose ionisante totale pour les différents scénarios [SPENVIS]

L'analyse des résultats montre que :

- Les particules piégées dans les ceintures de Van Allen contribuent presque à la totalité de la dose de radiation vu la grande différence entre la dose de radiation des orbites 400 km et 500 km par rapport à 800 km.
- Une augmentation de l'épaisseur de blindage de 1 mm à 2 mm permet de diminuer la dose de radiation par un facteur de 2.94.

4.2.2. Dose de Dommage de Déplacement (DDD)

La DDD est due au cumul de l'énergie non ionisante perdue (NIEL : Non Ionizing Energy Loss) qui définit la partie de l'énergie perdue par une particule en mouvement dans le matériau par unité de longueur. La collision entre une particule entrante et un atome du matériau déplace ce dernier de sa position initiale dans le réseau cristallin du semi-conducteur, en modifiant ou en endommageant ces propriétés [40].

On a utilisé le logiciel de simulation SPENVIS pour évaluer la dose de dommage de déplacement pour les différents scénarios en utilisant les paramètres d'entrée suivants (voir *figure 1-39*) :

- Modèle d'électrons : AE-8 MAX au maxima solaire
- Modèle de protons : AP-8 MAX au maxima solaire
- Durée : 2 ans à partir de 01 Août 2018
- Blindage en aluminium : 1 mm et 2 mm (modèle sphérique)

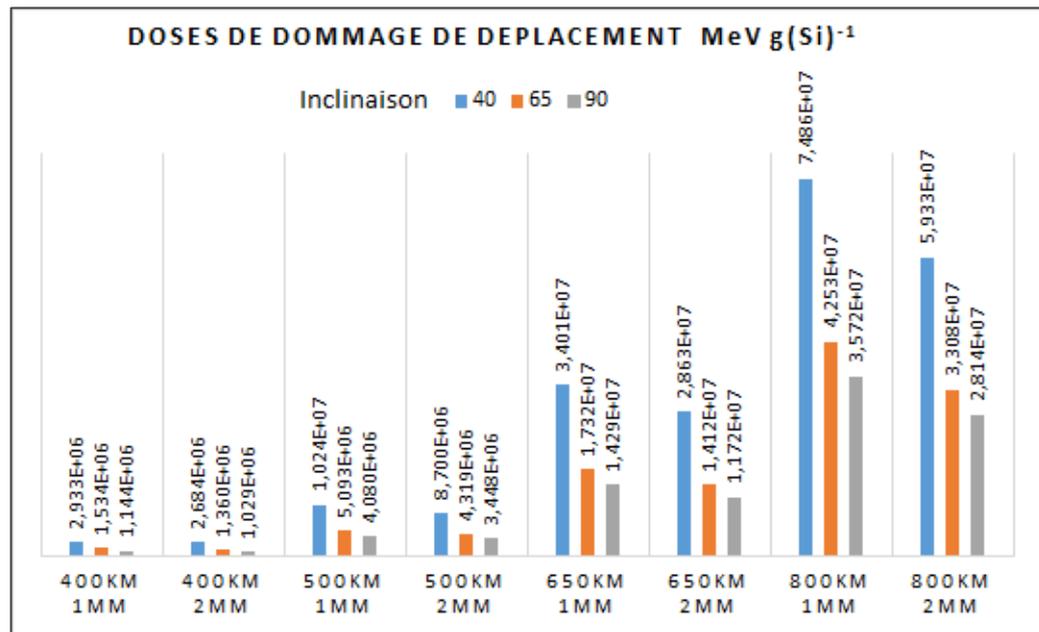


Figure 1-39 : Les doses de dommage de déplacement [SPENVIS]

L'analyse des résultats montre que :

- Les particules piégées dans les ceintures de Van Allen contribuent presque à la totalité de la dose de dommage de déplacement vue la grande différence entre la dose de dommage pour les orbites 400 km et 500 km par rapport à 800 km.
- Une augmentation de l'épaisseur de blindage de 1 mm à 2 mm permet de diminuer la DDD par un facteur de 1.19. Ce qui signifie que le blindage est moins efficace contre la DDD que contre la TID.

4.2.3. Effets des évènements singuliers (SEE)

Un effet singulier SEE se produit sous l'effet d'un évènement de radiation ionisant causé par une particule chargée qui traverse la matière. Cet effet est quantifié par un paramètre nommé **LET (Linear Energy Transfert)** et qui est défini comme l'énergie transférée par une particule ionisante traversant la matière, par unité de distance et pour une densité de matériaux spécifique (MeV/cm ou MeV·cm²/mg) [40].

En fonction de l'énergie transférée, différents effets singuliers SEE peuvent se produire :

- **SEU (Single Effect Upset)** : effet agissant sur les mémoires SRAM et entraînant la corruption des données. En effet, l'énergie transférée crée un changement d'état dans une structure de type bascule (1 passe à 0, et vice versa). Il s'agit d'un évènement dit « soft » car le circuit lui-même n'est pas endommagé.

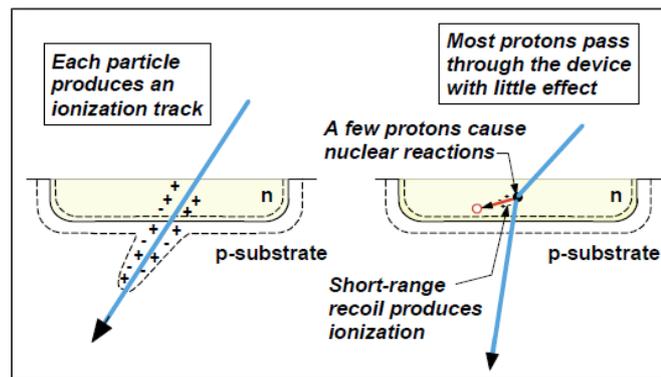


Figure 1-40 : Mécanismes des ions lourds et des protons causant les effets SEU [40]

- **SET (Single Effect Transient)** : c'est un changement inattendu de courte durée à la sortie d'un circuit combinatoire, provoqué par une particule chargée traversant le circuit [36]. Il s'agit aussi d'un évènement « soft » mais qui peut influencer la vitesse de traitement s'il modifie un signal d'horloge. Comme les particules chargées peuvent frapper le système électronique à tout moment et n'importe où, la seule protection possible est le blindage du système.
- **SEL (Single Effect latchUp)** : c'est un effet dû à une particule qui transfère une énergie suffisante pour mettre en conduction une structure parasite PNP (thyristor parasite), ce qui entraîne une forte augmentation du courant consommé par le composant pouvant mener jusqu'à sa destruction par effet joule. C'est l'évènement SEE le plus dangereux et ces effets peuvent être limités en surveillant l'alimentation des circuits et en prévoyant une mise hors tension rapide en cas d'évènement SEL [40].

On a utilisé le logiciel SPENVIS pour évaluer l'occurrence des évènements uniques SEE pour les différents scénarios en utilisant les paramètres d'entrée suivants :

- Modèle de protons : AP-8 MIN au minima solaire
- Modèle GCR (Galactic Cosmic Ray) : CREME96 au minima solaire
- Blindage en aluminium : 1 mm et 2 mm
- Composant cible : RAM MICRON 16M

L'analyse des résultats résumés par la **figure 1-41** montre que l'épaisseur du blindage peut diminuer l'occurrence des événements SEE et simplifier les mesures à prendre pour atténuer leurs effets.

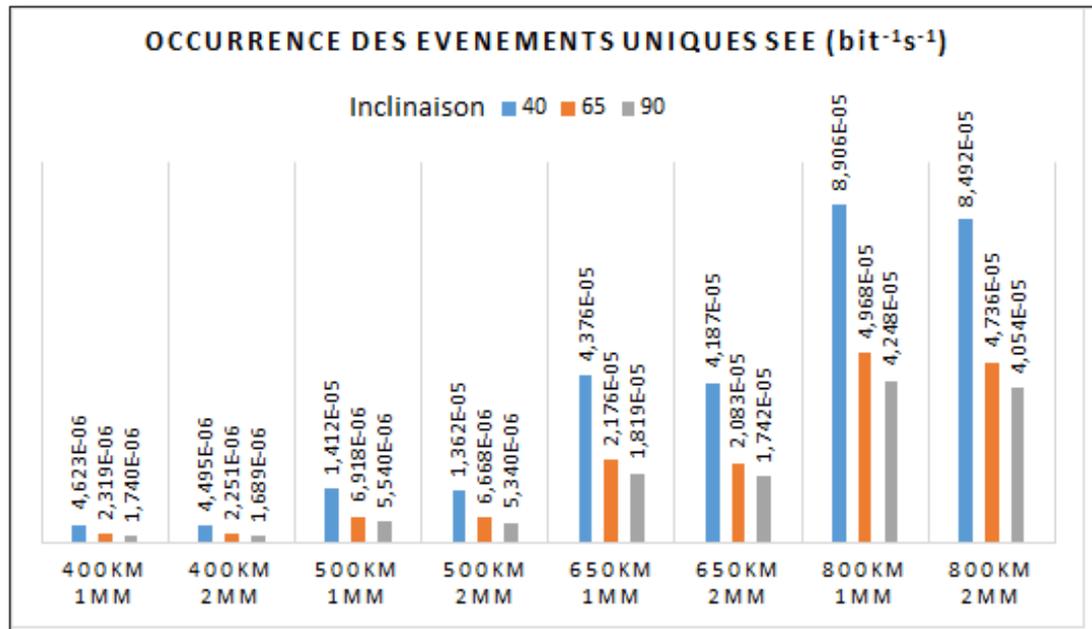


Figure 1-41 : L'occurrence des événements uniques [SPENVIS]

Comme conclusion, le **tableau 1-9** récapitule et résume tous les effets radiatifs pour une durée d'exposition de 2 ans derrière un blindage sphérique de 2 mm.

4.3. Dégazage

Le dégazage constitue une contrainte spatiale importante qu'on doit prendre en compte lors du choix des matériaux. En effet, à très faible pression, certains matériaux dont la pression de vapeur est plus élevée que la pression ambiante, vont s'évaporer ou se sublimer. Ces matériaux risquent alors de se condenser et de se déposer sur les parties les plus froides du segment spatial.

Ainsi, tous les matériaux utilisés dans le segment spatial doivent être conforme à la norme NASA-STD-6016 pour éviter d'avoir un dégazage à très faible pression [48] :

- TML (Total Mass Loss) < 0.1%.
- CVCM (Collected Volatile Condensable Mass) < 1%.

Scénario orbital	TID (rad)	DDD (MeV/g(Si))	SEE (/bit.s)
400 km / 40°	2,346E+02	2,684E+06	4,495E-06
400 km / 65°	3,032E+03	1,360E+06	2,251E-06
400 km / 90°	2,333E+03	1,029E+06	1,689E-06
500 km / 40°	6,904E+02	8,700E+06	1,362E-05
500 km / 65°	4,568E+03	4,319E+06	6,668E-06
500 km / 90°	3,398E+03	3,448E+06	5,340E-06
650 km / 40°	2,220E+03	2,863E+07	4,187E-05
650 km / 65°	7,187E+03	1,412E+07	2,083E-05
650 km / 90°	5,525E+03	1,172E+07	1,742E-05
800 km / 40°	4,634E+03	5,933E+07	8,492E-05
800 km / 65°	1,156E+04	3,308E+07	4,736E-05
800 km / 90°	8,406E+03	2,814E+07	4,054E-05

Tableau 1-9 : Récapitulatif des effets radiatifs avec un blindage de 2mm [SPENVIS]

5. EVALUATION DES CONTRAINTES DE LANCEMENT

Ce type de contraintes va conditionner les dimensionnements, les simulations, les choix des matériaux et les tests concernant le segment spatial de la mission. Puisque ni le mode de lancement ni le lanceur ne sont encore déterminés, on donne dans cette section un comparatif des contraintes concernant six lanceurs différents (ARIANE 5, SOYUZ, VEGA, H-IIA, DNEPR, et ROCKOT) pour prendre en compte les valeurs les plus contraignantes.

5.1. Charges quasi-statiques (CQS)

Elles représentent les accélérations auxquelles sera soumis le segment spatial durant les différentes phases de lancement. Elles peuvent être d'origine aérodynamique (vent, rafales, bourdonnement pendant la phase transsonique), ou causées par le système de propulsion (accélération longitudinale, poussée).

Le *tableau 1-10* suivant reprend les valeurs limites pour les six lanceurs à l'étude :

Lanceur	Accélération	
	Longitudinale (g) (Statique + dynamique)	Latérale (g) (Statique + dynamique)
ARIANE 5	+4.55	+0.25
ROCKOT	-1.5/+8.1	±0.9
H-IIA	-4/+1	±1.8
SOYUZ	-5 / +1.8	±1.8
VEGA	-6.8 / +3	±0.9
DNEPR	7.8 ±0.5	±0.8

Tableau 1-10 : Facteurs de charges quasi-statiques des différents lanceurs [41-47]

5.2. Vibrations sinusoïdales

Des excitations sinusoïdales affectent le véhicule de lancement durant son vol motorisé et pendant quelques phases transitoires.

La *figure 1-42* et la *figure 1-43* donnent respectivement les vibrations sinusoïdales longitudinales (suivant les axes x et y), ainsi que les vibrations sinusoïdales latérale (suivant l'axe z) pour les six lanceurs à l'étude :

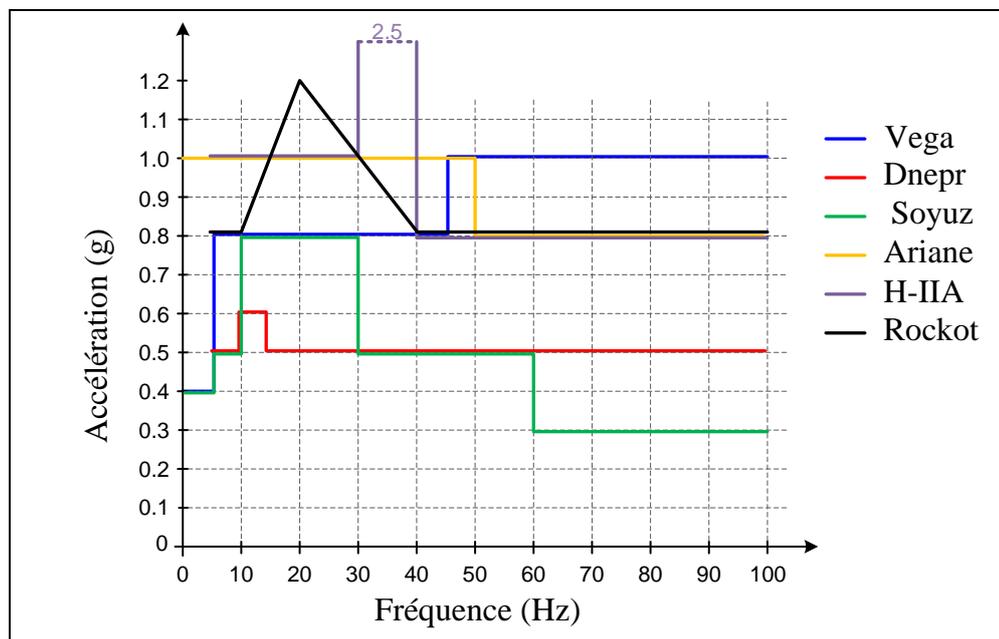


Figure 1-42 : Vibrations longitudinales sinusoïdales des différents lanceurs [41-47]

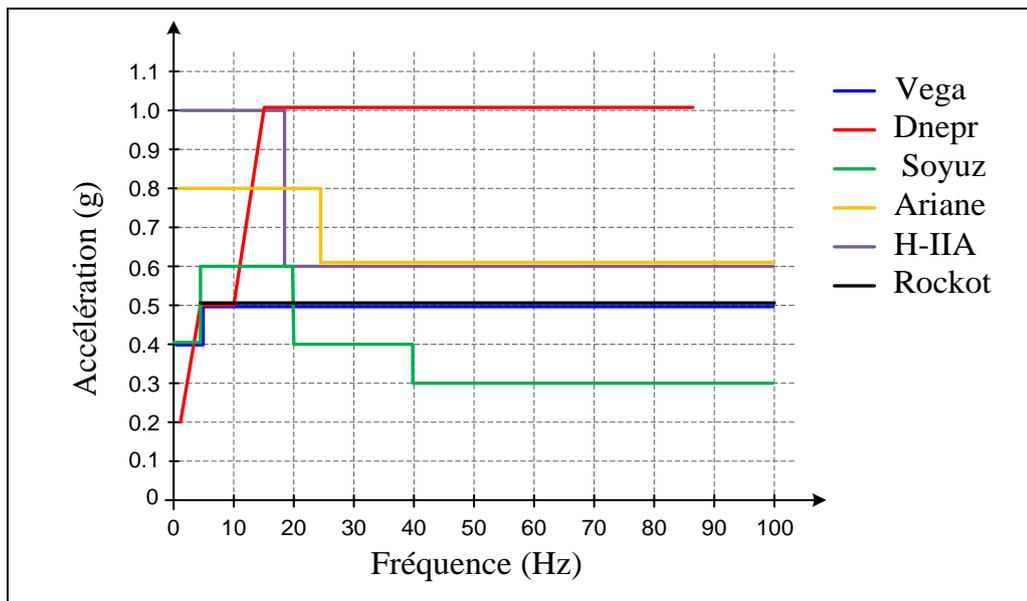


Figure 1-43 : Vibrations latérales sinusoïdales des différents lanceurs [41-47]

5.3. Spectre acoustique

Lors du lancement, des fluctuations de la pression acoustique sous le carénage du lanceur sont générées par le fonctionnement du moteur pendant le décollage, et par des phénomènes aérodynamiques instables pendant le vol atmosphérique. Le segment spatial doit satisfaire à des niveaux de spectres acoustiques dont les valeurs moyennes pendant le vol des différents lanceurs sont résumées par le *tableau 1.11* :

Bande d'octaves	31.5	63	125	250	500	1000	2000	Niveau pression globale
ARIANE 5	128	131	136	133	129	123	116	139.5
ROCKOT	123.3	125	127.4	126.7	124.4	120.7	111.1	137.9
H-IIA	125	126.5	131	133	128.5	125	120	137.5
SOYUZ	126	133	136	138	134	125	121	141.9
VEGA	110	123	126	135	138	127	120	140.3
DNEPR	125	132	135	134	132	129	126	140

Tableau 1-11 : Niveau de pression acoustique à l'intérieur du carénage (en dB) [41-47]

[Référence : 0 dB = $2 \cdot 10^{-5}$ Pa]

5.4. Les chocs mécaniques

Le segment spatial doit résister à un spectre de choc qualitatif dans toutes les directions des trois axes orthogonaux. Il doit être équivalent à une demi-sinusoïde dont l'impulsion dure 50 ms et avec une amplitude de 200 g (de 0 au pic).

Le comparatif des spectres de réponse au choc (valable au niveau de l'interface entre les satellites et le lanceur) pour les six lanceurs est donné par la *figure 1-44* suivante :

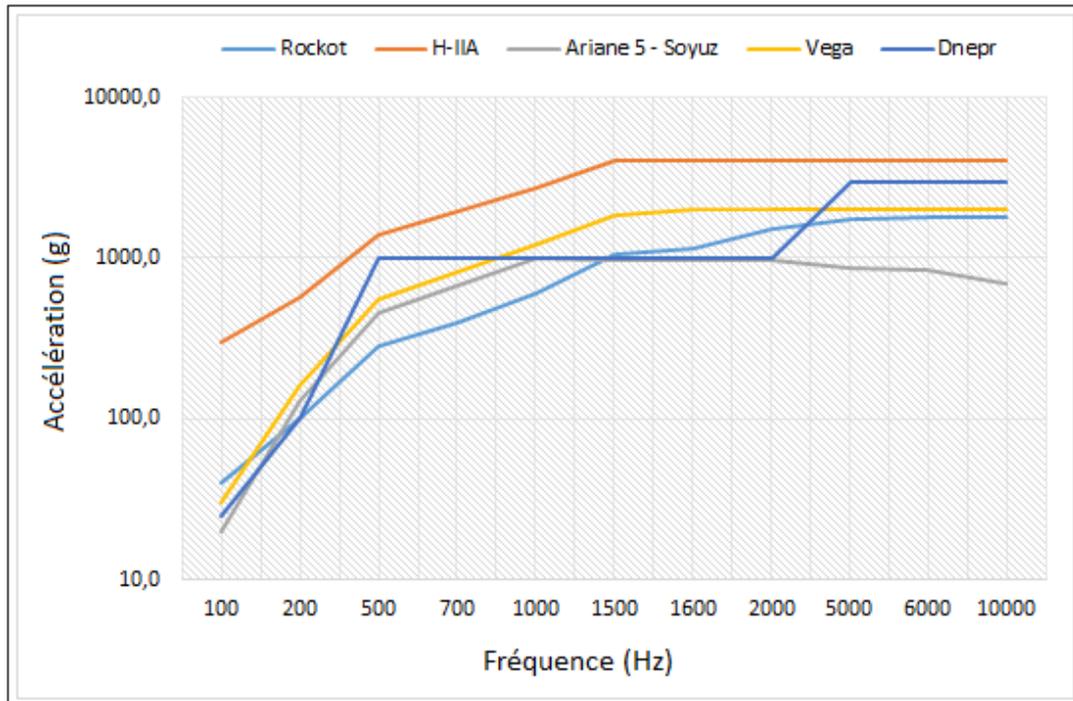


Figure 1-44 : Spectre de réponse aux chocs pour les différents lanceurs [41-47]

[Référence : 0 dB = $2 \cdot 10^{-5}$ Pa]

5.5. Pressions statiques

La cause de ces charges de pression statique est la dépressurisation lors de la montée du véhicule de lancement à travers l'atmosphère. Le taux de variation de la pression statique ainsi que sa valeur maximale sous le carénage des différents lanceurs sont résumés par la *figure 1-45* :

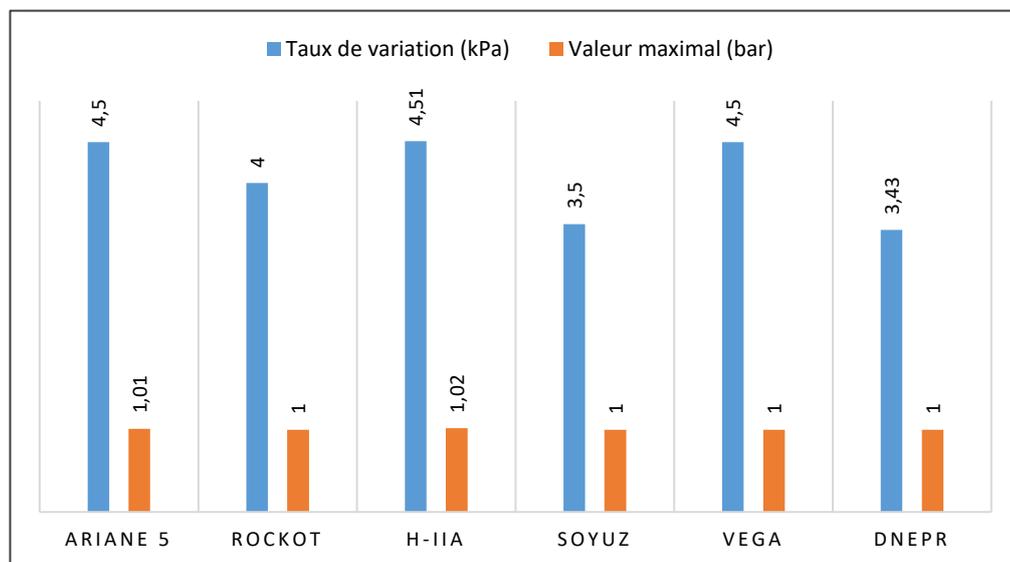


Figure 1-45 : Pression statique des différents lanceurs [41-47]

5.6. Facteurs de sécurité

Les valeurs données pour les différentes contraintes sont celles réellement mesurées lors d'un lancement. Il faut dès lors appliquer des facteurs de sécurité pour obtenir les niveaux de qualification sur lesquels les tests seront basés.

Pour que le segment spatial ait une marge de sécurité positive, les niveaux des tests sont déterminés par la multiplication des valeurs des différentes contraintes, préalablement définies, par des facteurs de sécurité. On distingue deux niveaux de test [10] :

- **Niveau de qualification** : c'est le plus éprouvant et il doit être réalisé par les concepteurs du segment spatial.
- **Niveau d'acceptation** : Il sera réalisé sur le modèle de vol et il est moins éprouvant afin de ne pas fragiliser la structure du segment spatial avant son intégration dans le lanceur.

Le *tableau 1-12* résume les facteurs de sécurité utilisés pour les différents types de contraintes de lancement :

Niveaux		Acceptation	Qualification	
			Modèle ingénieur	Modèle de vol
CQS	Facteur	1.0	1.25	1.25
	Durée	30 s	1 min	30 s
Vibrations Sinusoïdales	Facteur	1.0	1.25	1.25
	Vitesse de balayage	4 oct. /min	4 oct. /min	2 oct. /min
Vibrations Aléatoires	Facteur	1.0	+3 dB	+3 dB
	Durée	1 min /axe	2 min/axe	1 min/axe
Spectre Acoustique	Facteur	1.0	+3 dB	+3 dB
	Durée	1 min /axe	2 min/axe	1 min/axe
Choc mécanique	Facteur	1.0	1.4	1.4
	Durée	-	-	-

Tableau 1-12 : Niveaux d'acceptation et de qualification [41-47]

6. CONTRAINTES THERMIQUES

Les circuits électroniques doivent être choisis pour pouvoir travailler dans une large plage de température imposée par l'environnement spatial de fonctionnement (approximativement comprise entre -40°C et 85°C). En effet :

- Pendant la phase de préparation au lancement, l'environnement thermique typique dans des lanceurs comme Vega, Soyouz et Ariane 5 VL est maintenue autour de $23^{\circ}\text{C} \pm 2^{\circ}\text{C}$ pour une humidité relative de $55\% \pm 5\%$ [36].
- Avant le largage du carénage qui protège la charge utile, la densité de flux thermique rayonnée par le carénage ne dépasse pas en tout point 1000 W/m^2 pour Vega et Ariane 5, et 800 W/m^2 pour Soyouz [36].
- Après le largage du carénage qui protège la charge utile, le flux aérothermique varie typiquement de 1135 W/m^2 à moins de 200 W/m^2 dans les 20 secondes après le largage comme le montre la *figure 1-46* :

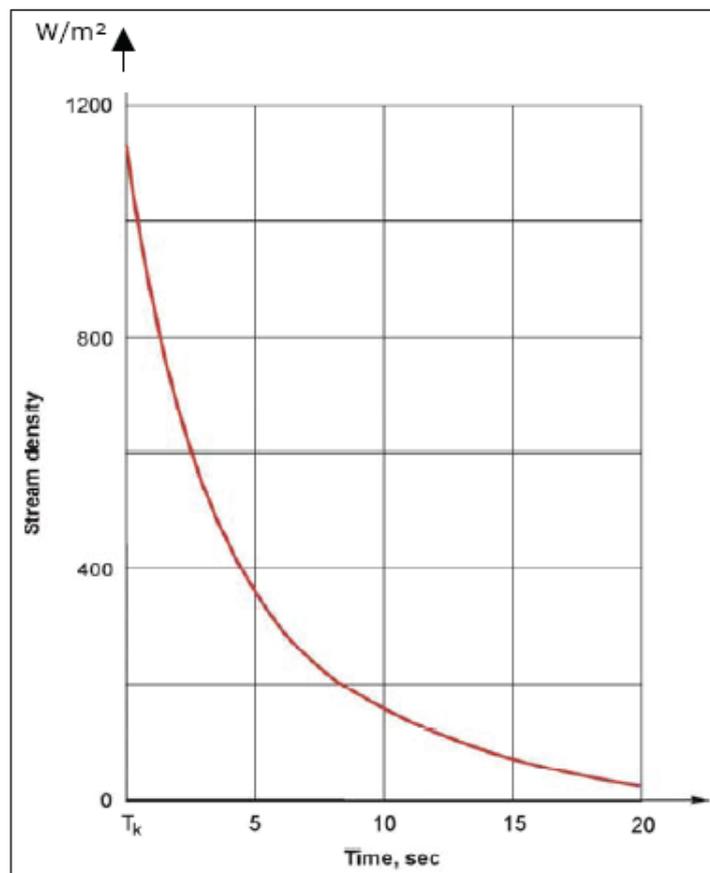


Figure 1-46 : Flux aérothermique typique après le largage du carénage [36]

- En orbite et pendant la phase d'éclipse, les composants ne seront pas soumis à un changement rapide de température puisque la diminution de température se fera uniquement par rayonnement (échange thermique lent).
- En orbite et pendant la phase d'ensoleillement, le segment spatial sera exposé à 1300 W/m^2 et les températures sur les faces du cubesat devraient atteindre les 80°C . La structure permettra alors de contrôler la conduction de la chaleur à l'intérieur du cubesat.

7. CONCLUSION

Comme conclusion à cette première étape concernant l'analyse de la mission MASAT1, on va donner une synthèse des exigences, des contraintes et des paramètres qui vont être déterminants dans la conception des sous-systèmes du segment spatial :

- Les paramètres orbitaux des différents scénarios identifiés pour la mission MASAT1 sont :
 - Altitude : 400 km – 500 km – 650 km
 - Inclinaison : 40° - 65° - 90°
- La durée de vie de la mission est estimée à 1.5 ans pour une altitude minimale de 400km et à 5.6 ans pour une altitude maximale de 650km.
- Le segment spatial orbitera en moyenne 15 fois autour de la terre avec, en moyenne, 5 occasions pour entrer en contact avec la station au sol d'Ifrane.
- La durée d'éclipse maximale que peut connaître le segment spatial avoisine les 36 minutes, avec 4766 cycles charge/décharge batterie par an.
- Une augmentation de l'épaisseur de blindage du Cubesat de 1 mm à 2 mm permet de diminuer la dose des radiations reçues de 2.94.
- Les composants électroniques doivent être choisis pour pouvoir fonctionner dans une plage de température comprise entre -40°C et +80°C.

Chapitre 2

ARCHITECTURE DE LA PLATEFORME MASAT1

1. INTRODUCTION

En se basant partiellement sur les directives du manuel de l'ingénierie système de la NASA [49], on a adopté une approche de conception descendante permettant de traduire les objectifs et les exigences de la mission, en sous-systèmes, composants matériels, interfaces et tâches logiciels. La conception du segment spatial passera ainsi par quatre processus interdépendants qui vont aboutir à des solutions de conception validées conformément aux objectifs de la mission (voir *figure 2-1*) :

- **Processus de description de la mission** qui permet la définition des objectifs, des attentes et des engagements de la mission MASAT1, en plus de la conception de la structure de décomposition du projet (voir Chap.1, § 2.2).
- **Processus de définition des exigences techniques** de la mission MASAT1 qui est itératif durant toute la phase de conception préliminaire (voir Chap.1, § 2.3).
- **Processus de décomposition logique et fonctionnelle** de la mission MASAT1 qui aboutira, via une boucle de conception itérative, à adopter une architecture pour le système ainsi que le concept des opérations (ConOps) qui va servir de base pour la planification opérationnelle et pour la conception du logiciel de vol.
- **Processus de définition des solutions de conception** dont la finalité est d'aboutir à une conception préliminaire du segment spatial, qui doit être fonctionnelle, fiable, abordable et répondante aux objectifs de la mission.

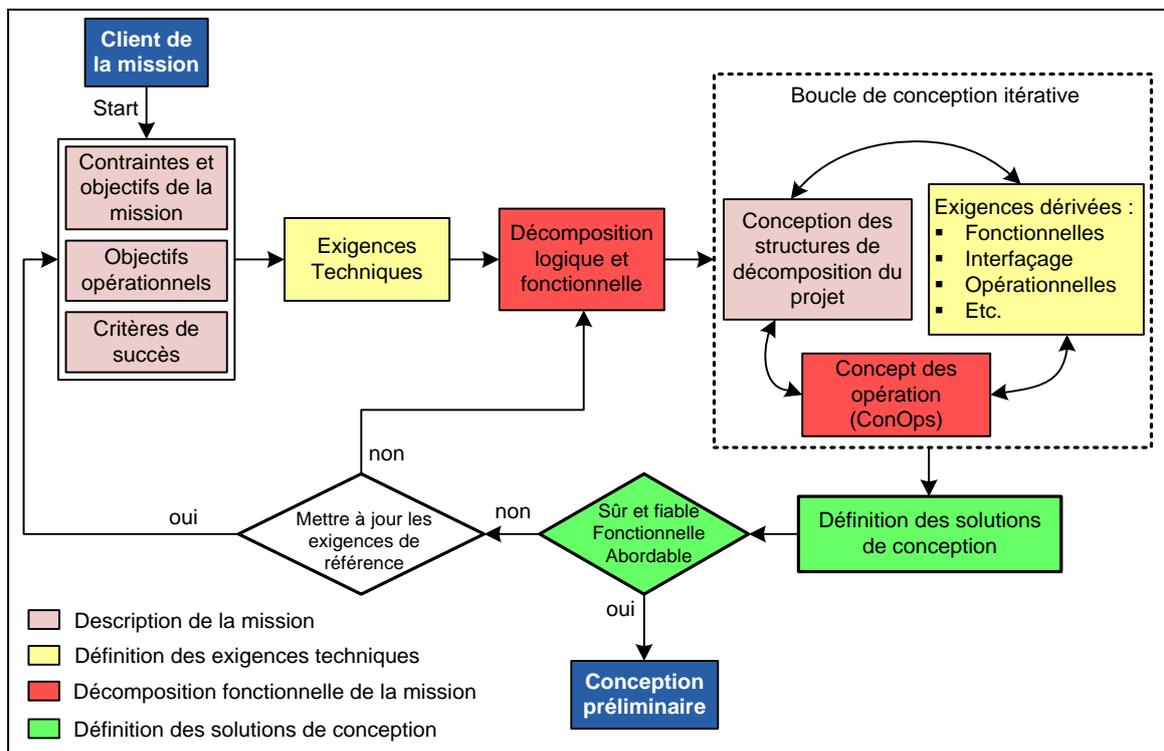


Figure 2-1 : Interrelations entre les processus de conception du système

2. DECOMPOSITION FONCTIONNELLE DE MASAT1

Ce processus constitue le point de départ de la formulation d'une architecture conceptuelle du segment spatial qui permettra de définir clairement le rôle, la relation et les interfaces entre les différents sous-systèmes [49]. Ces derniers peuvent ainsi être développés séparément les uns des autres, tout en s'assurant que leur intégration permettra d'atteindre les objectifs de la mission.

En se basant sur le processus de description de la mission et celui de définition des exigences techniques (Chap. 1 § 2.2 et § 2.3), nous avons réalisé une analyse fonctionnelle en deux étapes :

- Identification des fonctions à réaliser par chacun des sous-systèmes du segment spatial, ainsi que l'interaction en eux. La *figure 2-2* représente le diagramme bloc fonctionnel adopté pour la mission MASAT1.
- Identification et description des interfaces fonctionnelles (électriques et mécaniques) entre les différents sous-systèmes, en utilisant le diagramme N² ou N2 qui met en évidence les hypothèses et les exigences de dépendance entrées/sorties entre sous-systèmes, et permet également d'identifier des zones d'interfaces où des conflits pouvant survenir [49]. La *figure 2-3* représente le diagramme N² de la mission MASAT1.

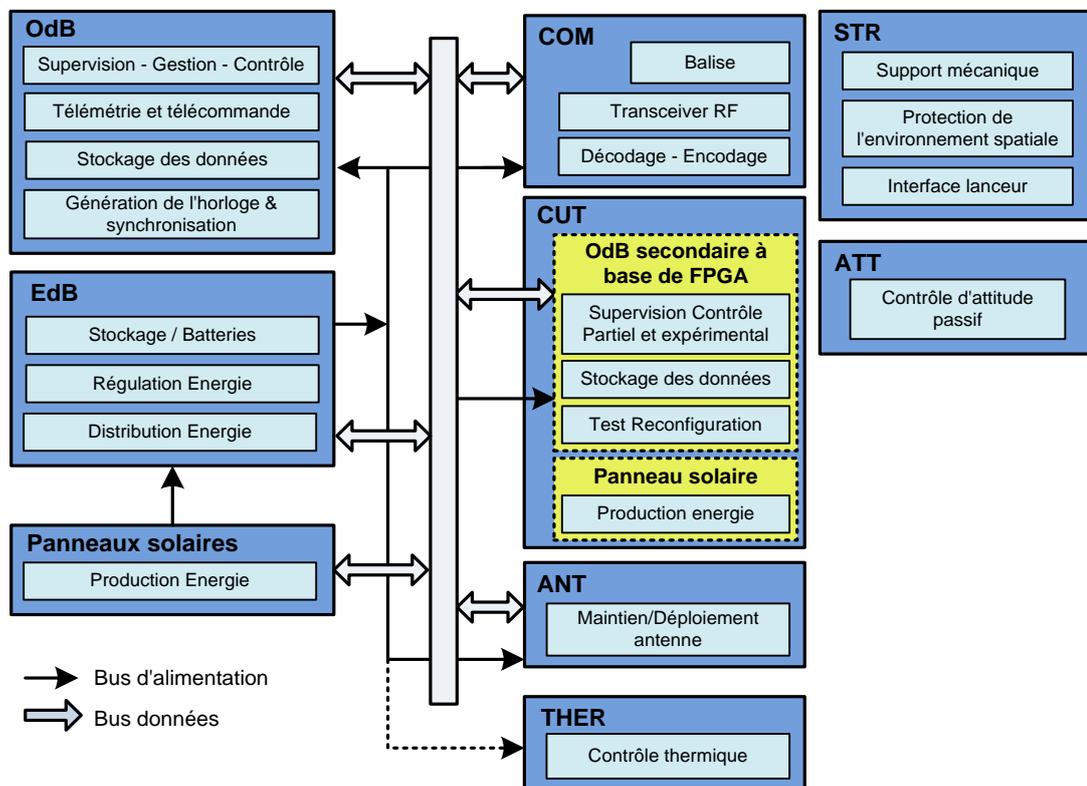


Figure 2-2 : Diagramme bloc fonctionnel de la mission MASAT1

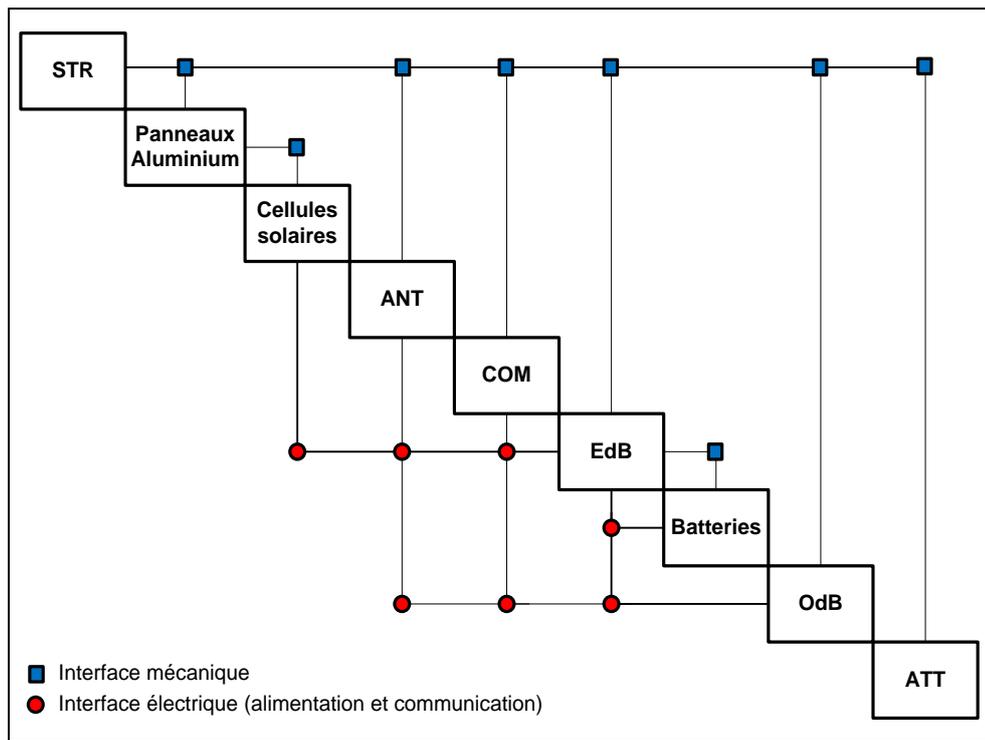


Figure 2-3 : Diagramme N² de la mission MASAT1

Les détails sur les exigences techniques et fonctionnelles seront abordés dans la description des solutions de conception adoptées pour chacun des sous-systèmes du segment spatial.

3. DEFINITION DES SOLUTIONS DE CONCEPTION DE MASAT1

Ce processus permet de transformer la modélisation de la mission, réalisée pendant le processus de décomposition fonctionnelle, en solutions de conception alternatives. Ces solutions techniques sont ensuite analysées à travers une étude comparative détaillée pour aboutir à la solution référence qui sera adoptée pour la conception préliminaire du segment spatial. Dans cette partie, on va aborder les solutions de conception références adoptées au niveau du segment spatial de MASAT1.

3.2. Architecture du segment spatial

Il y a deux types d'architectures qui peuvent être utilisées pour notre segment spatial :

- **Architecture centralisée** où l'OdB est le nœud central du système. Il est équipé d'un microcontrôleur qui va lui permettre de centraliser le contrôle de tous les autres sous-systèmes, d'effectuer tous les transferts de données entre ces sous-systèmes et de prendre en charge les communications avec la station au sol. L'OdB est directement connecté aux sous-systèmes avec lesquels il va interagir. Cette connexion se fait via des lignes d'entrée/sortie dédiées (voir *figure 2-4*).

- Architecture distribuée** où l’OdB n’est plus le nœud central du système. Au lieu de cela, chaque sous-système est équipé d’un microcontrôleur local qui va gérer les tâches propres au sous-système ainsi que toutes les interactions directes avec l’OdB. Tous les microcontrôleurs sont alors liés à un bus de communication série comme le montre la *figure 2-4*.

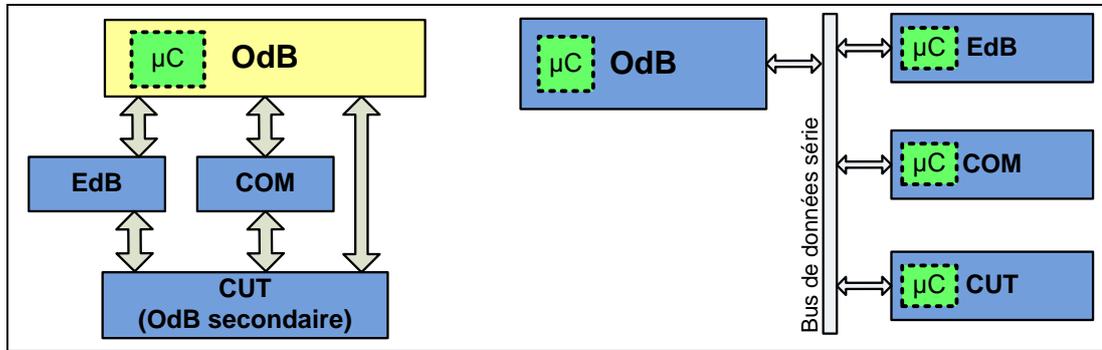


Figure 2-4 : Architecture centralisée (à droite) et distribuée (à gauche)

Pour pouvoir choisir, on a identifié les avantages et les inconvénients des deux architectures en étudiant les solutions adoptées par d’autres missions Cubesat [50-53] :

ARCHITECTURE CENTRALISEE	Avantages	<ul style="list-style-type: none"> Peu de problèmes de communication entre les sous-systèmes ce qui augmente la fiabilité. Conception matérielle simple des sous-systèmes et consommation d’énergie réduite.
	Inconvénients	<ul style="list-style-type: none"> Point de défaillance unique et majeure à savoir l’OdB lui-même. Architecture peu souple et peu évolutive. En effet, le simple fait d’ajouter un signal de mesure ou de contrôle impliquera une modification matérielle au niveau de l’OdB et du sous-système qui va évoluer. Problème d’interférences électromagnétiques causées par tous les signaux analogiques qui vont circuler entre les différents sous-systèmes et l’OdB. La puissance de traitement n’est pas partagée entre les sous-systèmes ce qui entraîne une conception logicielle complexe.
ARCHITECTURE CENTRALISEE	Avantages	<ul style="list-style-type: none"> Architecture modulaire et évolutive avec la possibilité de remplacer un sous-système en cas de panne ou d’amélioration. Le système ne dépend pas entièrement de l’OdB vue qu’il y a plusieurs voies de communication possibles. Les systèmes sont indépendants dans la conception et le test. Ce qui correspond mieux à l’organisation du projet MASAT1. La puissance de traitement est partagée entre les sous-systèmes ce qui entraîne une conception logicielle moins complexe.
	Inconvénients	<ul style="list-style-type: none"> Conception matérielle plus complexe des sous-systèmes et consommation d’énergie plus élevée. Point de défaillance unique ou point critique au niveau du bus série de communication ce qui diminue la fiabilité. Débogage du système complexe.

Tableau 2-1 : Avantages et inconvénients des architectures distribuée et centralisée

La solution de référence adoptée est l'architecture distribuée pour les raisons principales suivantes :

- Elle est moins complexe en termes de câblage à l'intérieur du segment spatial et ne va pas nécessiter un grand nombre de ports dédiés au niveau des microcontrôleurs.
- L'OdB n'est plus un point singulier de défaillance qui entrainera obligatoirement la défaillance de tout le système et/ou la perte de contact avec le segment spatial.
- La modularité au niveau de la conception matérielle et logicielle du système, ce qui est adapté à l'organisation du projet MASAT1 vu que les sous-systèmes seront développés par des structures universitaires indépendantes.
- Le développement logiciel au sein de chaque sous-système devient moins complexe. En effet, d'une part, le logiciel dans les microcontrôleurs sera plus petit et moins complexe, ce qui permet une meilleure vue d'ensemble de son comportement ; et d'autre part, il y a la possibilité de partager des segments de code entre les microcontrôleurs des sous-systèmes puisque certaines de leurs fonctionnalités sont identiques, comme par exemple, la communication avec le bus de données.
- Testabilité indépendante des sous-systèmes. En effet, puisque chaque sous-système possède une interface matérielle standard, il est possible de développer une plateforme de test générique comportant un ordinateur (ou des modules embarqués comme Raspberry Pi et Arduino) connecté à un sous-système via le bus de données série.

3.3. Bus de communication

L'architecture distribuée adoptée permet l'utilisation d'un bus de communication série unique qui permet aux sous-systèmes d'utiliser le même protocole. Ce bus doit être fiable, fournir un débit suffisant et avoir un protocole de communication qui gère une topologie multipoint avec une méthode d'accès multi-maître/esclave.

3.3.1. Liste de sélection

Pour assurer un minimum de fiabilité, le choix du bus de communication série se fera parmi le bus I2C (Inter-Integrated Circuit) ou le bus CAN (Controller Area Network) qui sont les plus largement utilisés dans des projets de CubeSat antérieurs, comme le montre la *figure 2-5* qui résume le benchmark fonctionnel réalisé en *ANNEXE C* et qui concerne 39 missions de nanosatellites lancés depuis l'année 2003 :

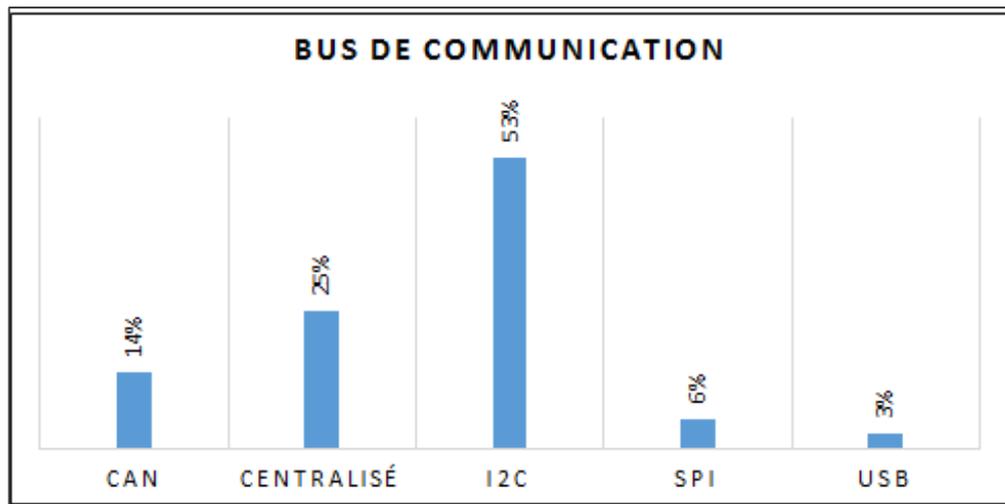


Figure 2-5 : les bus de communication série utilisés dans les missions Cubesat

a. Bus I2C

Le bus I2C est un bus de communication série, non différentiel, synchrone et bidirectionnel utilisant deux fils : une ligne d'horloge SCL (Serial Clock Line) et une ligne de données SDA (Serial Data Line) [54] (voir **Figure 2-6**). Pour certaines bibliothèques logicielles et certains constructeurs, ce bus porte parfois le nom de TWI (Two Wire Interface).

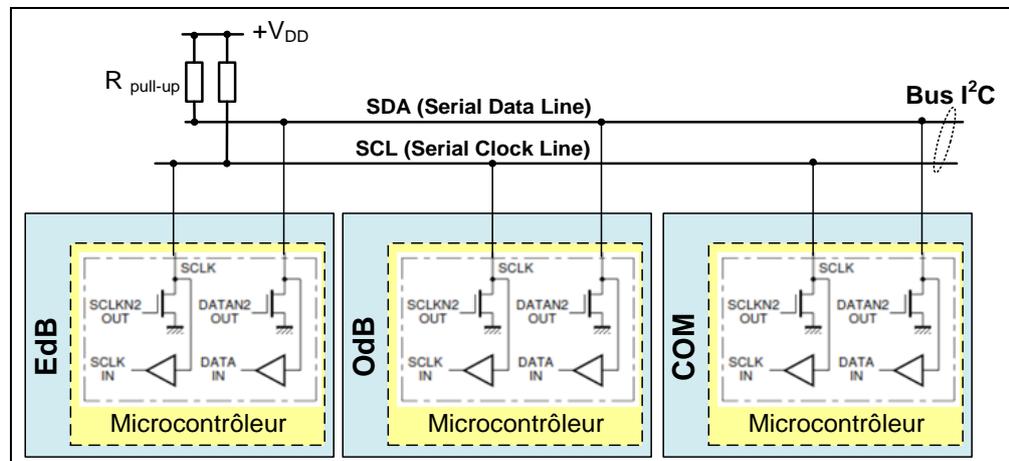


Figure 2-6 : Dispositif de connexion standard pour le bus I2C

La communication se fait uniquement en mode half-duplex. L'interface logicielle de communication utilise des mots de 8 bits avec une adresse unique pour chaque nœud du bus. Ces nœuds peuvent transmettre ou recevoir, et fonctionner soit en maître ou en esclave sachant que le nœud maître contrôle l'horloge. Ce bus est principalement caractérisé par :

- Peu de robustesse aux perturbations.
- Débit de transmission de données pouvant atteindre 3,2 Mbits/s (425 ko/s).
- Faible Consommation d'énergie.
- La plupart des microcontrôleurs possède un module de communication série I2C intégré, ce qui rend plus facile le développement logiciel.
- Les capteurs peuvent être reliés directement au bus (via des convertisseurs de données analogique numérique).

b. Bus CAN

Le bus CAN est un bus de communication série, différentiel, asynchrone et bidirectionnel qui opère uniquement en mode half-duplex. Il fonctionne en topologie BUS (mode d'accès multi-maîtres) avec l'utilisation de la notion d'objets ou de nœuds plutôt que d'adresses. Lorsque le bus est libre, chaque nœud peut décider d'envoyer un message et seul le message de plus haute priorité prend possession du bus. Chaque nœud est connecté au bus par l'intermédiaire d'une paire de fils équilibrée (CAN_H et CAN_L) [55] :

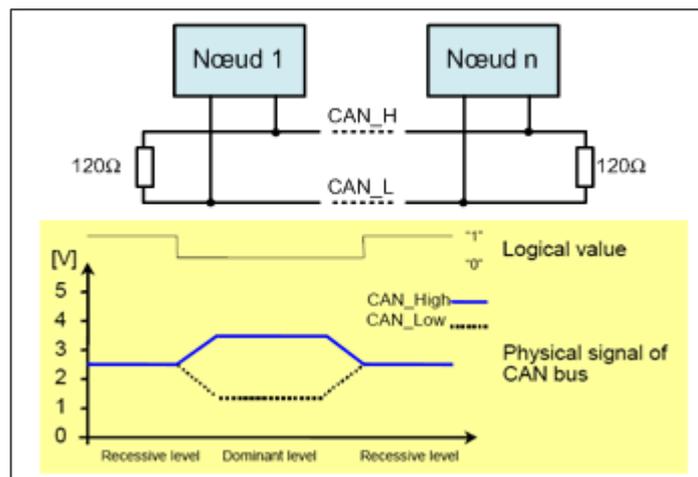


Figure 2-7 : Dispositif de connexion pour le bus CAN [ISO11898 High Speed]

L'interface de communication logicielle utilise un codage NRZ assurant des messages compacts avec un nombre minimum de transition. Ce bus est principalement caractérisé par :

- Plus de robustesse aux perturbations électromagnétiques.
- Plus grande consommation d'énergie.
- Débit de transmission de données pouvant atteindre 1 Mbits/s.
- Conception matérielle plus complexe avec l'utilisation obligatoire de Transceiver pour l'interfaçage entre les microcontrôleurs (équipé de module intégré CAN) et le bus de communication bifilaire.

3.3.2. Critères de choix

Chaque critère de choix adoptée est assorti d'un "poids" en fonction de son importance :

- **La fiabilité** : Ce critère de comparaison est basé sur l'immunité aux interférences électromagnétiques et les possibilités offertes pour vérifier l'avènement d'erreurs sur le bus, et pour gérer la priorité si plusieurs nœuds transmettent en même temps. Il a un poids égal à 3.
- **Consommation d'énergie** : C'est un critère très important vu les contraintes de notre système du point de vue puissance disponible. Il a un poids égal à 5.
- **Disponibilité sur le marché** : Ce critère indique si le bus est couramment utilisé ou non et s'il a été utilisé dans d'autres missions spatiales. Il a un poids égal à 4.
- **Débit de données** : Ce critère de comparaison reste relativement moins important puisque dans un premier temps, notre charge utile ne nécessite pas un débit important. Il a un poids égal à 2.
- **Mise en œuvre** : Ce critère de comparaison est basé sur la facilité de mise en œuvre matérielle et logicielle, ainsi que sur la disponibilité des outils de développement. Il a un poids égal à 4.

Le *tableau 2-2* présente le résultat de la comparaison des bus I2C et CAN :

Critères	Référence	Poids	I ² C	CAN
Fiabilité	Différentiel/CRC	3	1	3
Consommation	Active 25mW / Standby 5mW	5	3	1
Disponibilité	Industrie et CubeSat	4	3	2
Débit	1 Mbits/s	2	2	2
Mise en œuvre	Hard – Soft – Outils	4	3	2
		Résultat	46	34

Tableau 2-2 : Comparaison des bus de communication série I2C et CAN

Conformément aux résultats obtenus, le bus I2C semble le meilleur choix pour notre topologie vu qu'il est largement utilisé sur d'autres CubeSat en offrant la plus faible consommation de puissance. Mais ce choix présente 3 inconvénients :

- Peu de robustesse aux perturbations.
- Aucune protection de bus contre les nœuds défaillants.
- Absence d'un moyen de détection d'erreurs pour le protocole bas niveau I2C.

On doit alors prévoir dans la conception du segment spatial plusieurs solutions pour remédier à ces problèmes comme :

- Ajouter un bus de communication redondant et/ou un bus de communication dédiés liant les sous-systèmes cruciaux.
- Implémenter, au niveau logiciel, un protocole haut niveau avec une stratégie de correction et de détection d'erreurs.
- Augmenter la fiabilité matérielle du bus en utilisant par exemple :
 - Des temporisateurs (chien de garde) pour assurer une remise à zéro matérielle d'un nœud défaillant.
 - Des circuits protecteurs de bus au niveau de chaque nœud pour pouvoir isoler un sous-système du bus en cas de court-circuit comme l'exemple de la **figure 2-8**.

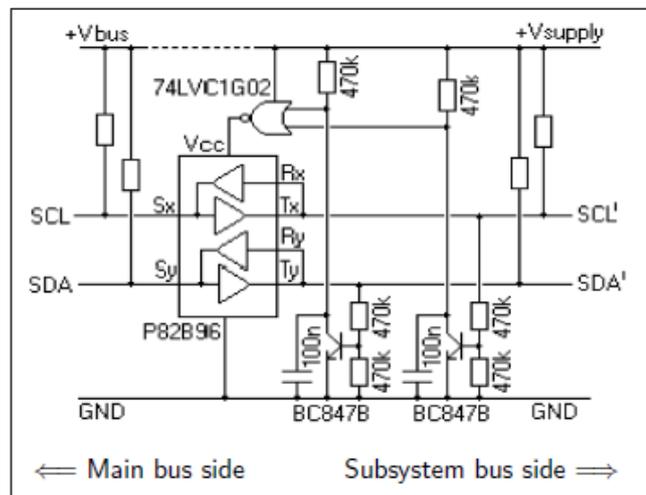


Figure 2-8 : Circuit protecteur de bus I2C à base du P82B96 [56]

3.4. Structure et configuration

En raison des objectifs et du budget financier limité de la mission, le choix d'une structure pour un Cubesat 1U a été jugé suffisant tout en maximisant l'espace intérieur utilisable, et en minimisant la complexité et le coût de la configuration choisie.

3.4.1. Le référentiel du segment spatial

Un référentiel doit être défini pour la structure afin de pouvoir donner l'emplacement exact de chaque composant du segment spatial. Son origine (point de référence) est choisie au centre géométrique du "cube". En se basant sur le modèle référence de la structure 1U du fournisseur Pumpkin [145], le cadre est orienté de telle manière que :

- La face avec les ports de communication et d'alimentation soit normale à l'axe X avec des coordonnées X négatives.
- La base du cube (face contenant les pieds et les commutateurs) soit normale à l'axe Z avec des coordonnées Z négatives.

La **figure 2-9** montre le point de référence et le système d'axe X, Y et Z du référentiel choisi pour la structure du segment spatial :

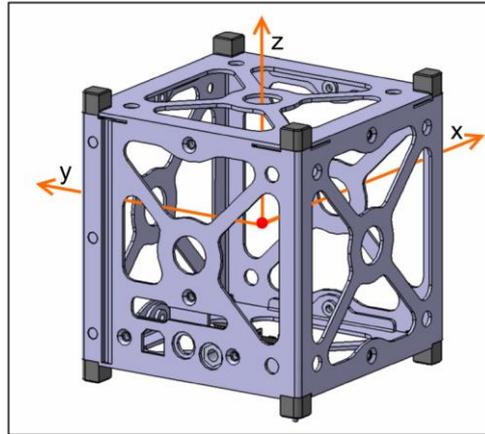


Figure 2-9 : Cadre de référence du Cubesat

3.4.2. Configuration

La configuration interne du segment spatial est un aspect très important dans la conception structurelle du CubeSat. Le choix de l'emplacement des éléments du segment spatial doit vérifier les exigences suivantes :

- Les spécifications de la norme CubeSat qui indiquent que le centre de gravité du nanosatellite doit être autour de 2cm de son centre géométrique. Par conséquent, le poids doit être bien distribué à l'intérieur du nanosatellite [10].
- L'accès aux composants électriques doit être facilité pour permettre d'économiser du temps pendant la phase de développement final du segment spatial (remplacement des cartes électroniques, débogage, alimentation), ainsi que la phase de préparation au lancement.

Il existe essentiellement deux types de configurations possibles :

- Arrangement verticale dans des "slot" avec l'utilisation d'une carte mère où sont attachées toutes les cartes électroniques des sous-systèmes. C'est la solution adoptée par la mission COMPASS-1 [57] et elle présente l'avantage que toutes les connexions entre cartes passent à travers la carte mère et on peut démonter n'importe quelle carte indépendamment des autres. En plus, l'ensemble des sous-systèmes électroniques peuvent être démontés de la structure principale en une seule opération. L'inconvénient de cette stratégie est l'obligation de concevoir une carte électronique supplémentaire à savoir la carte mère (voir **figure 2-10**).

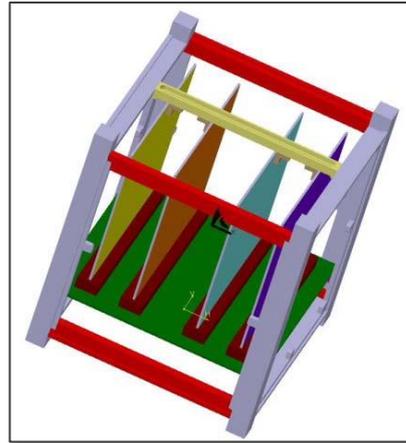


Figure 2-10 : Exemple de configuration en "slots" [57]

- Une disposition en couches où les cartes électroniques sont disposées en pile, l'ensemble formant une structure secondaire qui est fixée à la structure principale. C'est la solution adoptée par la mission OUFTI [62] et elle présente l'avantage d'augmenter la rigidité de la structure, avec en contrepartie, la difficulté d'établir les connexions entre les cartes non limitrophes (voir *figure 2-11*).

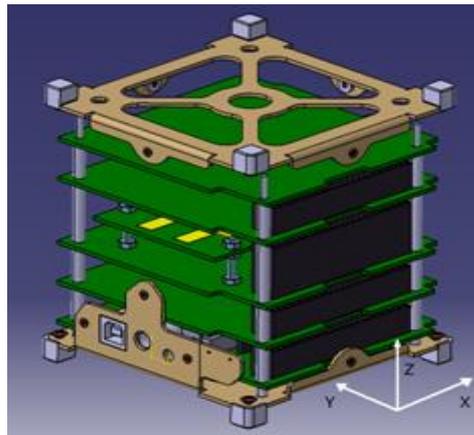


Figure 2-11 : Exemple de configuration en couches [62]

Vu que la charge utile de la mission est constituée d'une carte électronique et d'une cellule solaire expérimentale, elle n'impose pas des contraintes spécifiques au niveau de l'allocation d'espace. On a alors adopté une disposition en couches pour notre configuration.

Pour résoudre le problème d'interconnexion entre les différentes cartes électroniques, ces dernières seront empilées les unes sur les autres en suivant le format PC-104 comme le montre la *figure 2-12* :

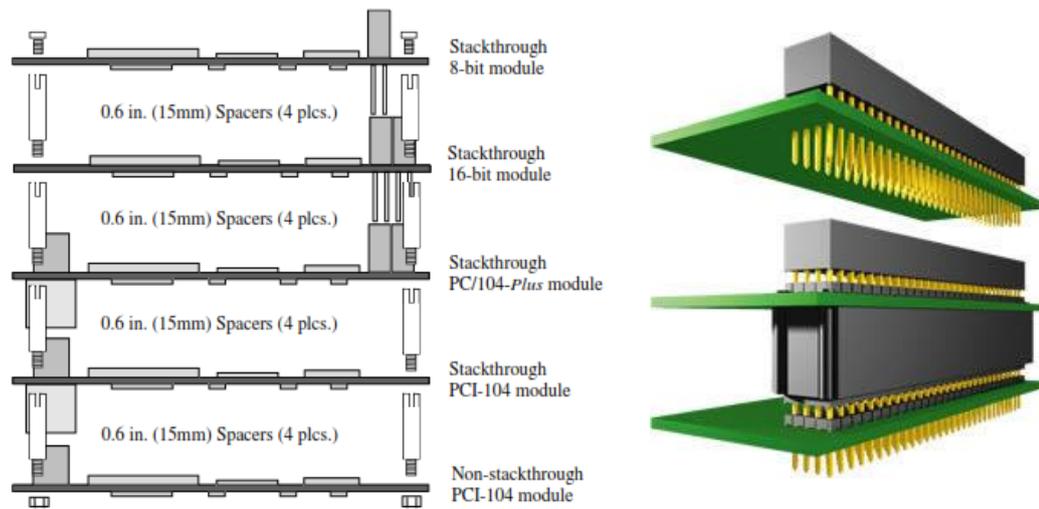


Figure 2-12 : configuration en pile selon la norme PC-104 [58]

La spécification PC-104 a été adoptée comme document de base pour le projet de norme IEEE P996.1 applicable pour les modules compacts embarqués. Elle est caractérisée par [58] :

- Un bus standard composé de 104 points (un connecteur de 64 points et un de 40 points).
- Une taille réduite de 90,17 x 95,89 mm (soit 3,55 x 3,775 pouces).
- Une consommation d'énergie réduite.

La norme PC-104 impose la consommation, la taille, le bus standard, mais pas l'architecture. Les cartes électroniques seront maintenues en place par quatre vis sans fin verticales. Pour satisfaire l'exigence du centre de gravité, les batteries (qui constituent les composants les plus lourds) seront placées avec la carte électronique EdB, à proximité du centre de la structure comme le montre la *figure 2-13*.

Cinq des six faces seront couvertes par des cellules solaires fixes sur des panneaux en aluminium (protection contre les radiations). La sixième face sera consacrée aux ports externes et au mécanisme de déploiement des antennes.

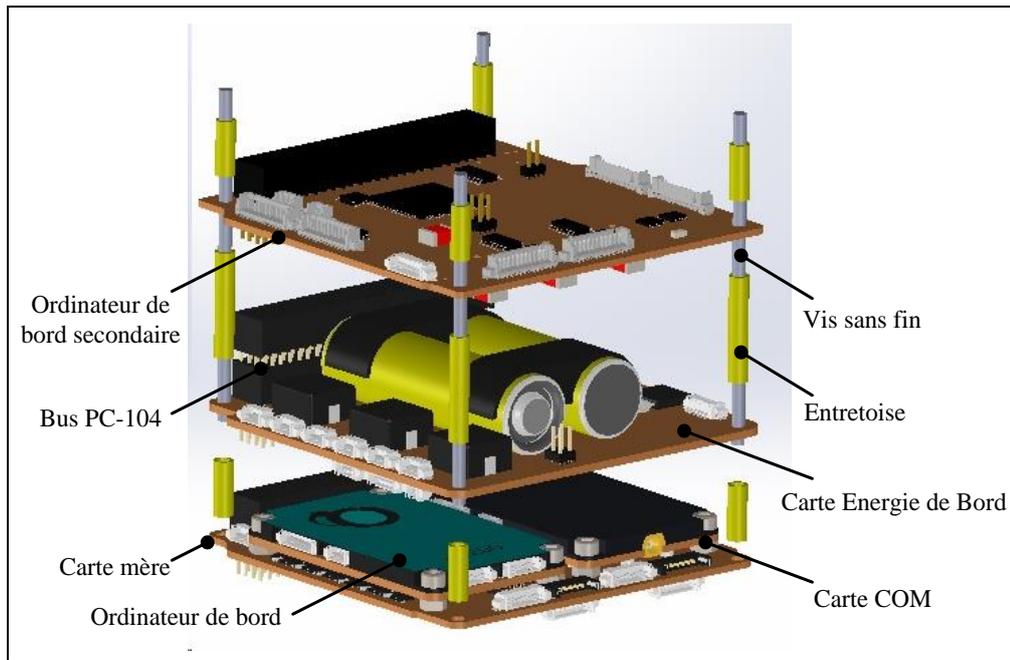


Figure 2-13 : Configuration de MASAT1 [SOLIDWORKS]

3.5. Communication

La communication entre le segment spatial et la station au sol consiste en deux types de liaisons :

- Une liaison montante permettant au segment spatial de recevoir les télécommandes (TC) provenant de la station au sol, afin de les exécuter via l'OdB.
- Une liaison descendante permettant à la station au sol de recevoir toutes les données de télémétrie (TM) et de la charge utile provenant du segment spatial, pour les traiter.

Pour les bandes de fréquences à utiliser pour ces deux liaisons, elles sont généralement situées dans le spectre radio allant d'environ 30MHz à 40GHz. Elles sont définies par IEEE depuis 2009 et résumées par la *figure 2-14* [59] :

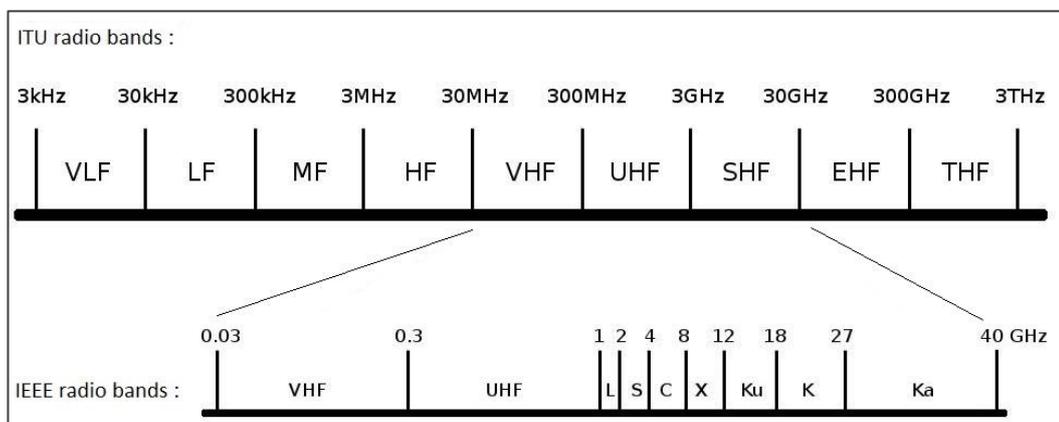


Figure 2-14 : Spectre radio utilisé pour la communication des engins spatiaux [3]

La plupart des cubesats utilisent les bandes radiofréquence amateur VHF et UHF car leurs licences sont libres et rapidement allouées par l'AMSAT (Radio Amateur Satellite Corporation) [60]. L'IARU (International Amateur Radio Union) et l'AMSAT ont publié un document permettant de normaliser l'attribution à l'échelle internationale de fréquences pour les services amateurs par satellite. Le **tableau 2-4** résume les bandes de fréquences sous licence attribuées à l'échelle internationale et répertoriées comme exclusives pour les communications des satellites amateurs Cubesat [61] :

Bande fréquence	VHF	UHF	L	S
Fréquences	144 – 146 MHz	435 – 438 MHz	1260 – 1270 MHz	2.4 – 2.45 GHz
Longueur d'onde	2 m	70 cm	24 cm	12.5 cm

Tableau 2-3 : Bandes de fréquences disponibles pour les CubeSat [61]

Pour le mode de communication entre le segment spatial et la station au sol, le choix a donc été fait parmi les trois alternatives suivantes :

- 1) Un système de communication simplex avec une seule fréquence radio VHF ou UHF à la fois pour la liaison montante et celle descendante. Cette solution présente l'avantage de réduire la taille, la masse et la puissance moyenne consommée par le sous-système COM [62], mais elle a été écartée pour deux raisons :
 - La première concerne la réglementation ITU (International Telecommunication Union) qui stipule à l'article 22.1 que *"les stations spatiales doivent être munis de dispositifs permettant la cessation immédiate de leur émission, chaque fois que cette cessation est requise en vertu des dispositions de ces règlements"* [63]. Or une communication simplex pourrait ne pas être considérée comme satisfaisante à cette exigence, surtout si le système est bloqué dans son état d'émission [62].
 - La deuxième concerne la non disponibilité chez les fournisseurs spécialisés dans les sous-systèmes pour Cubesat, de modules COM offrant le mode de communication simplex.
- 2) Un système de communication full-duplex à deux fréquences radio :
 - Une fréquence UHF pour la liaison descendante permettant d'obtenir plus de bande passante pour la transmission de données, et une antenne de taille plus petite au niveau du segment spatial.
 - Une fréquence VHF pour la liaison montante qui est la bande radio amateur primaire.

Ce choix reste intéressant si les liaisons montantes sont principalement utilisées pour les TC et donc un faible débit de données (faible bande passante) est suffisant.

- 3) Un système de communication half-duplex à deux fréquences radio UHF :
- Une fréquence UHF pour la liaison descendante permettant d'obtenir plus de bande passante pour la transmission de données, et une antenne de taille plus petite au niveau du segment spatial.
 - Une fréquence UHF pour la liaison montante sachant que cette bande de fréquence est principalement utilisée par les radars côtiers, ce qui pourrait brouiller le signal des liaisons montantes. Mais ce choix peut être intéressant si on prévoit d'utiliser une charge utile nécessitant un haut débit (forte bande passante) pour tout envoi de données depuis la station au sol.

Vu la nature d'une des charges utiles de la mission MASAT1 qui prévoit la conception d'un ordinateur de bord secondaire reconfigurable à base de FPGA, notre choix s'est porté sur un mode de communication half-duplex à deux fréquences radio UHF. En effet, l'envoi des fichiers de reconfiguration depuis la station au sol nécessite un haut débit de données pour la liaison montante.

En attendant d'entrer en contact avec l'Agence Nationale de Réglementation des Télécommunications (ANRT) et avec un membre du Groupe consultatif IARU pour valider notre choix des fréquences de communication, nous avons décidé d'utiliser deux canaux de communications dans la bande de fréquences UHF (435 – 438 MHz) :

- 1 canal de communication en Réception : Récepteur UHF
- 1 canal de communication en Transmission : Emetteur UHF – Balise

3.6. L'approche des composants COTS pour le choix des solutions technologiques

Si on se base sur la tolérance aux radiations pour la catégorisation des composants susceptibles d'être utilisés pour les missions spatiales, on trouve trois catégories [64] :

- **Composants COTS :**
 - Le processus de conception limite la résistance aux radiations.
 - Niveaux typiques de tolérance contre les radiations :
 - Dose TID : 2 à 10 krad
 - Seuil LET des SEU : 5 MeV/mg/cm²
 - Taux des erreurs SEU : 10⁻⁵ erreurs/jour-bit
 - C'est l'utilisateur qui évalue et assume les risques contre les radiations.

- **Composants tolérants aux radiations :**
 - Le processus de conception assure une résistance aux radiations jusqu'à un certain niveau.
 - Niveaux typiques de tolérance contre les radiations :
 - Dose TID : 20 à 50 krad
 - Seuil LET des SEU : 20 MeV/mg/cm²
 - Taux des erreurs SEU : 10⁻⁷ à 10⁻⁸ erreurs/jour-bit
 - Habituellement testé par le fabricant pour les défaillances fonctionnelles.
 - Evaluation des risques par l'utilisateur.
- **Composants résistants aux radiations :**
 - Le processus de conception assure un niveau élevé de résistance contre les radiations.
 - Niveaux typiques de tolérance contre les radiations :
 - Dose TID : de 200 krad à 1Mrad
 - Seuil LET des SEU : 80 - 150 MeV/mg/cm²
 - Taux des erreurs SEU : 10⁻¹⁰ à 10⁻¹² erreurs/jour-bit
 - Test des lots de substrat (Wafer) contre les radiations.

Les satellites utilisent généralement les composants résistants aux radiations qui sont plus coûteux et spécialement conçus pour les applications spatiales. Mais les missions Cubesat ont démontré l'utilisation avec succès des technologies basées sur les composants COTS ou leur version militaire qui offre une plus faible probabilité de défaillance et une plus grande plage de température de fonctionnement [3] [61 – 64].

Ce succès est justifié par le fait qu'en orbite LEO (altitude < 1000 km), un simple blindage en aluminium de quelques millimètres pour le segment spatial, permet d'avoir :

- Des niveaux de radiation relativement faibles avec une dose TID maximale de 11krad pour une altitude de 800 km (voir *tableau 1*).
- Des taux des erreurs SEE ne dépassant pas $8.492 \cdot 10^{-5} / bit.s$ pour une altitude de 800 km (voir *tableau 1*).

3.7. Concept des opérations de MASAT1

Le ConOps (Concept des Opérations) permet de percevoir la dynamique de fonctionnement des sous-systèmes constituant le segment spatial. Pour une mission scientifique comme MASAT1, il décrit les phases et les scénarios opérationnels en incluant les différents modes opérationnels et les interactions en eux.

Les opérations de la mission MASAT1 sont organisées en trois phases :

- **Phase Pré-lancement** : Cette phase est planifiée une fois le segment spatial est sur le site de lancement. Elle prévoit les opérations de préparation du segment spatial pour le lancement comme la charge des batteries ou la mise sous tension et la vérification des sous-systèmes.
- **Phase initiale de mise en orbite LEOP** (Launch and Early Orbit Phase) : Cette phase sera initialisée par la séparation du Cubesat de son P-POD. Elle prévoit les opérations qui prépareront la mise en service du segment spatial et la confirmation des performances opérationnelles des différents sous-systèmes de la plateforme.
- **Phase opérationnelle** : C'est la phase de mise en service du segment spatial avec l'exécution des différents scénarios opérationnels prévus pour mener à bien la mission scientifique de MASAT1.

La **figure 2-15** résume les différents phases et modes opérationnels prévus pour la mission MASAT1 ainsi qu'un aperçu préliminaire des transitions entre eux :

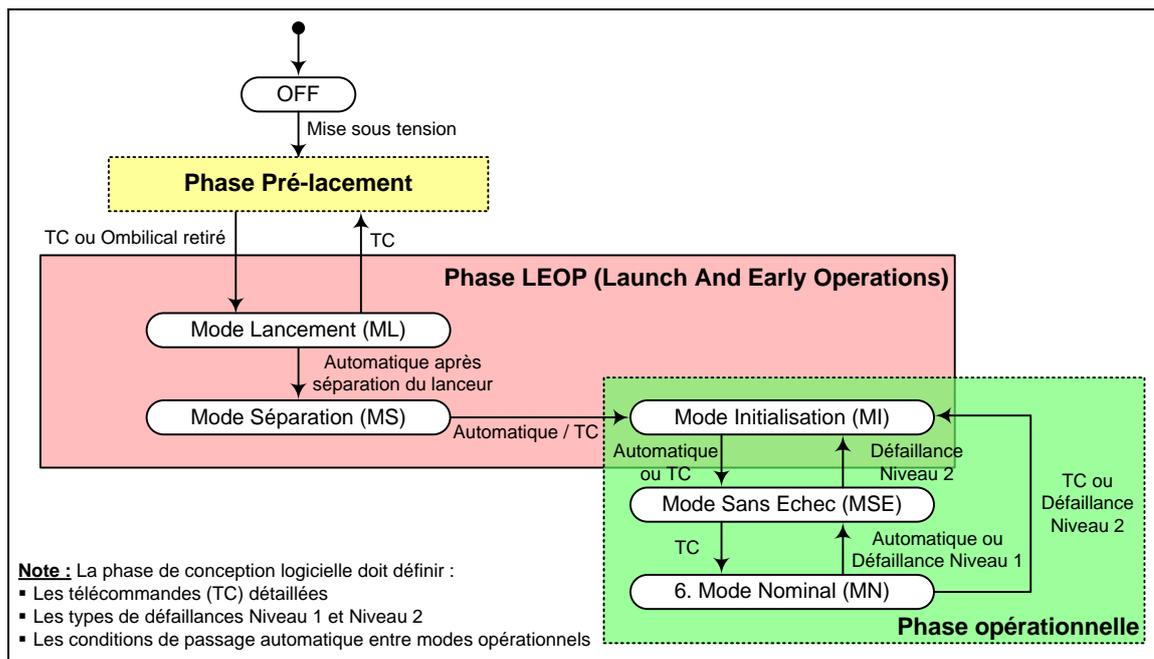


Figure 2-15 : Diagramme d'états et de transitions entre mode opérationnelles

Les modes opérationnels permettront de définir à tout moment les ressources actives du segment spatial ainsi que les sous-systèmes alimentés par l'énergie de bord. Les transitions entre les modes peuvent être effectuées via des télécommandes provenant de la station au sol, mais peuvent également, dans certaines situations, être déclenchées automatiquement par le logiciel de vol embarqué.

Dans la mission MASAT1, on distingue cinq principaux modes fonctionnels :

- **Mode Lancement (ML)** : Ce mode est activé avec l'intégration du segment spatial dans le P-POD et le retrait de la broche RBF (Remove Before Flight). Le commutateur de déploiement doit rester actionné pendant toute cette phase pour permettre de déconnecter le circuit d'alimentation du segment spatial qui reste donc éteint.
- **Mode Séparation (MS)** : Ce mode commence juste après l'activation du commutateur de déploiement qui indique la séparation du segment spatial de son P-POD. Les batteries sont alors connectées au bus d'alimentation et il y a un temps mort de 30 min [10] au bout duquel les opérations suivantes sont initiées :
 - Activation du sous-système EdB et déploiement des antennes.
 - Si le niveau de tension de la batterie est suffisant, activation de la balise après 15 min du déploiement des antennes [10] pour émettre périodiquement un signal prédéfini de survie. Ce signal généré par la COM permettra d'identifier le segment spatial et de valider la communication avec la station au sol.
 - Passage automatique en mode opérationnel "Initialisation".
- **Mode Initialisation (MI)** : Lorsque la tension de la batterie est suffisante, la séquence d'initialisation du segment spatial commence par :
 - Le démarrage/redémarrage de l'OdB qui entame, via des télécommandes, la vérification séquentielle du bon fonctionnement des autres sous-systèmes.
 - A la fin de la séquence de vérification, un signal balise contenant des informations sur l'état et les performances opérationnelles du segment spatial est envoyé à la station au sol. Ces données seront utiles pour corriger/calibrer les modèles de la station au sol et permettre de prendre les décisions.
 - Le système passe ensuite en mode Sans Echec automatiquement ou via des TC provenant de la station au sol.

Ce mode opérationnel est lancé à chaque défaillance majeure du segment spatial.

- **Mode Sans Echec (MSE)** : Dans ce mode, seuls les modules EdB et COM (en mode balise) sont opérationnels. En effet, si on sort d'un mode d'initialisation ou si la puissance fournie par l'EdB descend en dessous d'un certain seuil, ce dernier met hors tension tous les sous-systèmes (sauf la COM) et se tient prêt à passer en mode nominal lorsque la puissance fournie par les batteries est suffisante ou via des TC provenant de la station au sol.

- **Mode Nominal (MN)** : C'est un mode où le segment spatial est pleinement opérationnel. Il est ainsi possible de communiquer avec les stations au sol (réception de TC et envoi des données de télémétrie via le module COM), et de réaliser les tâches prévues pour le sous-système Charge Utile. On distingue dans ce mode, quatre cas de fonctionnement spécifiques qui dépendent surtout de la puissance fournie par le module EdB (voir *tableau 2-4*) :

Modes	Description
MN_TCU	Transmission et charge utile (FPGA) opérationnelles
MN_TnCU	Transmission opérationnelle et charge utile non opérationnelle
MN_nTCU	Transmission non opérationnelle et charge utile opérationnelle
MN_nTnCU	Transmission et charge utile (FPGA) non opérationnelles

Tableau 2-4 : Etats possibles en mode opérationnel nominal

Il est à noter que le concept des opérations a permis de dégager un certain nombre de recommandations pour les étapes de conception suivantes :

- La possibilité de travailler avec des stations au sol partenaires permettra de diminuer la durée de la phase LEOP.
- Pour chaque mode opérationnel, des scénarios de défaillances possibles et des plans de récupération doivent être élaborés au sein d'une stratégie globale de Détection, d'Isolation et de Récupération des Défaillances (DIRD). Cette stratégie doit être conçue de telle façon que toutes les défaillances prévues à bord puissent être surmontées, soit par l'action autonome du segment spatial ou par notification claire, non ambiguë et en temps opportun du problème à la station au sol.
- La nécessité pour le segment spatial d'effectuer des opérations autonomes pour au moins 48 heures (à valider) après séparation du P-POD.
- Emission d'un signal de balise facilement recevable par la communauté des radios amateurs.

4. DESCRIPTION DE LA PLATEFORME MASAT1

4.2. Diagramme bloc matériel et électrique

Le diagramme bloc de la *figure 2-16* décrit l'architecture matérielle du segment spatial (plateforme et charge utile) sachant qu'elle sera détaillée dans la partie étude et description de chacun des sous-systèmes. On notera les choix suivants :

- Les données de la chaîne d'acquisition située au sein du sous-système EdB sont accessibles directement via le bus I2C pour permettre aux modules OdB principal et secondaire d'acquérir indépendamment les données de télémétrie.
- Pour la distribution de l'énergie électrique dans l'ensemble du segment spatial, nous avons choisi de mettre les circuits de limitation de courant et de protection contre les événements latchup, au niveau du module EdB. Ce choix a l'avantage de permettre à l'EdB et l'OdB de protéger, d'activer/désactiver individuellement chacun des sous-systèmes selon les scénarios opérationnels. L'étude détaillée du module EdB permettra de valider ce choix.
- Le sous-système de contrôle thermique THER n'est pas représenté dans le diagramme bloc car jusqu'à la date de rédaction de ce rapport de thèse, les résultats de la modélisation thermique du MASAT1 n'ont pas encore abouti.

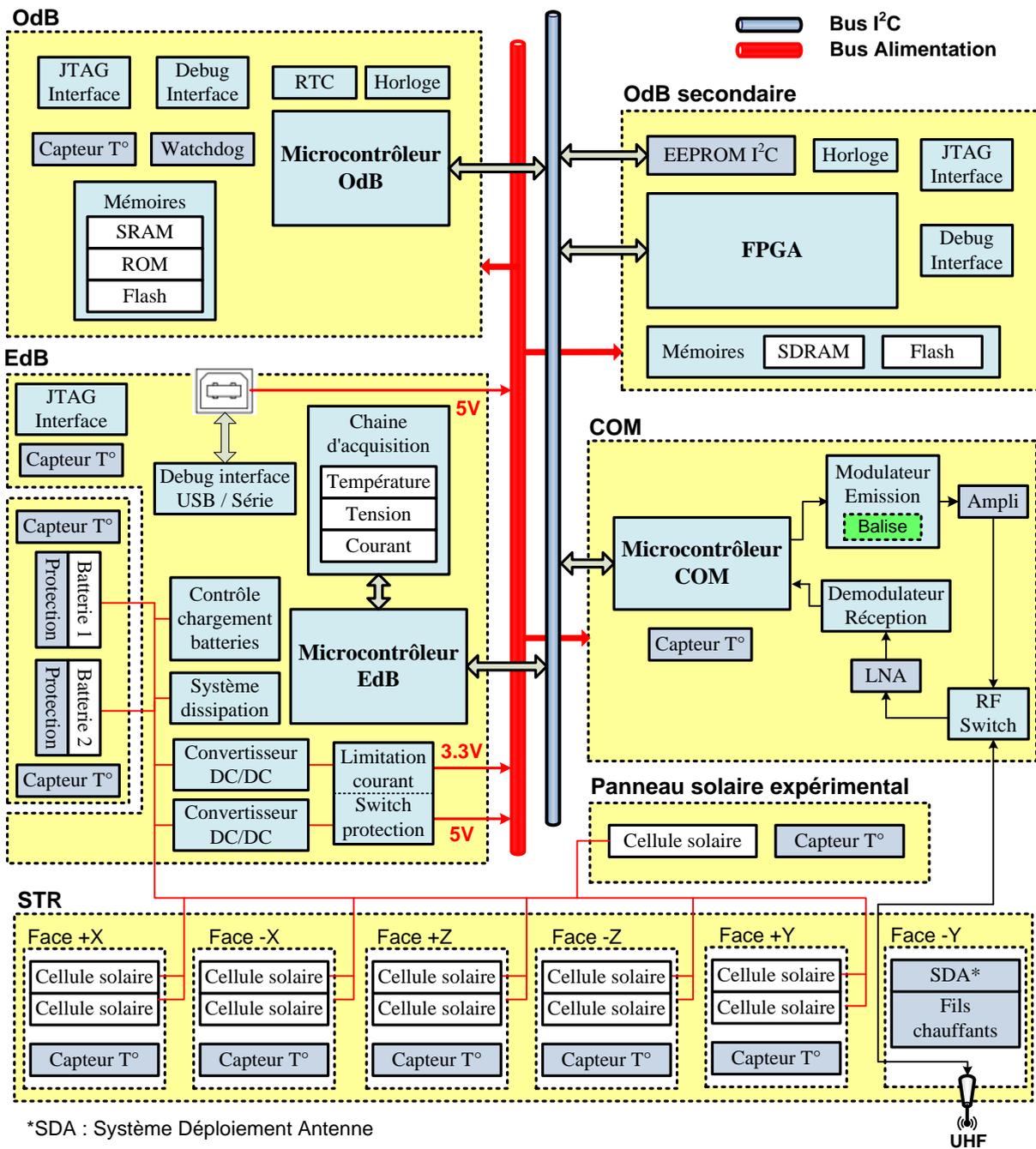


Figure 2-16 : Diagramme bloc matériel et électrique

4.3. Diagrammes Bloc des flux de données

Le ConOps a permis de réaliser le diagramme des flux de données de la **figure 2-17** qui représente l'ensemble des connexions permettant le transfert d'informations entre les différents sous-systèmes :

- Le support de transmission des données sera principalement la liaison série I2C sauf pour certains signaux qui vont transiter directement via le bus PC/104, comme les signaux provenant de l'EdB pour la mise sous ou hors tension des sous-systèmes.

- Les données de TM peuvent être acquises par l'OdB et l'OdB secondaire à partir de la chaîne d'acquisition de l'EdB via la liaison I2C.

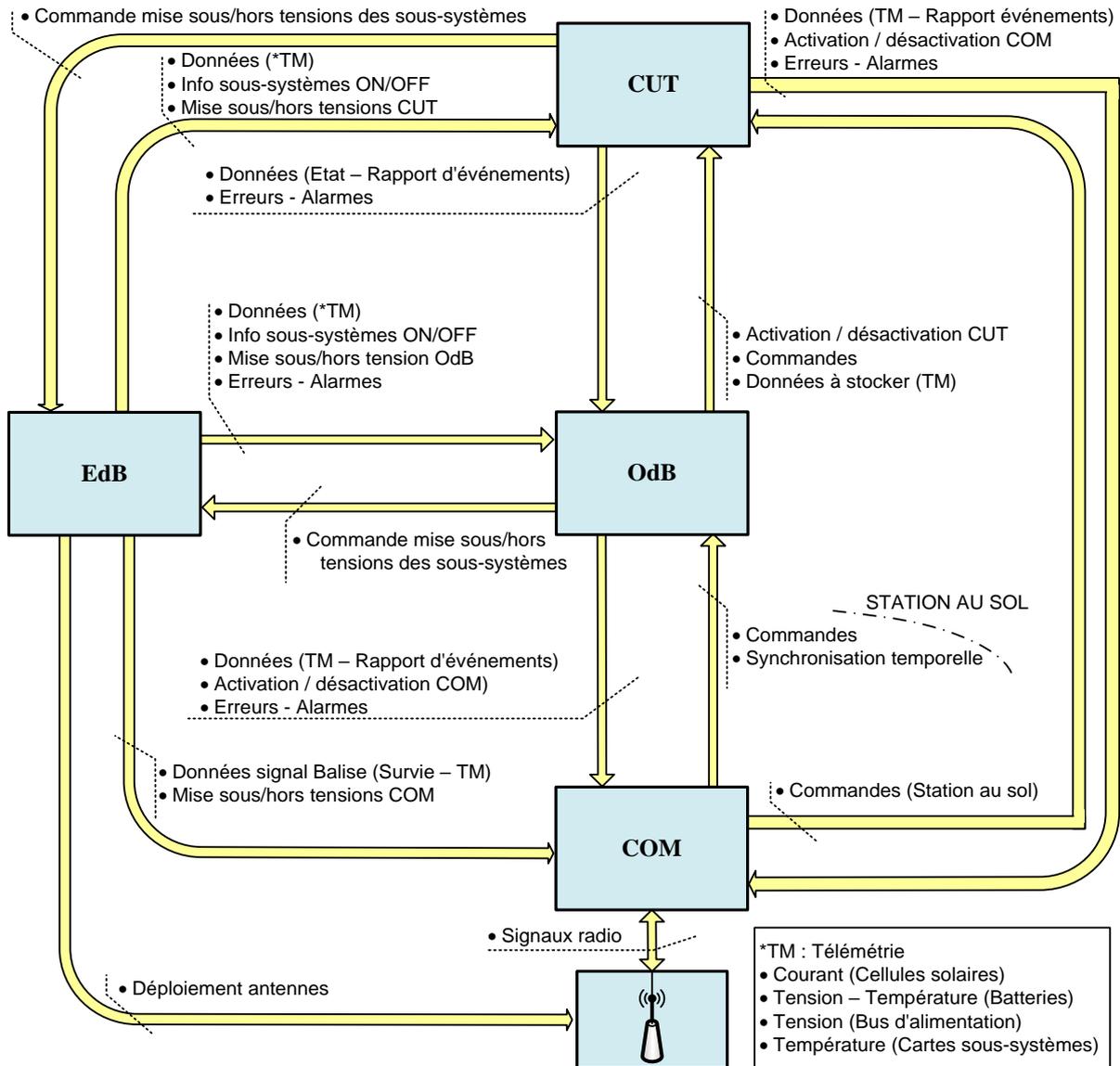


Figure 2-17 : Diagramme bloc représentant le flux des données

Chapitre 3

ETUDE ET DESCRIPTION DU SOUS- SYSTEME ENERGIE DE BORD

1. ETAT DE L'ART

1.1. Fonctionnalités d'un sous-système EdB

L'objectif principal d'un sous-système EdB consiste à produire suffisamment d'énergie pour déployer les antennes du segment spatial et alimenter ses différents sous-systèmes. Pour répondre à cette finalité, il est généralement constitué de trois principaux blocs :

- **Bloc de production de l'énergie** via des cellules solaires montées sur la structure du segment spatial.
- **Bloc de conditionnement et de distribution de l'énergie** vers les différents sous-systèmes, via des convertisseurs d'énergie DC-DC et des régulateurs de tension.
- **Bloc de stockage de l'énergie** via des batteries qui permettront d'alimenter le segment spatial pendant la phase LEOP et les périodes d'éclipse.

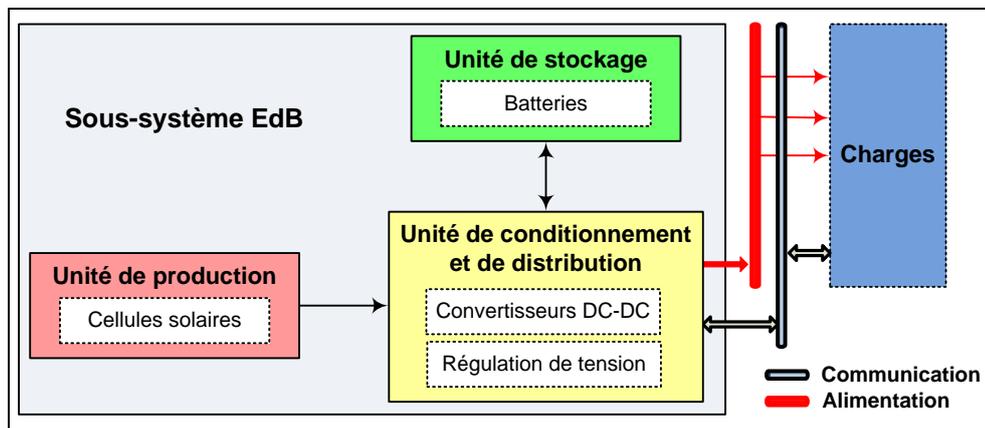


Figure 3-1 : Diagramme bloc du sous-système EdB

En fonction des exigences de la mission spatiale, le sous-système EdB doit être en mesure d'activer ou de désactiver chacun des sous-systèmes du segment spatial, en plus de communiquer ces données de télémétrie via le bus de communication de la plateforme.

1.2. Les cellules photovoltaïques

Une cellule photovoltaïque (PV) est faite de matériaux semi-conducteurs dopé type P-N permettant de générer de l'énergie en transformant l'énergie incidente des photons provenant du rayonnement solaire. Or la physique quantique montre que pour les applications spatiales, de telles cellules ont un rendement η_{PV} , au mieux de l'ordre de 30% [146] ce qui implique une limitation de la puissance pouvant être obtenue.

L'association de plusieurs cellules PV en série/parallèle donne lieu à un générateur PV (GPV) dont les caractéristiques I-V (courant-tension) et P-V (puissance-tension) sont non linéaires. La *figure 3-2* représente les courbes I-V et P-V d'une cellule PV à base du semi-conducteur Arséniure de Gallium (GaAs) :

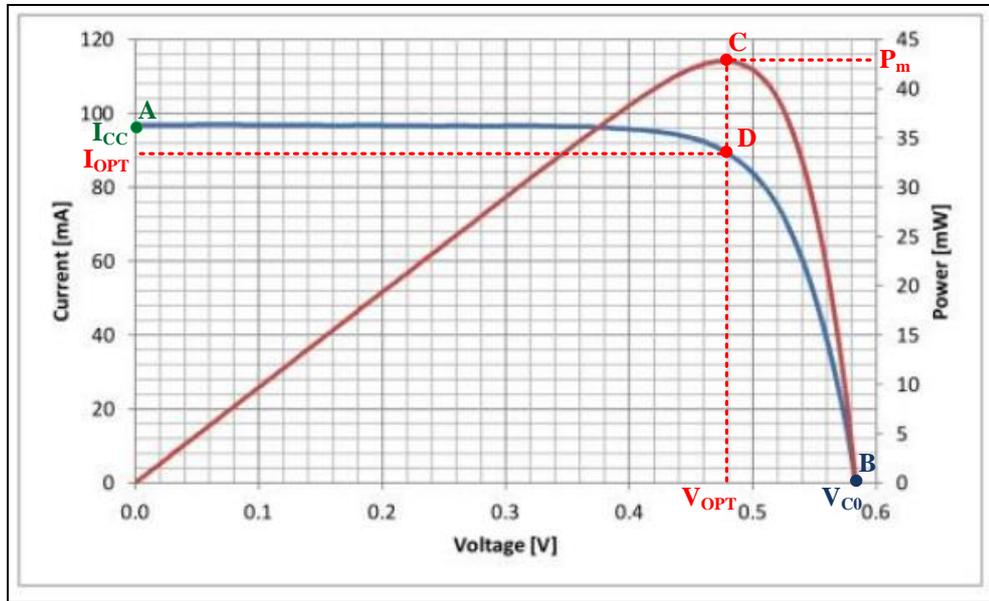


Figure 3-2 : Courbes I-V et P-V d'une cellule solaire type GaAs [65]

Les valeurs importantes qui peuvent être déduites des courbes I-V et P-V sont :

- Le courant maximal (point A : I_{cc}) obtenu lorsque la cellule PV est court-circuitée.
- La tension maximale (point B : V_{c0}) atteinte lorsque la cellule PV fonctionne en circuit ouvert.
- Le point maximum de puissance MPP (points C et D : P_m , I_{OPT} et V_{OPT}) correspondant au point de fonctionnement optimale de la cellule PV.

Ces valeurs dépendent de plusieurs paramètres environnementaux dont les plus important sont :

- La température qui est inversement proportionnelle à la tension V_{c0} en circuit ouvert et qui, en augmentant, entraîne la diminution de l'efficacité des cellules solaires et celle du courant I_{cc} de court-circuit [66].
- L'intensité et la longueur d'onde de la lumière à laquelle est exposée la cellule solaire. En effet, les courbes I-V sont généralement données pour des conditions d'éclairement et de température normalisées à savoir le spectre solaire après la traversée de l'atmosphère AM1.5 (Air Mass 1.5 : $1000 \text{ W/m}^2 - 25^\circ\text{C}$), sachant que AM0 correspond au spectre solaire hors atmosphère (1300 W/m^2) [147].

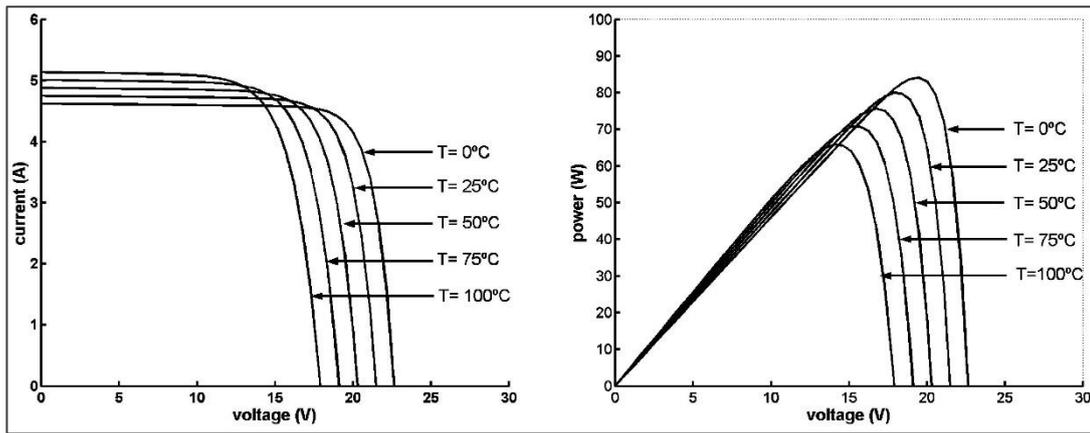


Figure 3-3 : Effet de la température sur les courbes I-V et P-V [66]

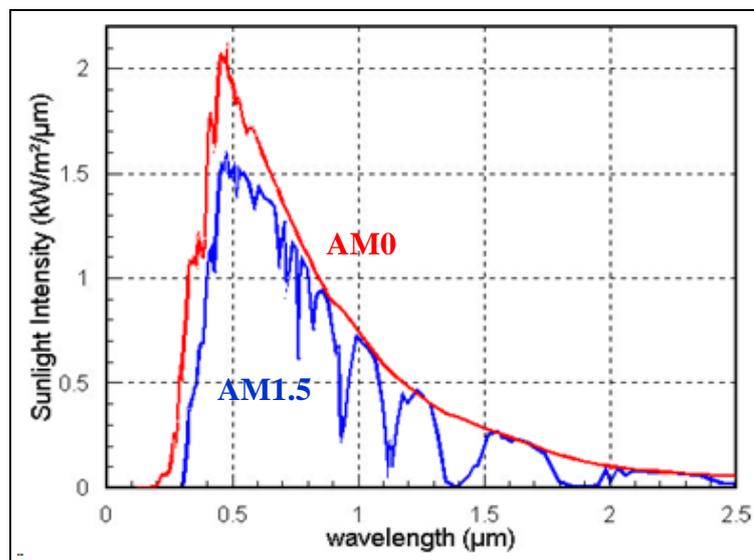


Figure 3-4 : Spectre solaire [147]

- Les radiations qui sont directement proportionnelles au courant I_{cc} de court-circuit, et qui ont tendance à diminuer l'efficacité des cellules solaires au fil du temps. Les termes couramment utilisés pour définir l'âge d'une cellule solaire sont BOL (Begin of Life) et EOL (End of Life).

La caractéristique I-V du GPV dépend donc du changement de ces paramètres dans l'environnement spatial, et son point de fonctionnement dépend directement de la charge qu'il alimente. Afin d'extraire en chaque instant le maximum de puissance disponible aux bornes du GPV en fonction de la charge alimentée, on implémente une fonction de recherche du point de puissance maximale MPPT (Maximum Power Point Tracking). Cette technique réalise, en effet, une adaptation d'impédance entre la charge et le GPV afin d'assurer un point de fonctionnement tout proche du point de puissance maximale (PPM), qui correspond à une tension et un courant optimaux.

1.3. Fonction MPPT de recherche du point de puissance maximale

1.3.1. Principe

Pour que la connexion GPV-charge soit possible, un point de fonctionnement correspondant à l'intersection de leurs caractéristiques I-V doit exister. Comme nous pouvons le constater dans la *figure 3-5*, dans le cas d'une connexion directe, ce point de fonctionnement dépend fortement de la charge qu'elle soit une simple résistance R (P1 et P3) ou bien une batterie (P2) :

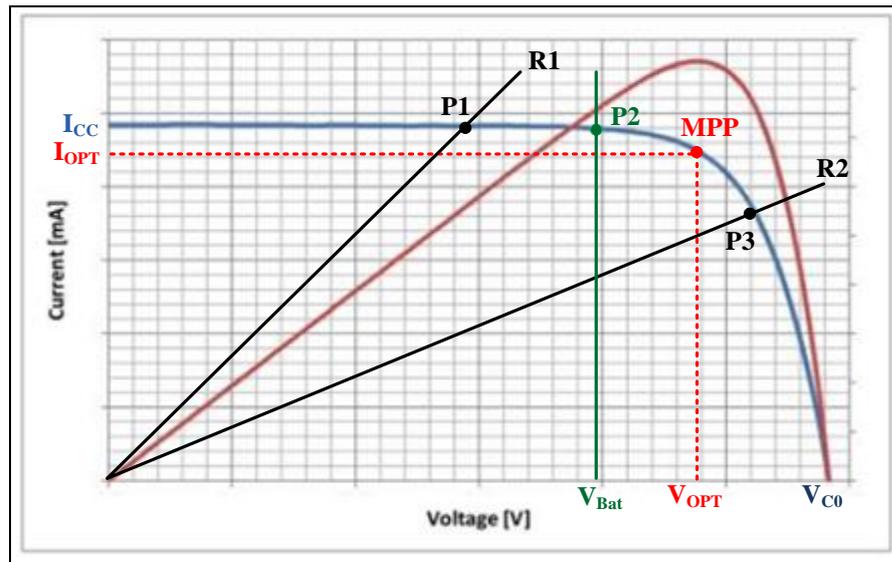


Figure 3-5 : Points de fonctionnement résultants de l'association directe d'un GPV avec une charge résistive (R1, R2) ou une batterie (V_{Bat})

Le rendement d'un point de fonctionnement, appelé rendement MPPT η_{MPPT} , permet d'évaluer l'efficacité d'une connexion GPV-charge :

$$\eta_{MPPT} = \frac{P}{P_{max}} = \frac{\text{Puissance délivrée à la charge}}{\text{Puissance max disponible à la sortie du GPV}} \quad (3.1)$$

Dans le cas d'une connexion directe avec une batterie, ce rendement dépend de l'écart entre la tension optimale V_{OPT} du point MPP et la tension de la batterie qui varie en fonction de son état de charge :

$$\eta_{MPPT} = \frac{V_{Bat} \cdot I_{cc}}{V_{OPT} \cdot I_{OPT}} \quad (3.2)$$

1.3.2. Etage d'adaptation entre GFV et charge

Afin d'extraire à chaque instant le maximum de puissance disponible aux bornes du GPV et la transférer à la charge, on insère un étage d'adaptation entre les deux. La solution communément utilisée est un convertisseur d'énergie statique DC-DC.

Ce dernier permet au GPV, par le biais d'une commande analogique ou numérique, de délivrer sa $P_{max} = V_{OPT} \cdot I_{OPT}$ tout en assurant la tension ou le courant caractéristique de la charge (voir *figure 3-6*) [67-69] :

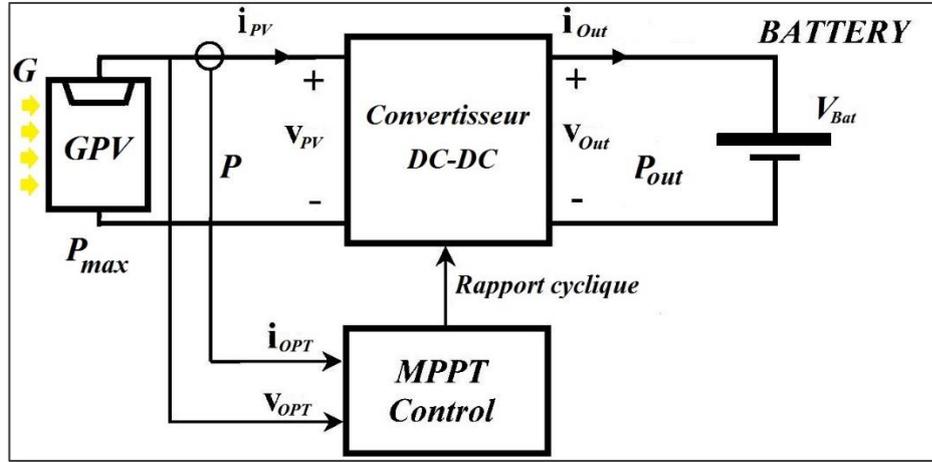


Figure 3-6 : Solution MPPT pour un système PV [66]

Les calculs d'un convertisseurs DC-DC démontrent que [66] :

$$V_{Out} = \frac{1}{1 - \alpha} \cdot V_{PV} \quad \text{et} \quad I_{Out} = (1 - \alpha) \cdot I_{PV} \quad \text{avec} \quad \alpha : \text{rapport cyclique} \quad (3.3)$$

Pour une adaptation optimale :

$$I_{PV} = I_{OPT} \quad \text{et} \quad V_{PV} = V_{OPT} \Rightarrow R_{OPT} = \frac{V_{OPT}}{I_{OPT}} = (1 - \alpha)^2 \cdot \frac{V_{Out}}{I_{Out}} = (1 - \alpha)^2 \cdot R_s \quad (3.4)$$

La connexion entre le panneau PV et la charge R_s peut être optimisée en ajustant le rapport cyclique α pour que, d'un côté, le GPV puisse fonctionner à R_{OPT} et que de l'autre côté, la charge puisse varier à sa guise dans la mesure où le point d'intersection source-charge continue d'exister. Il est à noter qu'une seule grandeur aux bornes de la charge peut être régulée.

1.3.3. Rendement de la chaîne de puissance

Pour évaluer l'efficacité d'une solution MPPT, nous devons déterminer le rendement de chaque partie de la chaîne de puissance :

- Rendement maximum η_{PV} de la conversion photons-électrons du panneau PV :

$$\eta_{PV} = \frac{P_{max}}{G \cdot A_{eff}} \quad (3.5)$$

Où P_{max} : puissance max disponible à la sortie du panneau PV.

G (W/m^2) : Irradiance ou la puissance lumineuse reçue par unité de surface.

A (m^2) : Surface effective du panneau PV.

- Rendement MPPT η_{MPPT} qui dépend du type de la commande MPPT utilisée :

$$\eta_{MPPT} = \frac{P}{P_{max}} \quad (3.6)$$

Où P : puissance effectivement délivrée par le GPV.

- Rendement du convertisseur η_{CONV} généralement fourni par le constructeur :

$$\eta_{CONV} = \frac{P_{out}}{P} \quad (3.7)$$

Où P_{out} : puissance délivrée par le GPV à la charge.

Le rendement total de la chaîne de puissance sera défini par :

$$\eta_{total} = \eta_{PV} \cdot \eta_{MPPT} \cdot \eta_{CONV} \quad (3.8)$$

2. APPROCHES D'IMPLEMENTATION ET ARCHITECTURE

Pour les petits satellites, les trois approches de mise en œuvre du sous-système EdB communément utilisées sont les suivantes [70] :

- 1) **Transfert direct d'énergie avec le bus d'alimentation connecté à la batterie** : c'est la topologie la plus simple avec une masse et un volume réduits. Mais le couplage direct de la batterie avec le bus d'alimentation de la plateforme implique la variation de la tension en fonction de l'état de charge de la batterie, et donc la nécessité d'utiliser des panneaux PV plus larges et donc plus coûteux et plus lourds. Cette solution n'est pas adaptée à la plupart des scénarios de mission et surtout les missions avec des contraintes de masse et de volume.
- 2) **Transfert direct d'énergie avec un bus d'alimentation régulé** : c'est une topologie où l'énergie des panneaux solaires alimente directement le bus via des diodes de protection (voir *figure 3-7*). Pendant les périodes d'ensoleillement, la tension du bus est régulée par un régulateur de dérivation en commutation séquentielle S3R (Sequential Switching Shunt Regulator). Pour les périodes de forte demande de puissance, la batterie commence à alimenter le bus via le régulateur de décharge de la batterie BDR (Battery Discharge Regulator). Mais pendant les périodes d'éclipse, il y a la nécessité de décharger la batterie via le BDR pour conserver la tension du bus constante, ce qui diminue l'efficacité du système. Cette solution est donc particulièrement inadaptée pour les missions avec de fréquentes périodes d'éclipse comme les Cubesats qui évoluent à une orbite LEO.

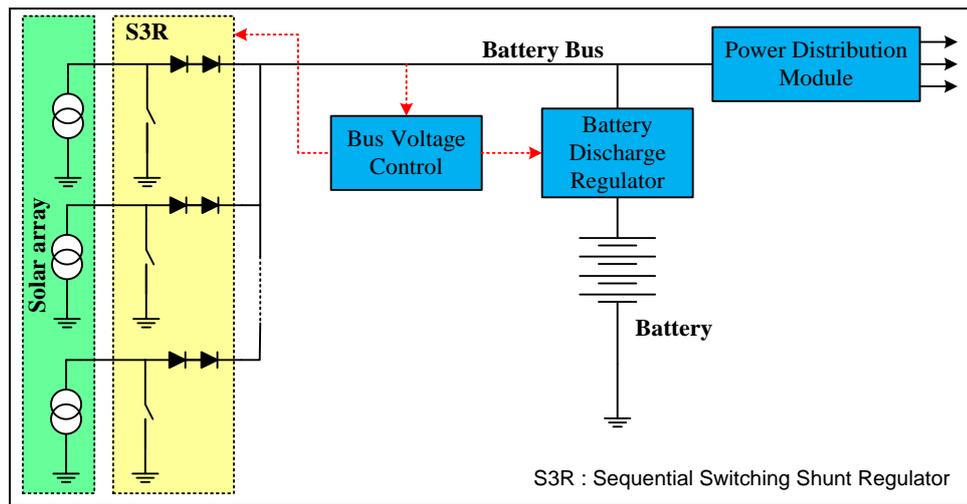


Figure 3-7 : Transfert direct d'énergie avec un bus d'alimentation régulé [70]

- 3) **MPPT avec le bus d'alimentation connecté à la batterie** : c'est une architecture qui intègre un système de recherche du point de puissance maximale MPPT entre les panneaux PV et la batterie. Comme on l'a décrit précédemment, ce système fonctionne selon le principe de la charge de la batterie et l'alimentation du bus pendant les périodes d'ensoleillement, tout en réglant la tension au point PPM. Pendant l'éclipse, l'énergie est directement transférée de la batterie vers le bus d'alimentation. Cette solution est particulièrement adaptée pour les missions où le point PPM change considérablement pendant la période d'ensoleillement du segment spatial, ainsi que pour les missions qui ont des contraintes de surfaces au niveau des panneaux solaires.

Pour MASAT1, le meilleur choix pour le sous-système EdB s'avère être l'architecture MPPT avec le bus d'alimentation connecté à la batterie. Ceci pour les raisons suivantes :

- Les contraintes de volume et de masse d'un Cubesat 1U qui limitent la surface des panneaux photovoltaïques à 362,16 cm².
- L'utilisation d'un contrôle d'attitude passif entraîne des changements rapides au niveau de l'irradiation des panneaux photovoltaïques et par conséquent une variation du point MPP qui impose l'utilisation d'un système MPPT.
- Le Cubesat évolue en orbite terrestre basse LEO ce qui implique de fréquentes périodes d'éclipse.

La **figure 3-8** montre l'architecture adoptée pour le sous-système Energie de Bord de la mission MASAT1 :

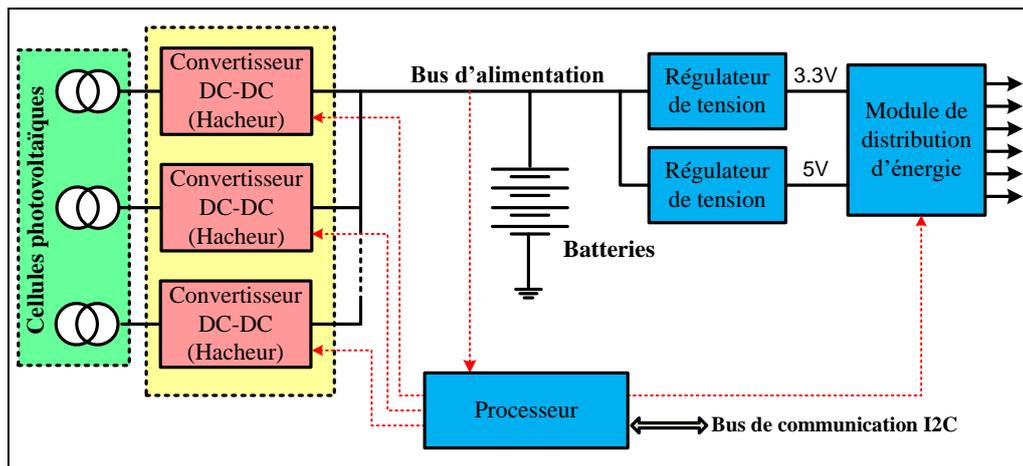


Figure 3-8 : Architecture adoptée pour le sous-système EdB de MASAT1

3. DESCRIPTION DU SOUS-SYSTEME EDB DE LA MISSION MASAT1

3.1. Exigences du sous-système EdB

La tâche principale du module EdB est de fournir l'énergie nécessaire au bon fonctionnement du segment spatial tout en permettant de satisfaire aux exigences préliminaires spécifiées dans le DES [71] :

[DES] FONC-0400

L'EdB doit fournir la puissance électrique nécessaire pour satisfaire toutes les exigences de charge pendant toutes les phases de la mission et pour tous les modes de fonctionnement.

[DES] CONF-0410

L'alimentation électrique doit être conçue à base d'un générateur photovoltaïque (GPV) et sa configuration électrique doit être définie sur la base de la topologie sélectionnée pour l'EdB.

[DES] CONF-0420

Le sous-système EdB doit être capable de stocker de l'énergie.

[DES] OPER-0430

L'EdB doit être capable de fonctionner en continu avec des charges changeantes comme l'exige les modes opérationnels de la mission.

[DES] OPER-0460

Le fonctionnement de l'EdB doit être entièrement automatique, y compris la gestion du stockage d'énergie.

[DES] OPER-0470

L'EdB doit accepter la modification des paramètres de fonctionnement via les commandes provenant du segment au sol.

[DES] OPER-0480

L'EdB doit fournir des informations de TM suffisantes pour assurer la supervision et la commande éventuelle au cours des essais au sol et de l'exploitation en orbite spatiale.

[DES] OPER-0490

En orbite, l'EdB doit au minimum rapporter les données TM suivantes :

- L'état de chaque circuit de puissance commutable au point de distribution d'énergie.
- Le courant pour chaque circuit de puissance au point de distribution d'énergie.
- La tension pour chaque circuit de puissance au point de distribution d'énergie.

[DES] CONC-0500

Les facteurs de dégradations dues à l'environnement spatial, les variations de l'éclairement solaire y compris les effets thermiques et les incertitudes de conception qui en découlent, doivent être pris en compte pour répondre aux changements d'efficacité du processus de conversion d'énergie.

[DES] CONC-0510

Les performances des cellulaires photovoltaïques et les facteurs de dégradation doivent être justifiés en fonction des expériences orbitales et soutenues par les essais au sol.

[DES] CONC-0530

La topologie de l'EdB doit être optimisée pour le transfert de puissance entre les sources et les charges et pour créer un environnement CEM approprié.

[DES] CONC-0540

La distribution de puissance doit être faite en conformité avec les besoins des différentes charges, à la fois statiques et dynamiques, ainsi que les caractéristiques puissance/énergie des sources. Un concept de protection doit être mis en place pour éviter la propagation des défaillances entre les sous-systèmes.

[DES] CONC-0545

L'EdB doit prévoir un moyen pour recharger ces batteries au sol, avant le lancement.

[DES] FMST-0560

L'EdB doit fournir une protection contre les court-circuit ou une limitation du courant pour chaque circuit d'alimentation commutable (découplage de la charge de chaque bus d'énergie).

[DES] FMST-0570

L'EdB doit accepter d'être alimentée à partir d'équipements de tests en utilisant des batteries installées à bord ou externes afin de faciliter les essais au sol ou sur le lanceur.

3.2. Interfaces électriques

Les interfaces électriques du sous-système EdB sont résumées par la **figure 3-9**. On distingue :

- Les entrées **Vsc_Pi** provenant des cellules photovoltaïques des panneaux solaires.
- Les entrées **KS** (Kill switch) et **RBF** (Remove Before Flight)
- Les sorties configurables pour alimenter les sous-systèmes du segment spatial.
- L'interface I2C de communication et celle JTAG (Joint Test Action Group) de configuration/débogage.

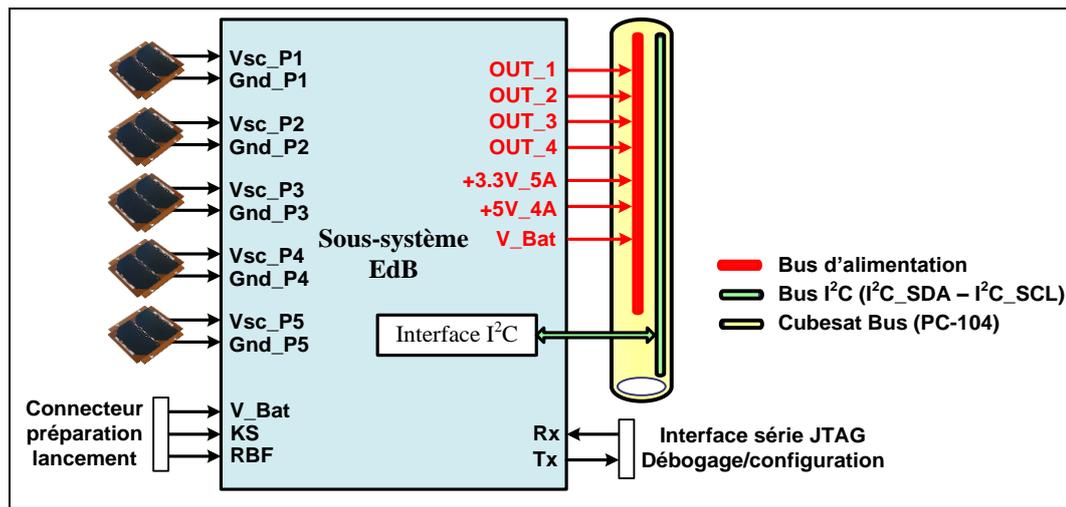


Figure 3-9 : Interface électrique du sous-système EdB

4. CHOIX DE LA SOLUTION TECHNOLOGIQUE

4.1. Cellules photovoltaïques

4.1.1. Choix de la technologie

Les technologies de cellules photovoltaïques disponibles comprennent les cellules simple jonction au Silicium (Si) ou à l'arséniure de gallium (GaAs), les cellules double jonction (GaInP2/GaAs) et les cellules triple jonction (GaInP2/GaAs/Ge). Cependant, la seule technologie viable pour un Cubesat 1U orbitant en orbite LEO est la triple jonction, et ceci pour les raisons suivantes :

- Une efficacité plus élevée par rapport à une cellule à base de Si de même taille (jusqu'à 30% contre 17%) [77, 86, 87].
- Une meilleure résistance au rayonnement [73] et de meilleurs coefficients de température [72], assurant une plus grande puissance de livraison à EOL.
- Une tension de sortie plus élevées de 2,35 V (contre 0,6 V pour les cellules Si), ce qui implique l'utilisation de moins de cellules en série pour atteindre les niveaux de tensions désirables.

Le choix s'est porté sur des cellules à triple jonction InGaP/GaAs/Ge type 3G30A d'AZURSPACE qui ont une efficacité BOL maximale de 30%. Elles sont fournies avec une diode Schottky de protection intégrée et leurs caractéristiques sont résumés dans le *tableau 3-1* :

1 cellule 30.18 cm ² – BOL – Spectre AM0 (1367 W/m ² - 28 °C)			
Caractéristique		Valeur	Unité
Tension moyenne en circuit ouvert	V _{CO}	2690	mV
Courant de court-circuit moyen	I _{CC}	519.6	mA
Tension au point MPP	V _{OPT}	2409	mV
Courant au point MPP	I _{OPT}	502.9	mA
Puissance maximale	P _{max}	1.211	W
Efficacité moyenne	η	29.3	%
Gradients de température	∂V _{CO} /∂T	-6.2	mV/°C
	∂I _{CC} /∂T	0.36	mA/°C
	∂V _{OPT} /∂T	-6.7	mV/°C
	∂I _{OPT} /∂T	0.24	mA/°C

Tableau 3-1 : Caractéristiques électriques des cellules PV type 3G30A [146]

4.1.2. Modèle électrique simplifié

Une cellule photovoltaïque idéale peut être modélisée par le schéma équivalent simplifié de la **figure 3-10** où le courant produit est modélisé par un générateur de courant et la jonction p-n par une diode, avec R_{SH} et R_S des résistances en série et en parallèle équivalentes représentant les pertes du système [74] :

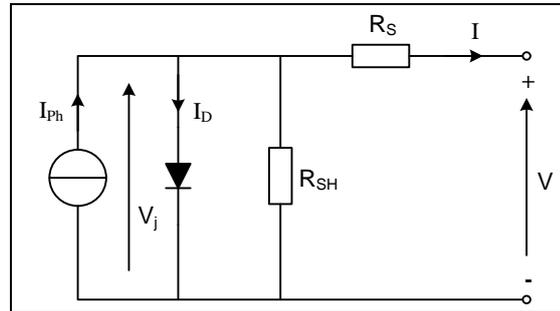


Figure 3-10 : Circuit équivalent simplifié d'une cellule PV [74]

En négligeant la résistance de shunt R_{SH}, les équations donnant la relation de la caractéristique I-V deviennent :

$$V = V_j - R_S \cdot I \quad \text{et} \quad I = I_{ph} - I_D \quad (3.9)$$

$$I_D = I_0 \cdot \left(\exp \left[\frac{q \cdot V_j}{n \cdot k \cdot T} \right] - 1 \right) \quad \text{avec} \quad I_0 = K_1 \cdot T^3 \cdot \exp \left[\frac{-E_g}{k \cdot T} \right] \quad (3.10)$$

- où : I₀ (A) : courant inverse de saturation.
I_{ph}(A) : courant photovoltaïque générée.
V_j (V) : tension aux bornes de la jonction P-N.
q (A.s) : charge électrique.
k (J/K) : constante de Boltzmann.
T (°K) : température.
n : facteur d'idéalité de la jonction P-N.
K₁ (A/cm²k³) : constante thermique du modèle de cellule.
E_g (J) : Energie de la bande interdite de la jonction P-N.

4.1.3. Panneaux photovoltaïques de MASAT1

Le segment spatial de MASAT1 sera équipé de panneaux photovoltaïques sur cinq de ces six faces ; la sixième face étant réservée aux antennes et à la charge utile. Ces panneaux PV sont configurés de telle sorte que les cellules de chaque panneau soient connectées en série et que les panneaux soient connectés en parallèle pour alimenter les convertisseurs DC-DC de la chaîne de puissance MPPT comme le montre la *figure 3-11* :

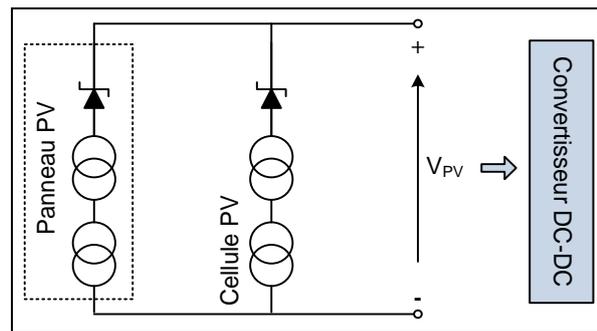


Figure 3-11 : Configuration des panneaux PV de MASAT1

Si l'un des panneaux branchés en parallèle n'est pas autant éclairé que l'autre, il ne produit pas suffisamment de puissance et risque d'être traversé par un courant inverse qui l'endommagera. C'est pour cela que chaque panneau PV est équipé d'une diode Schottky (tension directe basse) en série qui permettra, en plus, d'éviter qu'un panneau défectueux court circuité n'affecte le reste des panneaux PV.

Le choix s'est porté sur les panneau solaires NanoPower P110-A de GOMspace qui sont formés de deux cellules InGaP/GaAs/Ge 3G30A d'AZURSPACE, connectées en série et fournissant une surface effective de 63,36 cm² et une tension de sortie de 4.2 V au point MPP :

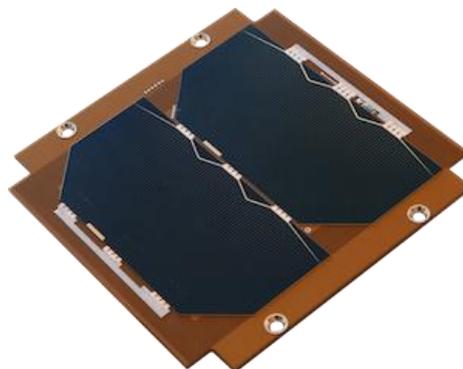


Figure 3-12 : Panneau solaire de GOMSPACE équipé de 2 cellules PV type 3G30A

4.1.4. Modélisation des panneaux photovoltaïques sous MATLAB

Pour modéliser un panneau PV sous Matlab, on a adapté le code proposé par l'équipe du Cubesat OUFTI-1 [75] en utilisant les paramètres constructeurs fournis par AZURSPACE pour les cellules PV type 3G30A, et un modèle pour la diode Schottky.

Le code Matlab utilisé est donné à l'ANNEXE D. Les *figures 3-13* et *3-14* montrent les caractéristiques I-V et P-V du panneau solaire pour des températures de -45°C , 5°C et 80°C , lorsque le niveau d'éclairement est nominal ($G_{\text{nom}} = 1350\text{W/m}^2$). Les *figures 3-15* et *3-16* montrent les caractéristiques I-V et P-V pour des niveaux d'éclairement de $0.5 \cdot G_{\text{nom}}$, $0.75 \cdot G_{\text{nom}}$ et G_{nom} , lorsque la température est de 5°C :

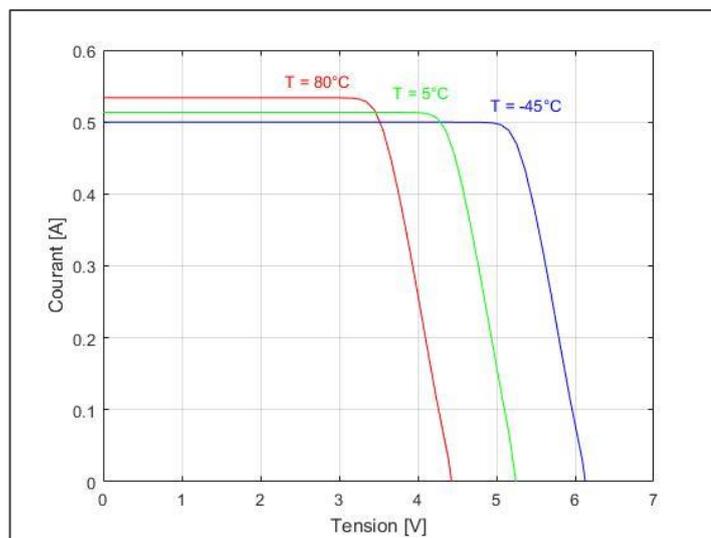


Figure 3-13 : Courbe I-V d'un panneau PV pour différentes températures [Matlab]

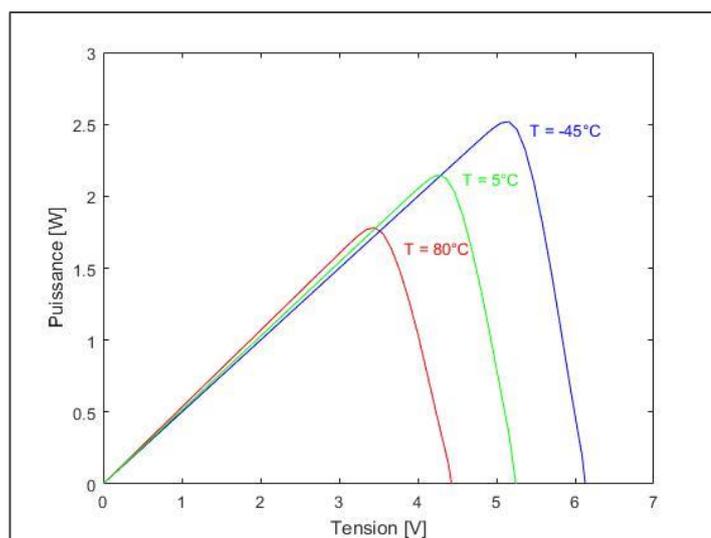


Figure 3-14 : Courbe P-V d'un panneau PV pour différentes températures [Matlab]

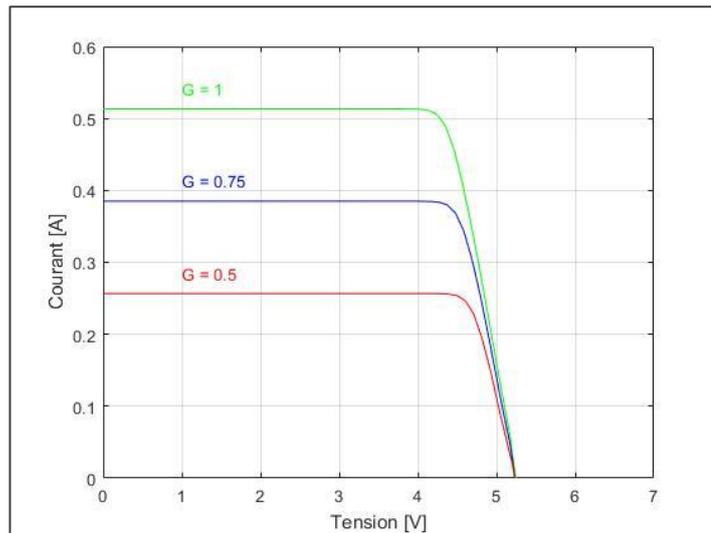


Figure 3-15 : Courbe I-V d'un panneau PV pour différents niveaux d'éclairement

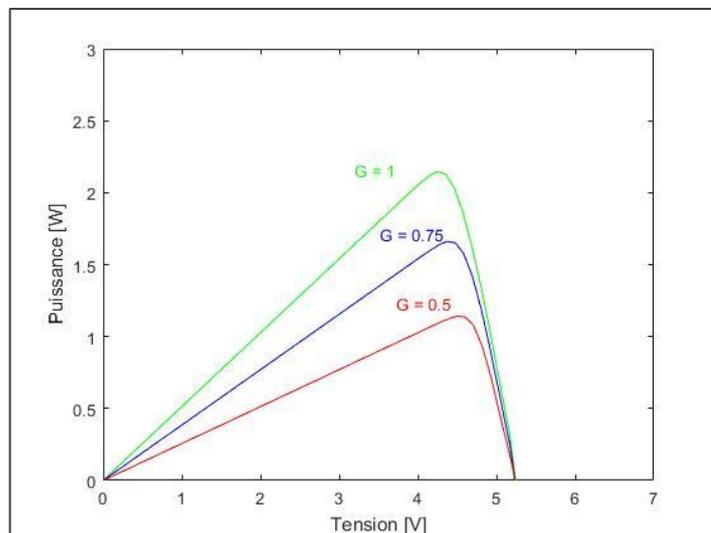


Figure 3-16 : Courbe P-V d'un panneau PV pour différents niveaux d'éclairement

4.2. Module Energie de Bord

L'architecture MPPT des modules EdB est généralement similaire pour les trois principaux fournisseurs spécialisés présents sur le marché : Clyde Space, GOMSpace et Crystalspace. La différence se situe au niveau des performances et l'une des plus importante de ces performances est le rendement des convertisseurs DC/DC (Régulateurs de tension) qui permettent de convertir la tension issue des batteries en une tension continue permettant d'alimenter les sous-systèmes du segment spatial.

La *figure 3-17* compare l'efficacité des principaux modules EdB présents sur le marché. On remarque une grande variation de cette efficacité dans la région des faibles puissances consommée qui est la région de consommation des Cubesats 1U [76] :

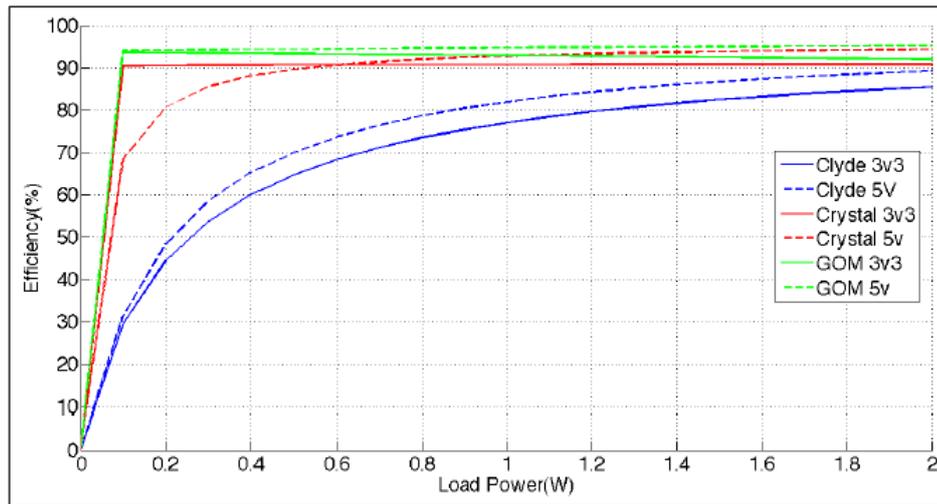


Figure 3-17 : Rendement des convertisseurs DC/DC des différents modules EdB [76]

La *figure 3-17* montre que la société Clyde-Space ont optimisé leurs convertisseurs pour des charges à fortes consommation d'énergie, tandis que GOMSpace et CrystalSpace ont de meilleures performances avec les charges à faible consommation d'énergie.

Les autres critères qui ont motivé notre choix sont résumés dans le *tableau 3-2*. Le poids donné à chaque critère est dicté par les contraintes et la nature de notre mission. Ainsi, le prix a constitué le principal critère de choix, suivi de la fiabilité et de l'héritage de vol du module. L'évaluation des points donnés pour chaque critère est faite en fonction des documents constructeurs et du bon sens de l'ingénierie des systèmes.

Critères	Poids	Clyde space	GOMSpace	Crystalspace
Prix	5	1	5	3
Héritage vol / fiabilité	4	5	4	2
Taille / poids	2	3	2	5
Capacités opérationnelles et de communication	3	3	3	1
Total	-	40	54	36

Tableau 3-2 : Comparaison des modules EdB COTS des différents fournisseurs [90 - 93]

Le choix s'est ainsi porté sur le module GomSpace NanoPower P31u [78] qui satisfait la plupart des exigences avec un fort potentiel de paramétrage et de reconfiguration logicielle de ces caractéristiques.



Figure 3-18: Module EdB GOMSpace Nanopower P31U [79]

Comme le montre la **figure 3-19**, l'architecture du module GOMSpace Nanopower P31U possède les fonctionnalités suivantes :

- 3 canaux d'entrée pour les panneaux PV avec une puissance d'entrée de 30 W.
- Implémentation de l'algorithme MPPT.
- Protection des batteries contre les sous-tensions et les surtensions.
- Deux bus d'alimentation régulés : 3,3 V @ 5 A et 5 V @ 4 A
- 6 commutateurs de sortie configurables et commandés avec limitation du courant.
- Temporisateurs "chien de garde".
- Mesures des données de télémétrie propres au module.
- 2 Batteries lithium ion embarquées avec un système de chauffage intégré.
- Interface de communication I2C et compatibilité avec la norme PC104.

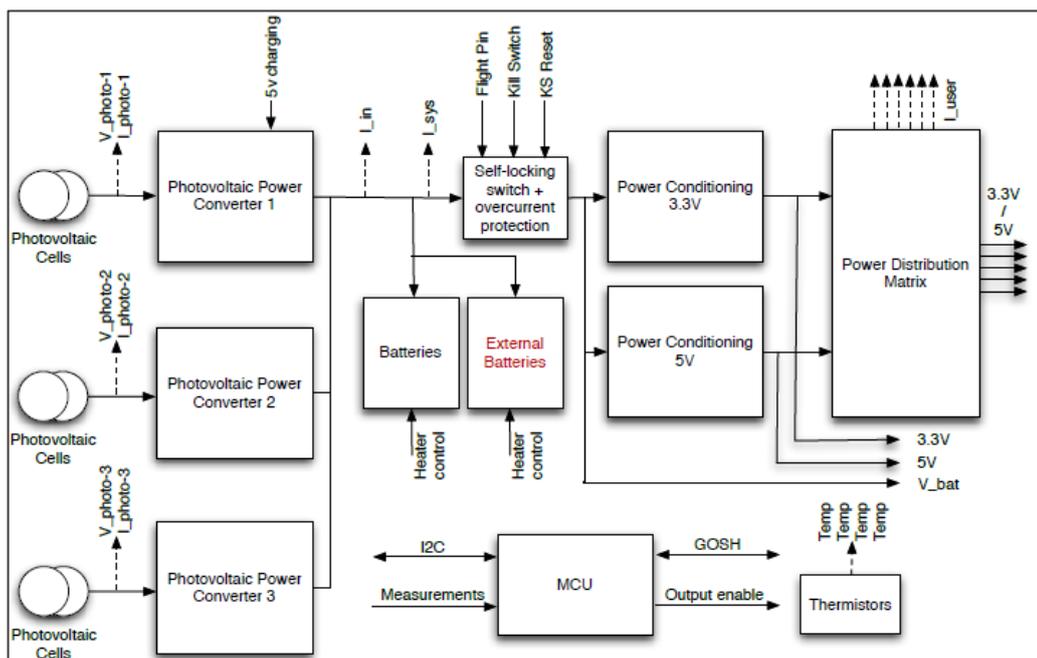


Figure 3-19 : Diagramme fonctionnel du module Nanopower P31U [79]

La *figure 3-20* et le *tableau 3-3* résument les connecteurs et la description des broches qui permettront de valider la compatibilité de cette solution technologique avec les choix matériels réalisés au niveau des sous-systèmes COM et OdB :

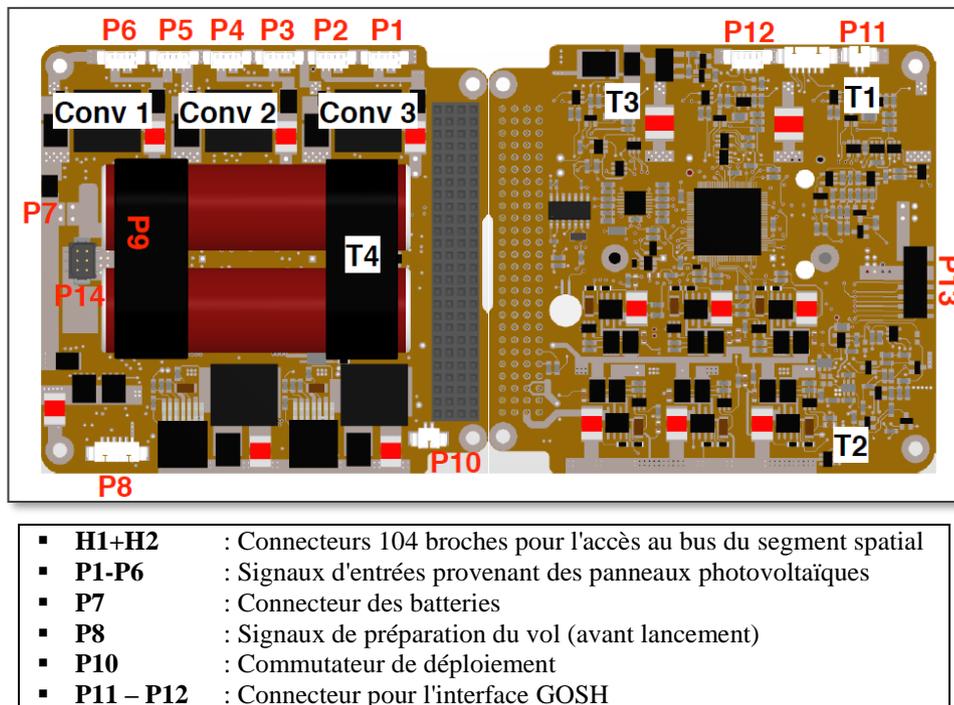


Figure 3-20 : Connecteurs de la carte NanoPower P31u [79]

Broche	Label	Direction	Description
H1.32	5V in	I	Entrée chargement de la batterie
H1.41	I2C – SDA	I/O	I ² C DATA
H1.43	I2C – SCL	I/O	I ² C CLOCK
H1.47	OUT-1	O	Sorties d'alimentation configurables et commandées avec limiteur de courant
H1.49	OUT-2	O	
H1.51	OUT-3	O	
H1.48	OUT-4	O	
H1.50	OUT-5	O	
H1.52	OUT-6	O	
H2.25	+5V	O	Sorties +5V permanente
H2.26	+5V	O	
H2.27	+3.3V	O	Sorties +3.3V permanente
H2.28	+3.3V	O	
H2.29	GND	O	Masse
H2.30	GND	O	
H2.31	AGND	O	
H2.32	GND	O	
H2.36	EPS RX	I	Interface de communication série
H2.38	EPS TX	O	
H2.45	V BAT	O	Sortie tension batterie
H2.46	V BAT	O	

Tableau 3-3 : Description des broches des connecteurs H1 et H2 [79]

4.3. Batteries

Tout engin spatial utilisant les panneaux PV comme source d'énergie primaire, nécessite l'utilisation d'une batterie rechargeable pour stocker de l'énergie pour les périodes d'éclipse et les demandes de pointe [8]. Son choix dépend des contraintes et paramètres suivants :

- La taille (10 cm x 10 cm x 10 cm) et le poids (1.3 kg) d'un Cubesat 1U.
- La température de fonctionnement qui est comprise entre -40 °C et +80°C.
- La durée de vie ou cycle de vie qui est défini par le nombre de cycles charge/décharge complets que peut supporter la batterie.
- La capacité qui doit permettre à la batterie d'emmagasiner suffisamment d'énergie pour faire fonctionner le segment spatial.

4.3.1. Exigences de stockage d'énergie et capacité de la batterie

Le dimensionnement de la capacité de la batterie a été réalisé en respectant les exigences de stockage d'énergie résumées par le *tableau 3-4* :

Considération	Valeur
Durée de la mission	▪ 1 an
Paramètres orbitaux (500km – 40°) : (Chapitre 1, § 7)	▪ 15 éclipses par jour ▪ 36 min ▪ 4766 par an
Profile d'utilisation de l'énergie :	▪ 1107 mWh ▪ 25%
Configuration des batteries :	▪ N = 1 (non) ▪ 2 batteries en série ▪ $\eta \cong 90\%$ (Mesuré de la batterie aux sorties d'alimentation)

Tableau 3-4 : Exigences de stockage d'énergie pour MASAT1

Le bilan d'énergie détaillé par la suite a permis d'estimer la consommation maximale d'énergie pendant l'éclipse, en considérant le mode de fonctionnement EP-TnCU : Eclipse avec Transmission et sans charge utile.

Sachant que la capacité des batteries diminue légèrement à chaque cycle de charge et de décharge, le meilleur moyen pour limiter cet effet est de choisir une faible profondeur de décharge DoD (Depth of Discharge) [8]. Vu que MASAT1 est une mission satellitaire en orbite LEO, on a choisi une DoD de 25%, ce qui signifie qu'une batterie doit se décharger à 25% de sa capacité avant d'être rechargée.

Pour estimer la valeur de la capacité de notre batterie, on utilise la formule [8] :

$$C = \frac{W_e}{DoD \cdot n \cdot \eta} = 4920(mWh) \quad (3.11)$$

Ainsi, les batteries doivent avoir une capacité minimale d'environ 9800 mWh (≈ 2648 mAh sous 3.7V), si on choisit un facteur 2 comme marge de sécurité.

4.3.2. Choix de la batterie

Pour MASAT1, on a choisi les piles rechargeables au lithium-ion 1850 de GOMSpace qui offrent les caractéristiques suivantes [82] :

- Tension nominale : 3.7V (varie de 3 à 4.2V)
- Capacité : 3.7V@2600mAh
- Cycle de vie (DoD 25%) : 2100
- Température de charge : entre -5°C et 45°C
- Température de décharge : entre -20°C et +60°C

5. BILAN DE PUISSANCE

Pour le segment spatial, la puissance électrique disponible est fixée par la surface externe de la structure et par le nombre de cellules solaires qui seront montées sur les faces de cette structure. La puissance électrique disponible sera alors fixe et limitée. Par conséquent le bilan de puissance qui est établi en fonction de la puissance consommée par les différents systèmes, imposera des contraintes spécifiques concernant les modes de fonctionnement du segment spatial.

5.1. Contraintes et hypothèses

Le bilan de puissance a été établi en se basant sur des contraintes fixées par l'étude orbitale de la mission et des hypothèses qui restent à confirmer dans la phase de test et d'intégration :

CHARGES		
Sous-système	Valeur	Unité
Energie de bord	160	mW
Réchauffeur de batterie	272	mW
Ordinateur de bord	130	mW
COM - Réception	180	mW
COM - Transmission	2640	mW
Antenne (nominal)	20	mW
Antenne (déploiement)	2000	mW
Panneaux solaires x5	10	mW
Charge utile (Ordinateur de bord secondaire)	1500	mW

PARAMETRES ORBITAUX (500 km – 40°)		
Durée orbite	94.61	min
Durée éclipse	35.75	min
Durée éclairement	58.86	min
Durée transmission moyenne /passage	8.9	min

AUTRES CONTRAINTES ET HYPOTHESES		
<ul style="list-style-type: none"> ▪ Rendement des convertisseurs DC-DC est de 95% ▪ COM-Rx toujours active sauf pendant la transmission de la balise et pendant la communication avec le sol ▪ Balise envoie un message de 212 octets/30s avec 1200 bps ▪ Les sous-systèmes EdB et OdB toujours activés ▪ CUT fonctionnera pendant 10 min quand elle est active ▪ Le nombre de panneaux solaires : 5 ▪ Surface des cellules solaires par panneau solaire : 60,36cm² ▪ Réchauffeur de batterie activée pendant la durée d'éclipse 		

Tableau 3-5 : Contraintes et hypothèses pour le bilan de puissance

5.2. Puissance générée par les cellules solaires

La puissance électrique instantanée maximale qui peut être capturée par les cellules solaires est donnée par :

$$P = C_s \cdot \eta \cdot A_{eff} \quad (3.12)$$

- η : Efficacité des panneaux solaires
- A_{eff} : Surface effective de la projection du panneau solaire sur un plan perpendiculaire à la direction des rayons solaires frappant le panneau.
- C_s : Constante solaire, dont la valeur moyenne est 1358 [watt/m2].

Pour déterminer la puissance générée par les cellules solaires, à l'aide de l'outil "Solar Panel" du logiciel STK, on a créé un modèle 3D du segment spatial à l'aides des logiciels SolidWorks et Modo :

- Le modèle contient un panneau PV, formée de deux cellules PV en série, sur cinq faces du Cubesat.
- Chacune des cellules PV a une efficacité moyenne de 29% qui peut être paramétrée dans le fichier ".dae" qui sera exporté vers STK :

```

<extra type="attach_point">
  <technique profile="AGI">
    <solar_panel group="Plus-Y" efficiency="29" />
  </technique>
</extra>

```

La **figure 3-21** montre le modèle 3D qui a été exporté vers STK pour modéliser l'exposition à la lumière du soleil des panneaux PV montés sur le Cubesat, en utilisant le type d'attitude : alignement Nadir avec la contrainte de vitesse ECI.

Ce type d'attitude reste possible à réaliser avec un contrôle d'attitude passif comme l'ont démontré les missions KySat-1 et KySat-2 [81] [96].

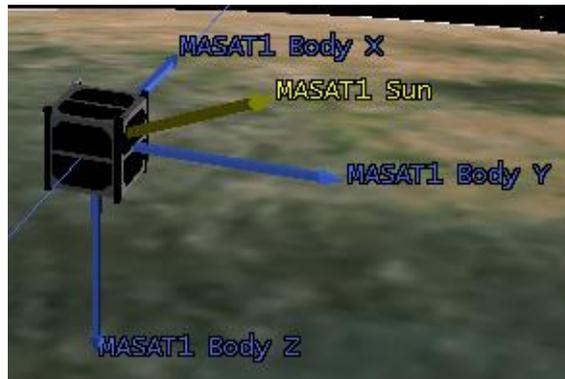


Figure 3-21 : Modèle 3D du Cubesat exporté vers STK

La **figure 3-22** résume les résultats obtenus pour une simulation sur une journée c'est-à-dire quinze orbites. On obtient ainsi :

- Une puissance maximale générée de 3369 mW.
- Une puissance moyenne générée pendant les périodes d'éclairement de 2186 mW.
- Une énergie potentiellement générée par orbite de 2035 mWh.

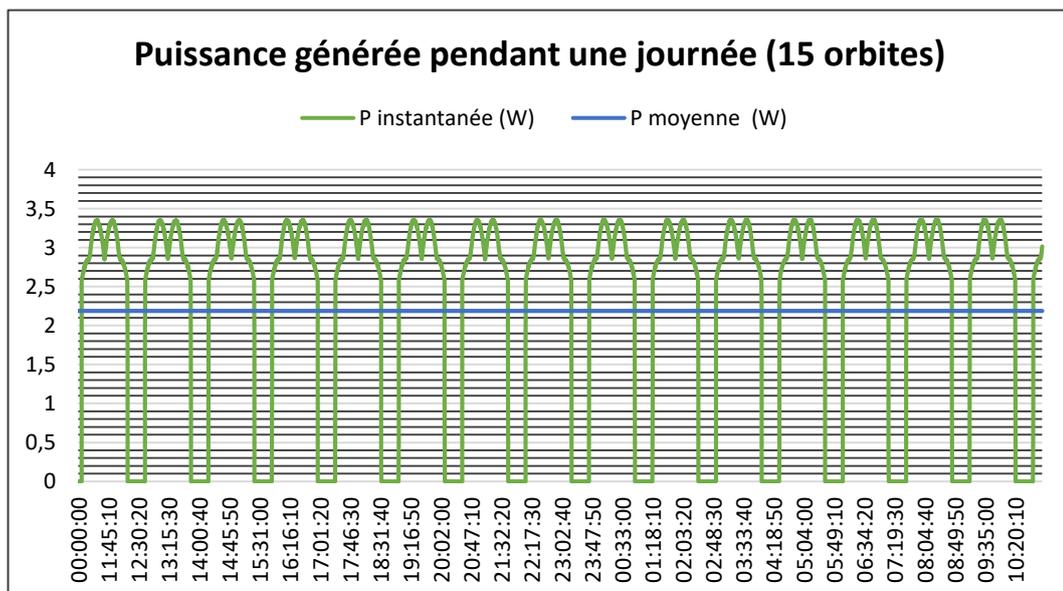


Figure 3-22 : Puissance instantanée générée par les cellules PV [STK]

5.3. Puissance consommée en mode nominal

Selon le concept des opérations de MASAT1 (§ 3.7), une fois que le segment spatial pourra fonctionner en mode nominal, il y aura 8 différents cas de consommation d'énergies possibles (4 pendant l'éclairement du segment spatial et 4 autres pendant l'éclipse) comme le montre le *tableau 3-6* :

Modes	Description
EL-TCU	Transmission et CU (FPGA) opérationnelles pendant l'éclairement
EL-TnCU	Transmission opérationnelle et CU non opérationnelle pendant l'éclairement
EL-nTCU	Transmission non opérationnelle et CU opérationnelle pendant l'éclairement
EL-nTnCU	Transmission et CU (FPGA) non opérationnelles pendant l'éclairement
EP-TCU	Transmission et charge utile (FPGA) opérationnelles pendant l'éclipse
EP-TnCU	Transmission opérationnelle et CU non opérationnelle pendant l'éclipse
EP-nTCU	Transmission non opérationnelle et CU opérationnelle pendant l'éclipse
EP-nTnCU	Transmission et CU (FPGA) non opérationnelles pendant l'éclipse

Tableau 3-6 : Cas de consommation d'énergie possibles en mode nominal

La consommation d'énergie est calculée en fonction de la puissance consommée par chacun des sous-systèmes pendant l'éclairement et l'éclipse, avec une marge de 20% qui est ajoutée à la consommation totale :

Sous-système	P (mW)	Durée (hrs)	Energie (mWh)
EDB	160	0,98	156,96
OBC	130	0,98	127,53
COM-RX	180	0,77	138,11
COM-TX	2640	0,15	391,60
CUT	1500	0,17	250,00
Antenne	20	0,98	19,62
Panneau PV	10	0,98	9,81
Balise	2640	0,07	172,66
Total			1266,28
Total + 20%			1519,54

Tableau 3-7 : Consommation mode EL-TCU

Sous-système	P (mW)	Durée (hrs)	Energie (mWh)
EDB	160	0,98	156,96
OBC	130	0,98	127,53
COM-RX	180	0,92	164,81
COM-TX	2640	0	0,00
CUT	1500	0,17	250,00
Antenne	20	0,98	19,62
Panneau PV	10	0,98	9,81
Balise	2640	0,07	172,66
Total			901,38
Total + 20%			1081,66

Tableau 3-8 : Consommation mode EL-nTCU

Sous-système	P (mW)	Durée (hrs)	Energie (mWh)
EDB	160	0,98	156,96
OBC	130	0,98	127,53
COM-RX	180	0,77	138,11
COM-TX	2640	0,15	391,60
CUT	1500	0,00	0,00
Antenne	20	0,98	19,62
Panneau PV	10	0,98	9,81
Balise	2640	0,07	172,66
Total			1016,28
Total + 20%			1219,54

Tableau 3-9 : Consommation mode EL-TnCU

Sous-système	P (mW)	Durée (hrs)	Energie (mWh)
EDB	160	0,98	156,96
OBC	130	0,98	127,53
COM-RX	180	0,92	164,81
COM-TX	2640	0,00	0,00
CUT	1500	0,00	0,00
Antenne	20	0,98	19,62
Panneau PV	10	0,98	9,81
Balise	2640	0,07	172,66
Total			651,38
Total + 20%			781,66

Tableau 3-10 : Consommation mode EL-nTnCU

Sous-système	P (mW)	Durée (hrs)	Energie (mWh)
EDB	432	0,60	257,40
OBC	130	0,60	77,46
COM-RX	180	0,41	73,40
COM-TX	2640	0,15	391,60
CUT	1500	0,17	250,00
Antenne	20	0,60	11,92
Panneau PV	10	0,60	5,96
Balise	2640	0,04	104,87
Total			1172,60
Total + 20%			1407,12

Tableau 3-11 : Consommation mode EP-TCU

Sous-système	P (mW)	Durée (hrs)	Energie (mWh)
EDB	432	0,60	257,40
OBC	130	0,60	77,46
COM-RX	180	0,56	100,10
COM-TX	2640	0	0,00
CUT	1500	0,17	250,00
Antenne	20	0,60	11,92
Panneau PV	10	0,60	5,96
Balise	2640	0,04	104,87
Total			807,70
Total + 20%			969,24

Tableau 3-12 : Consommation mode EP-nTCU

Sous-système	P (mW)	Durée (hrs)	Energie (mWh)
EDB	432	0,60	257,40
OBC	130	0,60	77,46
COM-RX	180	0,41	73,40
COM-TX	2640	0,15	391,60
CUT	1500	0,00	0,00
Antenne	20	0,60	11,92
Panneau PV	10	0,60	5,96
Balise	2640	0,04	104,87
Total			922,60
Total + 20%			1107,12

Tableau 3-13 : Consommation mode EP-TnCU

Sous-système	P (mW)	Durée (hrs)	Energie (mWh)
EDB	432	0,60	257,40
OBC	130	0,60	77,46
COM-RX	180	0,56	100,10
COM-TX	2640	0	0,00
CUT	1500	0,00	0,00
Antenne	20	0,60	11,92
Panneau PV	10	0,60	5,96
Balise	2640	0,04	104,87
Total			557,70
Total + 20%			669,24

Tableau 3-14: Consommation mode EP-nTnCU

Pour pouvoir élaborer un scénario opérationnel sur plusieurs orbites, en fonction des modes de consommation de puissance, on détermine les combinaisons de consommation d'énergie en mode nominal qui sont résumées dans le *tableau 3-15* :

	EL-TCU	EL-TnCU	EL-nTCU	EL-nTnCU
EP-TCU	2926,66	2626,66	2488,78	2188,78
EP-TnCU	2626,66	2326,66	2188,78	1888,78
EP-nTCU	2488,78	2188,78	2050,90	1750,90
EP-nTnCU	2188,78	1888,78	1750,90	1450,90

Tableau 3-15 : Combinaisons de consommation d'énergie en mode nominal

5.4. Scénario de consommation

Selon l'étude orbitale (500km, 40°) réalisée au Chapitre N°1 (§ 3.5.2), le segment spatial orbitera environ 15 fois autour de la Terre avec six occasions pour communiquer avec la station au sol. Un des scénarios proposés est que :

- La communication avec la station au sol sera établie six fois par jour.
- La charge utile fonctionnera trois fois par jour pendant une orbite sans communication avec la station au sol.

- Pendant les six orbites restantes, aucune charge utile ni aucune transmission avec la station au sol ne seront réalisées.

Le **tableau 3-16** résume la puissance totale consommée en une journée pour un tel scénario. L'énergie consommée pour chaque orbite est additionnée pour déterminer la consommation totale par jour et comme on peut le voir dans le tableau, la puissance totale consommée en une journée est inférieure à la quantité totale d'énergie produite pendant les 15 orbites.

Mode	Nombre d'orbites	Energie/Mode	Energie
EL-TCU/EP-TCU		2926,66	0,00
EL-TCU/EP-TnCU		2626,66	0,00
EL-TCU/EP-nTCU		2488,78	0,00
EL-TCU/EP-nTnCU		2188,78	0,00
EL-TnCU/EP-TCU		2626,66	0,00
EL-TnCU/EP-TnCU		2326,66	0,00
EL-TnCU/EP-nTCU		2188,78	0,00
EL-TnCU/EP-nTnCU	6	1888,78	11332,68
EL-nTCU/EP-TCU		2488,78	0,00
EL-nTCU/EP-TnCU		2188,78	0,00
EL-nTCU/EP-nTCU		2050,90	0,00
EL-nTCU/EP-nTnCU	3	1750,90	5252,70
EL-nTnCU/EP-TCU		2188,78	0,00
EL-nTnCU/EP-TnCU		1888,78	0,00
EL-nTnCU/EP-nTCU		1750,90	0,00
EL-nTnCU/EP-nTnCU	6	1450,90	8705,40
Puissance Totale Consommée			25290,79
Puissance Totale Produite	15	2035,00	30525,00

Tableau 3-16 : Scénario de consommation d'énergie en mode nominal

Chapitre 4

ETUDE ET DESCRIPTION DU SOUS- SYSTEME COMMUNICATION

1. INTRODUCTION

Le sous-système COM constitue la passerelle permettant aux opérateurs de la station au sol d'accéder au segment spatial. La communication consiste en deux types de liaison :

- Une liaison montante permettant au segment spatial de recevoir les TC provenant de la station au sol, afin de les exécuter via l'OdB.
- Une liaison descendante permettant à la station au sol de recevoir toutes les données de TM et de la charge utile provenant du segment spatial, pour les traiter.

La *figure 4-1* donne le diagramme bloc fonctionnel d'un sous-système COM :

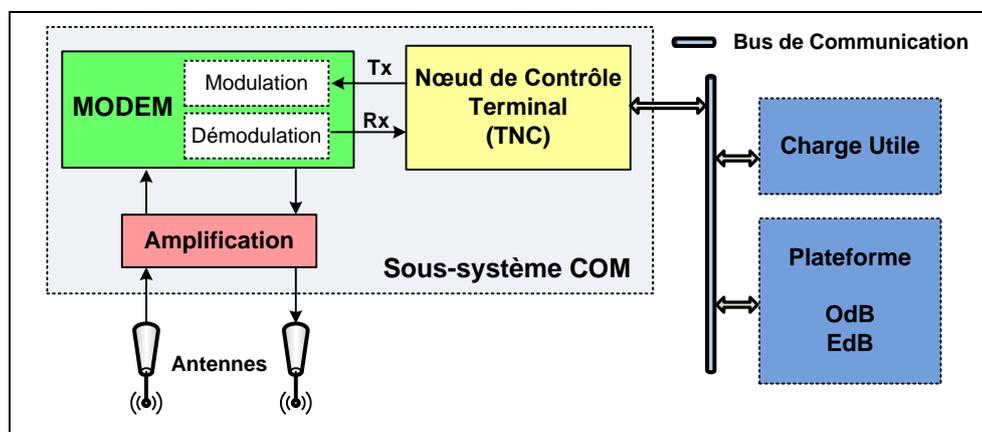


Figure 4-1 : Diagramme bloc fonctionnel d'un sous-système COM

Le TNC (Terminal Node Control) est un microcontrôleur qui réalise les fonctions suivantes :

Liaison montante	<ul style="list-style-type: none"> ▪ Lire les données (TM et charge utile) à transmettre et les conserver si nécessaire. ▪ Coder les données selon le format du protocole de communication. ▪ Envoyer les données au modem.
Liaison descendante	<ul style="list-style-type: none"> ▪ Recevoir les commandes depuis le modem. ▪ Décoder les commandes selon le format du protocole de communication. ▪ Rediriger les commandes vers l'OdB via le bus de communication série de la plateforme.

Le Modem permet de :

- Moduler les trames reçues du TNC avant de les envoyer vers la station au sol.
- Démoduler le signal reçu de la station au sol pour récupérer les commandes.

2. EXIGENCES DU SOUS-SYSTEME COM POUR LA MISSION MASAT1

La tâche principale du module COM est de permettre un contact permanent entre le segment spatial et les stations au sol tout en permettant de satisfaire aux exigences préliminaires spécifiées dans le DES [71] :

[DES] OPER-1145

Le récepteur doit toujours être fonctionnel et disponible à tout moment pour assurer la réception des télécommandes transmises par le centre de contrôle de la station au sol.

[DES] FONC-1150

Le nanosatellite doit être en mesure de transmettre des données (TM et charge utile) à la station au sol.

[DES] FONC-1160

Le nanosatellite doit être en mesure de recevoir des commandes à partir de la station au sol

[DES] FONC-1170

Le sous-système COM doit être capable de résister aux conditions environnementales.

[DES] FONC-1175

Le sous-système COM doit être capable de communiquer avec des stations au sol radio amateurs externes à la mission MASAT1.

[DES] CONC-1180

Le nanosatellite doit transmettre et recevoir sur des fréquences fixes et autorisées (autorisation d'organismes officielles comme l'IARU "International Amateur Radio Union").

[DES] CONC-1190

Le sous-système COM doit être compatible au bus adopté dans l'architecture du système.

[DES] CONC-1195

Le taux d'erreur par bit pour toute transmission du signal RF doit être inférieur ou égale à 10^{-5} pour assurer la fiabilité de transmission requise.

[DES] CONC-1196

Une marge de liaison minimale de 6 dB est nécessaire pour tenir compte des conditions réelles de propagation et des pertes inattendues de conception, et ceci pour toutes les liaisons de communication avec le segment spatial.

[DES] CONC-1200

Le débit de données pour la télémétrie doit être dérivé des exigences relatives à la supervision et à la charge utile du segment spatial du nanosatellite.

[DES] CONC-1210

Le débit de données pour les télécommandes doit être déduit des contraintes de commande et d'exploitation du segment spatial du nanosatellite.

3. INTERFACES ELECTRIQUES

Les interfaces électriques du sous-système COM sont résumées par la *figure 4-2*. On distingue les principaux signaux suivants :

- L'entrée/sortie *RF_Signal* qui représente le signal provenant ou partant vers l'antenne UHF.
- L'entrée d'alimentation *Vcc* provenant du sous-système EdB
- L'interface I2C de communication et les interfaces JTAG et série pour respectivement la configuration et le débogage.

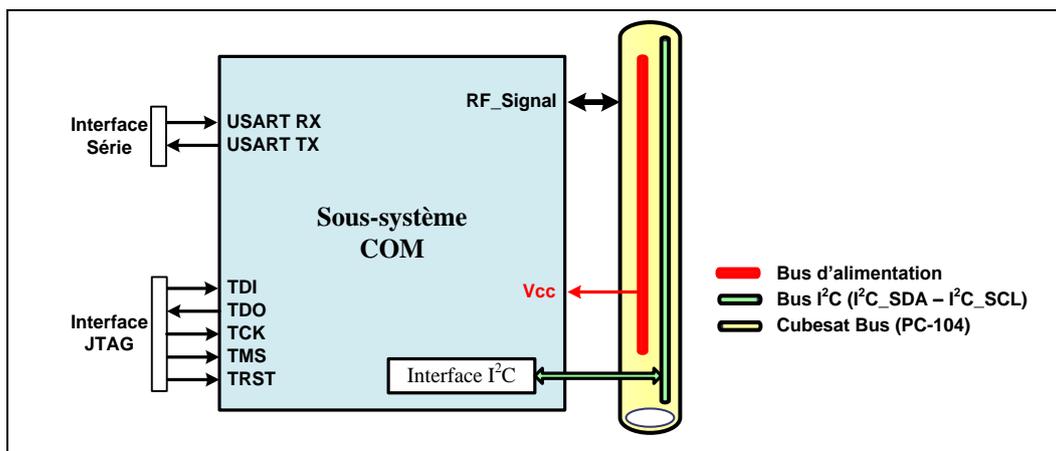


Figure 4-2 : Interfaces électriques du sous-système COM

4. ARCHITECTURE DU SOUS-SYSTEME COM DE LA MISSION MASAT1

Comme le montre la *figure 4-3*, l'architecture adoptée pour le sous-système COM de la mission MASAT1 peut être divisée en trois principales parties : un émetteur radio, un récepteur radio et un bloc de traitement de données à base de microcontrôleur.

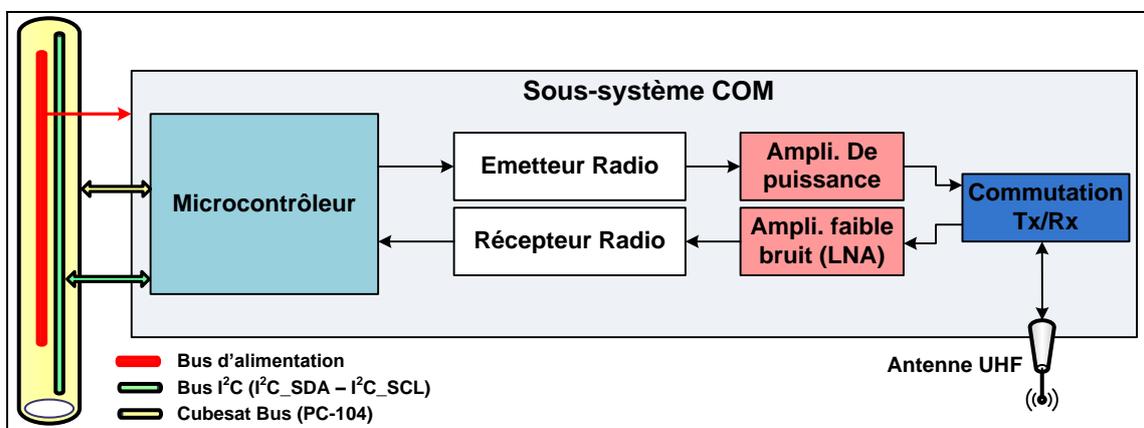


Figure 4-3 : Architecture du sous-système COM

Vu le choix d'une communication UHF half-duplex (Chap. 2 § 3.5) :

- Le récepteur et l'émetteur sont des dispositifs radio de base hétérodynes et leurs fréquences seront configurées pour avoir une liaison descendante et une liaison montante en bande UHF.
- Une seule antenne UHF sera utilisée d'où la nécessité d'utiliser un commutateur Tx/Rx permettant de connecter l'antenne soit avec l'émetteur, soit avec le récepteur.

Le bloc de traitement des données aura pour rôle :

- La communication avec l'OdB ou l'EdB (selon le mode opérationnel) via le bus de communication série I2C.
- Le traitement des données de télémétrie (codage) et des télécommandes (décodage) en utilisant le protocole AX25 pour la communication station sol \leftrightarrow segment spatial.

Le choix du protocole AX25 a été dicté par les raisons suivantes :

- Large utilisation par les radioamateurs et Conforme aux standards ISO.
- Couverture de la couche physique et la couche liaison.
- Indépendance des couches supérieures.
- Assure une communication point à point fiable.

Une étude plus détaillée du protocole AX25 est donnée dans la partie description logicielle de la mission MASAT1.

5. BILAN DE LIAISON

Cette section a été réalisée en collaboration avec l'équipe de l'université Al Akhawayn en charge de la conception du sous-système COM. Elle présente l'état de l'art et les différents paramètres qui ont permis d'établir un bilan de liaison préliminaire permettant de valider les spécifications pour le choix du sous-système de communication COM. Ce bilan prend en compte les différents types de pertes dans la chaîne de transmission et il doit être positif pour avoir une forte probabilité de communication fiable entre le segment spatial et la station au sol.

La finalité du bilan de liaison est donc d'estimer la qualité du signal ainsi que le dimensionnement de l'émetteur, du récepteur et des limites en termes de performance (taux d'erreur, puissance minimale, amplification nécessaire, etc.).

5.1. Contraintes orbitales

Lors de la conception du système de communication, on doit tenir compte des caractéristiques de l'orbite choisie ainsi que du type de contrôle d'attitude adopté.

Une étude orbitale préliminaire a proposé une orbite circulaire candidate : altitude de 650 Km et inclinaison de 90°. Les calculs du bilan de liaison seront faits pour cette orbite limite.

Pour le système de contrôle d'attitude, on a opté pour un contrôle d'attitude passif par stabilisation magnétique qui permet d'aligner le satellite avec le vecteur du champ magnétique terrestre local, pour ensuite contrôler le Cubesat dans deux directions autres que l'axe magnétique qui reste incontrôlé (choix suffisant pour les missions Cubesat qui requièrent une précision de pointage de l'ordre de 10° [8]).

5.2. Spécifications du module de transmission du segment spatial

Pour tout système de communication, les spécifications importantes qui doivent être connues avant le calcul du bilan de liaison sont : la puissance, le débit de données et la technique de modulation. Les *figures 4-4* et *4-5* donnent une étude statistique des solutions technologiques adoptées pour les Cubesat (entre 2003 et 2016) :

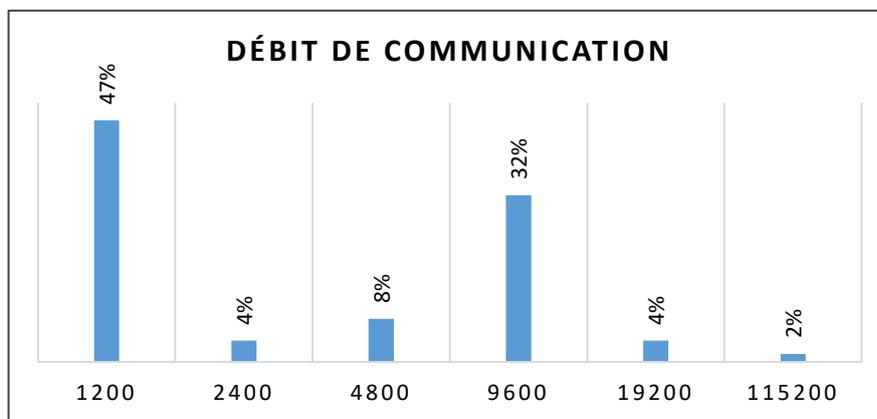


Figure 4-4 : Débits de communication utilisés dans les missions Cubesat [83]

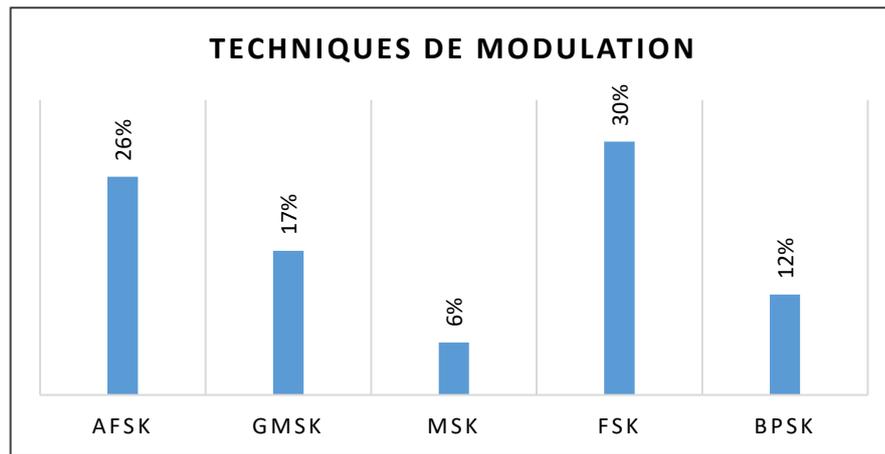


Figure 4-5 : Techniques de modulation utilisés dans les missions Cubesat [83]

5.2.1. Débit de transmission

Le bilan de liaison a été établi, à la fois pour la liaison montante et celle descendante, en prenant en compte les deux cas limites les plus utilisés : 1200bps et 6900bps. Il est à noter que l'augmentation du débit, entrainera une augmentation de la puissance et du bruit [8].

5.2.2. Technique de modulation

Le choix d'une technique de modulation donnée va influencer la largeur de bande passante et la puissance. L'étude réalisée par l'équipe du sous-système COM a conclu que pour assurer une efficacité en puissance et en bande passante, on doit utiliser une modulation MSK ou GMSK vu leurs performances en termes de BER (Bit Error rate) et d'efficacité spectrale comme le montre les *figure 4-6* et *4-7*. La modulation FSK peut toujours être utilisée mais pour de faibles débits de l'ordre de 1200 bps.

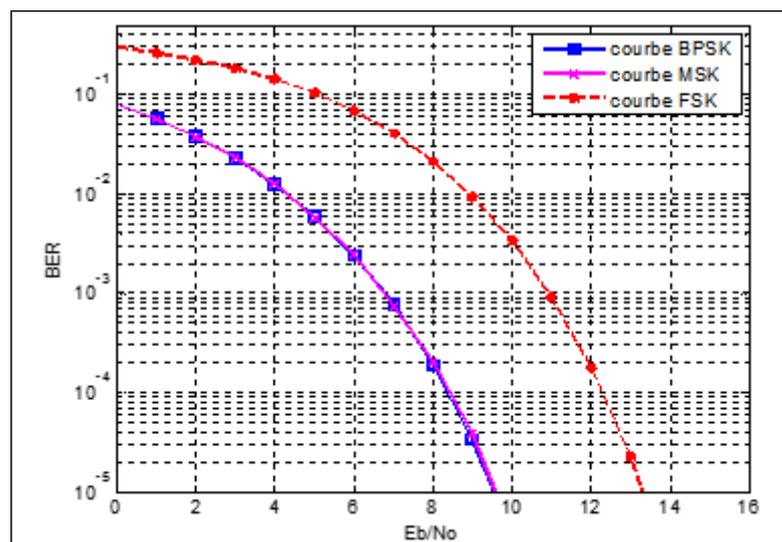


Figure 4-6 : Taux d'erreur BER pour les modulations BPSK, MSK et FSK [84]

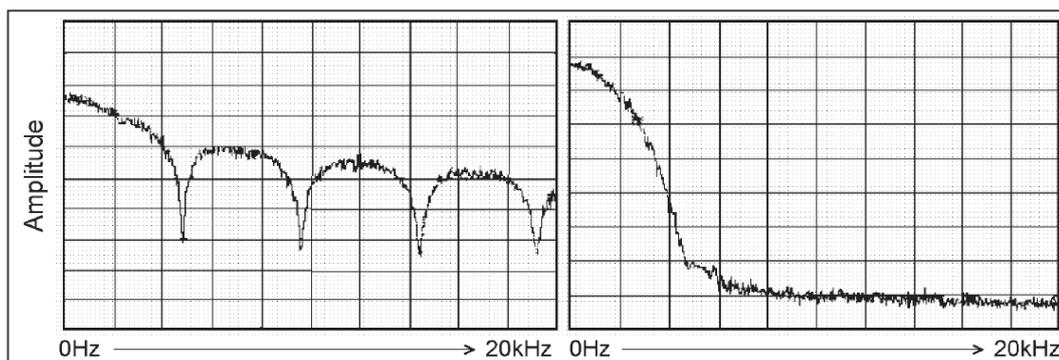


Figure 4-7 : Densité spectrale (à gauche : Signal FSK et à droite : signal GMSK)

5.2.3. Consommation d'énergie

Pour la liaison descendante et puisqu'il existe des contraintes de puissance dans le segment spatial, la puissance consommée par l'émetteur doit être minimisée sachant que pour un CubeSat 1U, la puissance maximale pouvant être consommée pendant la transmission des données est comprise entre 2 et 3W.

Pour la liaison montante, la puissance n'est pas un critère principal, mais une puissance élevée peut déformer les signaux dans des fréquences adjacentes.

5.2.4. Fréquence de fonctionnement

Vu les choix réalisés au niveau des solutions de conception de MASAT1 (Chapitre 2 § 3.5), les communications montantes et descendantes doivent avoir des fréquences différentes dans la bande de fréquence UHF. En attendant d'entrer en contact avec l'Agence Nationale de Réglementation des Télécommunications (ANRT) et un membre du Groupe consultatif IARU pour valider les choix de fréquences de communication, nous avons décidé d'utiliser 2 canaux de communications : 437MHz pour les liaisons montantes et 435MHz pour les liaisons descendantes.

5.3. Modélisation du canal de transmission

Comme le montre la *figure 4-8*, le canal de transmission entre le segment spatial et la station au sol est défini par plusieurs paramètres de base qui sont utilisés dans l'évaluation de la fiabilité de la communication :

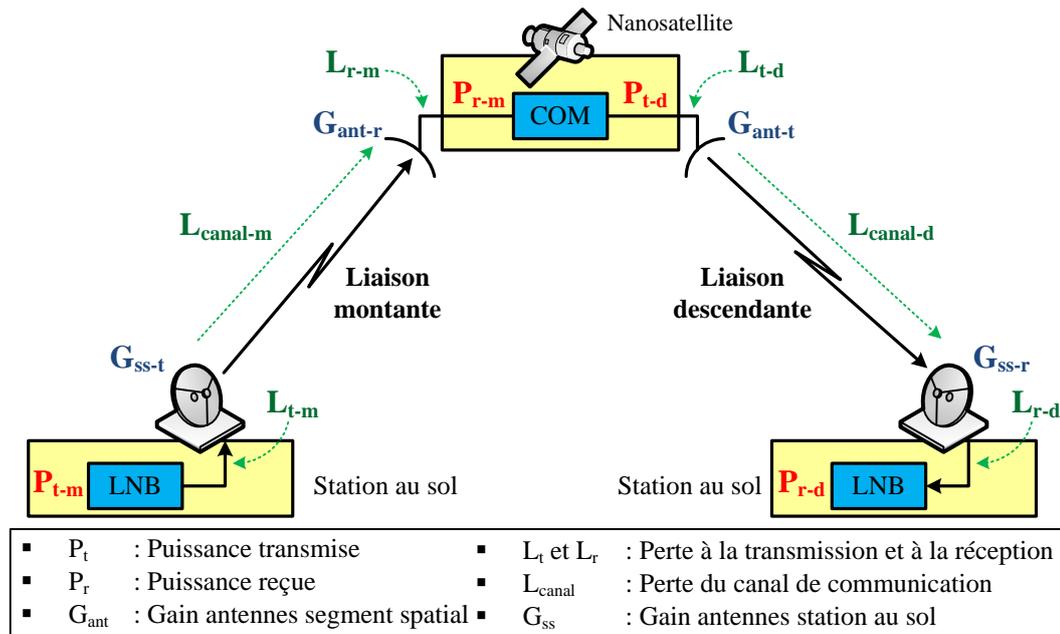


Figure 4-8 : Modélisation de la communication entre segment spatial et station au sol

Pour déterminer la puissance reçue aux bornes d'une antenne réceptrice dans une communication par satellite, on utilise l'équation d'équilibre des puissances qui représente l'équation de base d'un bilan de liaison [85] :

$$\mathbf{Puissance}_{re\grave{c}ue} = \mathbf{Puissance}_{transmise} + \mathbf{Gain}_{antennes} - \mathbf{Pertes} \quad (4.1)$$

Si on considère par exemple la liaison montante, la puissance reçue par l'antenne du segment spatial est donnée par la relation suivante :

$$P_{r-m} = P_{t-m} + G_{ss-t} + G_{ant-r} - L_{t-m} - L_{canal-m} - L_{r-m} \quad (4.2)$$

Avec :

- P_{r-m} : Puissance reçue par l'antenne du segment spatial.
- P_{t-m} : Puissance transmise par la station au sol.
- G_{ss-t} : Gain de l'antenne émettrice de la station au sol.
- G_{ant-r} : Gain de l'antenne réceptrice du segment spatial.
- L_{t-m} : Perte de transmission au niveau de la station sol qui comprennent les pertes de ligne et les pertes de pointage.
- L_{r-m} : Perte de réception au niveau du segment spatial qui comprennent les pertes de ligne et les pertes de pointage.
- $L_{canal-m}$: Perte du canal de communication qui comprennent les pertes de polarisation, les pertes atmosphériques et les pertes à l'air libre.

5.4. Paramètres des antennes

5.4.1. Densité de puissance

Pour définir le concept de densité de puissance, on va considérer une antenne isotrope ou omnidirectionnelle qui est une antenne idéale sans pertes qui rayonne une puissance P_t de manière uniforme dans toutes les directions de l'espace. La densité de puissance est alors définie comme la puissance rayonnée par une antenne isotrope à une distance r , par unité de surface [87] :

$$\psi = \frac{P_t}{4 \cdot \pi \cdot r^2} \quad (4.3)$$

5.4.2. Gain d'une antenne

C'est la principale caractéristique de l'antenne qui entre en jeu dans les calculs du bilan de liaison. Il est défini comme le rapport entre la densité de puissance moyenne rayonnée par l'antenne dans une direction θ et la densité de puissance rayonnée par une antenne isotrope dans la même direction, les deux antennes étant alimentées par la même puissance P_t [86] :

$$G = \frac{\text{Densité de puissance de l'antenne dans une direction donnée}}{\text{Densité de puissance d'une antenne isotrope dans la même direction}}$$

$$G = \frac{P(\theta)}{P_0} = \frac{P(\theta)}{P_t \cdot 4\pi} \quad (dBi) \quad (4.4)$$

L'unité dBi du gain vient du fait qu'une antenne isotrope est utilisée comme référence.

L'une des relations fondamentales pour le gain d'une antenne est sa relation avec la longueur d'onde du signal à transmettre ou à recevoir [87] :

$$G = \frac{4\pi \cdot A_{eff}}{\lambda^2} \quad (4.5)$$

Avec :

- λ : longueur d'onde.
- A_{eff} : Surface d'ouverture effective de l'antenne.

5.4.3. Pertes de polarisation

La polarisation d'une onde électromagnétique (OEM) est la direction de son champ électrique E . Si E garde une direction constante, on dit que la polarisation est rectiligne et c'est le cas le plus courant. On parle alors de polarisation horizontale si le plan de vibration électrique est horizontal, et de polarisation verticale si le plan de vibration électrique est vertical.

Il existe aussi des polarisations circulaires comme le montre la *figure 4-9* :

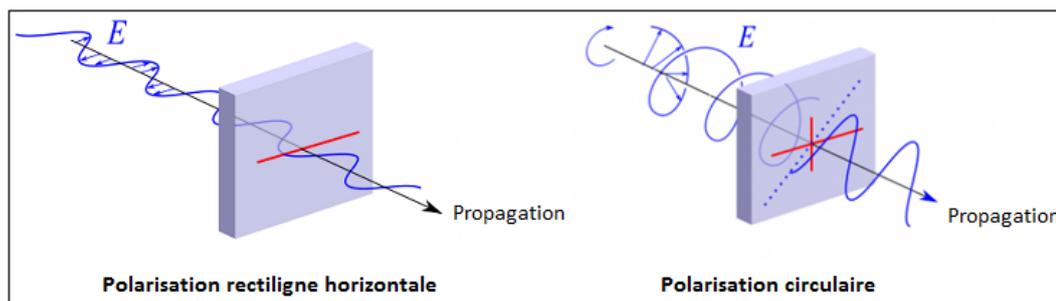


Figure 4-9 : Exemples de polarisation d'une onde électromagnétique

La polarisation d'une onde dépend de la géométrie et de la polarisation électrique de l'antenne émettrice. Idéalement, pour diminuer les pertes de polarisation et optimiser la réception d'un signal radioélectrique, la polarisation de l'onde électromagnétique et celle de l'antenne réceptrice doivent être les mêmes. Le *tableau 4-1* répertorie un ensemble de différentes combinaisons de polarisation et les pertes de polarisation qu'elles entraînent [84] :

Combinaison de polarisations	Pertes de polarisation (dB)
Verticale – Verticale	0
Verticale – horizontale	$-\infty$
Horizontale – Horizontale	0
Circulaire (droite ou gauche) – rectiligne	-3
Circulaire droite – circulaire droite	0
Circulaire droite – circulaire gauche	$-\infty$

Tableau 4-1 : Pertes de polarisation des antennes [84]

Pour limiter les pertes de polarisation et vu que l'orientation d'un Cubesat est imprévisible, nous avons choisi d'utiliser pour la mission MASAT1 une polarisation circulaire au niveau de la station au sol et une polarisation rectiligne via une antenne dipôle au niveau du segment spatial. Ce choix limite les pertes de polarisation dans le bilan de liaison à -3dB.

5.5. Pertes de propagation radiofréquence dans l'espace

5.5.1. Pertes à l'air libre (FSPL : Free Space Path Loss)

Le calcul des pertes dans l'espace libre est une étape de base pour le calcul d'un bilan de liaison de communication par satellite. Pour notre système de communication, on va supposer que l'antenne de transmission et l'antenne de réception sont alignées face à face dans l'espace libre et séparées par une distance suffisamment élevée d (mètres).

Ces pertes sont données par l'équation de Friis tel que [85] [87] :

$$L_f = \frac{(4 \cdot \pi \cdot d)^2}{\lambda^2} \quad (4.6)$$

En forme logarithmique, les pertes à l'air libre s'écrivent sous la forme :

$$FSPL = 20 \cdot \log\left(\frac{4 \cdot \pi \cdot d}{\lambda}\right) \text{ dB} \quad (4.7)$$

La **figure 4-10** montre que les pertes à l'air libre augmentent avec la fréquence et sont minimales lorsque le satellite est directement au-dessus de la station sol (angle d'élévation égal à 90°).

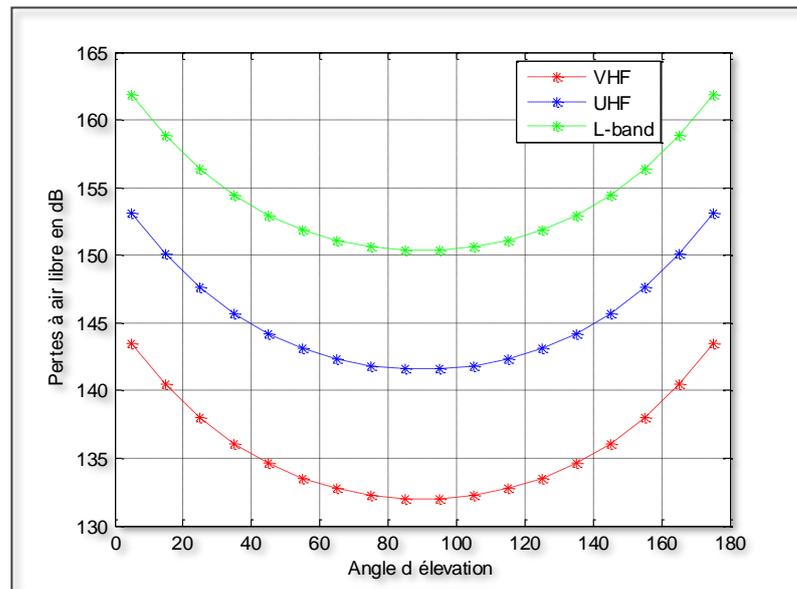


Figure 4-10 : Pertes à l'air libre pour les différentes bandes de fréquences [84]

5.5.2. Effet doppler

L'effet doppler représente la désynchronisation en fréquence, entre le signal transmis et le signal reçu, résultante du mouvement du satellite à une grande vitesse de l'ordre de 8 Km/s. Il est donné par la relation suivante [84] :

$$f_{dopp} = \frac{v_c \cdot f}{c} \cdot \frac{R \cdot \cos(\alpha)}{R + h} \quad (4.8)$$

Avec :

- f : fréquence du signal transmis
- v_c : vitesse orbitale du satellite
- c : célérité de la lumière dans le vide
- R : rayon de la terre (6378 Km)
- α : angle d'élévation
- h : altitude de l'orbite

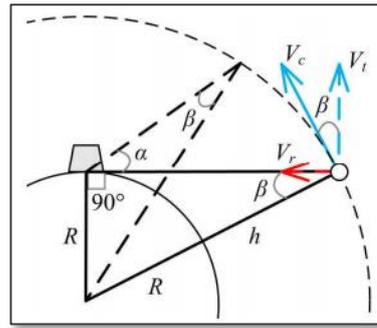


Figure 4-11 : La composante radiale de la vitesse radiale d'un satellite [84]

Comme le montre la **figure 4-12**, la déviation de la fréquence par effet doppler varie linéairement avec la fréquence de transmission et augmente avec la fréquence :

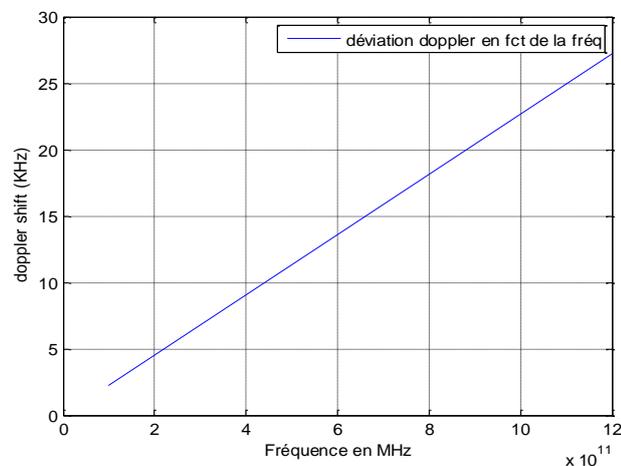


Figure 4-12 : Déviation Doppler en fonction de la fréquence de transmission [84]

5.5.3. Pertes atmosphériques

Une liaison radioélectrique entre le sol et un satellite traverse l'atmosphère terrestre et entre en collision avec les molécules et les particules qui la compose. On distingue alors deux régions susceptibles d'influencer la propagation des ondes :

- La troposphère composée de particules gazeuses qui entraînent un transfert d'énergie et l'absorption du rayonnement qui cède tout ou partie de son énergie, conduisant par conséquent à une atténuation du signal dans la direction de propagation. Les principales molécules concernées par cette atténuation sont l'oxygène et la vapeur d'eau (gouttes d'eau et particules de glace dont le rayon est suffisamment petit par rapport à la longueur d'onde du signal radio).
- L'ionosphère qui est la couche supérieure de l'atmosphère contenant essentiellement des particules ionisées. Son principal effet est la scintillation qui est le résultat de fluctuations rapides d'amplitude, de phase et d'angle d'arrivée des signaux reçus après la traversée de l'atmosphère.

Ces fluctuations sont dues soit à la variation de l'indice de réfraction provoquées soit par la présence de turbulences atmosphériques, soit par des irrégularités de l'ionisation au sein de l'ionosphère causée par les activités solaires.

Les pertes atmosphériques totales ont été calculées (voir *figure 4-13*) à l'aide des modèles ITU-R P.676-618-840 de l'Union Internationale des Télécommunications (ITU), qui sont des modèles empiriques et statistiques développés au cours des deux dernières décennies pour la prévision de l'atténuation atmosphérique sur un lien spatial [85] [87] [148] :

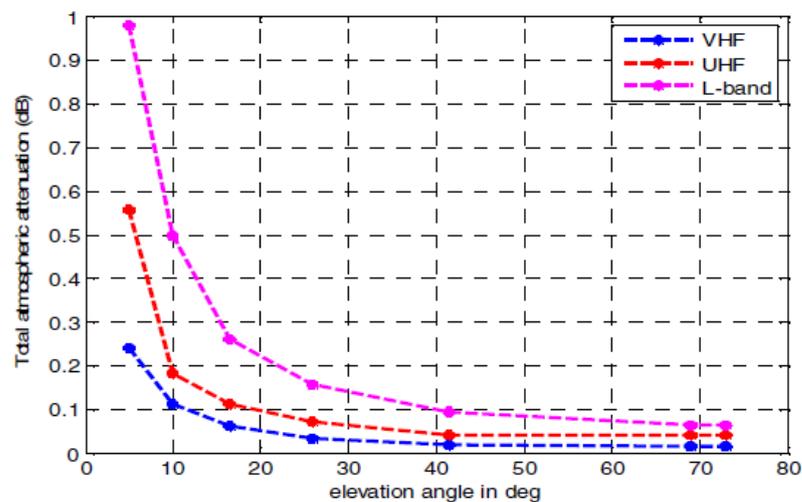


Figure 4-13 : Atténuations atmosphériques en fonction de l'angle d'élévation [84]

Les résultats précédents montrent que les atténuations de propagation RF dans l'espace et l'effet Doppler augmentent avec la fréquence, ce qui peut réduire considérablement la marge de liaison. Donc si on veut limiter ces atténuations tout en conservant un débit des communications élevé, le choix des bandes de fréquence UHF pour la mission MASAT1 s'avère être un juste milieu.

5.6. Bruit thermique

L'agitation thermique des électrons dans les dispositifs passifs et actifs qui ne sont pas à température nulle produit une différence de tension entre leurs bornes qui peut être assimilée à un bruit thermique. L'équation de Nyquist suivante permet de déterminer ce bruit thermique au niveau d'un récepteur RF [88] :

$$P_N = k \cdot B \cdot T_S \text{ (W)} \quad (4.9)$$

Avec :

- P_N : Puissance du bruit thermique.
- k : constante de Boltzmann ($1.39 \cdot 10^{-23} J/^{\circ}K$ ou $-198 dBm/^{\circ}K/Hz$)
- T_S : Température de bruit équivalente à la source de bruit, en $^{\circ}K$
- B : Bande passante du bruit.

La puissance du bruit thermique d'un système de réception RF est indépendante de la puissance de transmission du système, et dépend du bruit de la température ambiante qui provient de plusieurs sources : le ciel, l'atmosphère, l'antenne, les composants passifs et actifs du système et d'autre signaux parasites supplémentaires.

La température du bruit est ainsi donnée par la formule suivante [87] :

$$T_S = T_{antenne} + T_{récepteur} \quad (4.10)$$

Avec :

- $T_{antenne}$: bruit reçu par l'antenne incluant le bruit du ciel (bruit de fond galactique), le bruit thermique atmosphérique et le bruit thermique terrestre. Cette température est obtenue à partir de la convolution de la température de luminosité T_S d'un corps rayonnant situé dans une direction (θ, φ) (comme le ciel, le soleil et la terre) et le diagramme de rayonnement de l'antenne avec un gain G [88] :

$$T_{antenne} = \frac{1}{4\pi} \iint T_b(\theta, \varphi) \cdot G \cdot \sin \theta \cdot d\theta \cdot d\varphi \quad (4.11)$$

- $T_{récepteur}$: bruit thermique générée par les composants passifs et actifs du récepteur et plus spécialement l'amplificateur à faible bruit LNA. Cette température de bruit est dérivée de la formule de FRIIS du bruit pour n composants en cascade [85] :

$$F_{récepteur} = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_2 \cdot G_1} + \dots + \frac{F_n - 1}{G_2 \cdot G_1 \dots G_{n-1}} \quad (4.12)$$

$$T_{récepteur} = T_0 \cdot (F_{récepteur} - 1) \quad (4.13)$$

Avec :

- G_i : Gain du composant s'il est actif.
- F_i : Facteur de bruit du composant défini par le quotient des rapports signal sur bruit (exprimés en puissance) en sortie et en entrée.
- T_0 : Température référence du bruit qui est égale à $290^{\circ}K$.
- $F_{récepteur}$: Facteur de bruit du récepteur

Pour le calcul des températures du bruit pour les liaisons montante et descendante de la mission MASAT1, nous avons utilisé le logiciel de simulation STK en prenant en compte l'influence de la Terre, du Soleil, de l'atmosphère, de la pluie, de la scintillation ionosphérique et de l'arrière-plan cosmique (méthode proposée dans les recommandations ITU- RP 372 et 618 [148]. Ce calcul préliminaire a permis d'estimer la température du système de réception au sol équivalent à 1077 K [84].

5.7. Equations utilisées dans le bilan de liaison

5.7.1. Equation généralisée de FRIIS

La puissance reçue par une antenne réceptrice placée à une distance R est donnée par la relation de FRIIS :

$$P_r = \psi \cdot A_{eff} \quad (W) \quad (4.14)$$

Avec ψ la densité de flux du signal incident et A_{eff} la surface effective de couverture d'une antenne réceptrice. D'où :

$$P_r = P_t \cdot G_t \cdot G_r \cdot \left(\frac{\lambda}{4 \cdot \pi \cdot R} \right)^2 \quad (4.15)$$

Avec :

- $P_r - P_t$: Puissance reçue par l'antenne réceptrice et celle transmise par l'antenne émettrice.
- $G_t - G_r$: Gain de l'antenne émettrice et Gain de l'antenne réceptrice.
- λ : longueur d'onde.
- R : Distance entre l'antenne émettrice et celle réceptrice.

Pour calculer la puissance reçue tout en prenant en considération les conditions réelles de propagation, on introduit un terme L_{pertes} représentant les pertes. D'où l'équation générale de FRIIS qui modélise le lien de communication de bout en bout :

$$P_r = \frac{P_t \cdot G_t \cdot G_r}{L_{pertes}} \cdot \left(\frac{\lambda}{4 \cdot \pi \cdot R} \right)^2 \quad (4.16)$$

En pratique, on utilise plus souvent la forme logarithmique C de la puissance du signal reçu P_r qui s'écrit sous les formes :

$$C = 10 \cdot \log(P_t \cdot G_t) + 10 \cdot \log(G_r) + 20 \cdot \log\left(\frac{\lambda}{4 \cdot \pi \cdot R}\right) - 10 \cdot \log(L_{pertes}) \quad (4.17)$$

$$C = PIRE + 10 \cdot \log(G_r) - FSPL - 10 \cdot \log(L_{pertes}) \quad (4.18)$$

Avec :

- $PIRE = 10 \cdot \log(P_t \cdot G_t)$: Puissance Isotropique Rayonnée Equivalente
- $FSPL = 20 \cdot \log\left(\frac{4 \cdot \pi \cdot d}{\lambda}\right)$: Pertes à l'air libre

5.7.2. Le rapport signal/bruit

Le rapport signal/bruit (SNR) est le principal paramètre à calculer pour définir la performance globale des liaisons montantes et descendantes d'un système de communication satellitaire. Il est défini par [87] :

$$SNR = \frac{\text{Puissance du signal reçu}}{\text{Puissance du bruit}} = \frac{P_r(W)}{P_N(W)} = \frac{C(dbW)}{N(dbW)} \quad (4.19)$$

Avec :

- P_r et C : puissance du signal reçu.
- P_N et N : puissance du bruit.

5.7.3. La marge de liaison

La marge de liaison, mesurée en dB, constitue la différence entre le rapport SNR calculé par le bilan de liaison, et le rapport (Energie requise par bit / la densité du bruit) fixé par la qualité minimale demandée au niveau du taux d'erreurs par bit (BER) :

$$\text{Marge de liaison} = SNR - \left(\frac{E_b}{N_0} \right)_{BER \text{ requis}} \quad (4.20)$$

Avec E_b l'énergie requise par bit et N_0 la densité du bruit.

Parmi les contraintes de conception du sous-système COM (§ 2), le BER est fixé à 10^{-5} et la marge de liaison minimale à 6 dB pour tenir compte des conditions réelles de propagation et des pertes inattendues de conception.

La valeur du rapport E_b/N_0 dépend de la technique de modulation choisie et elle peut être trouvée à partir de la courbe de la *figure 4-14* :

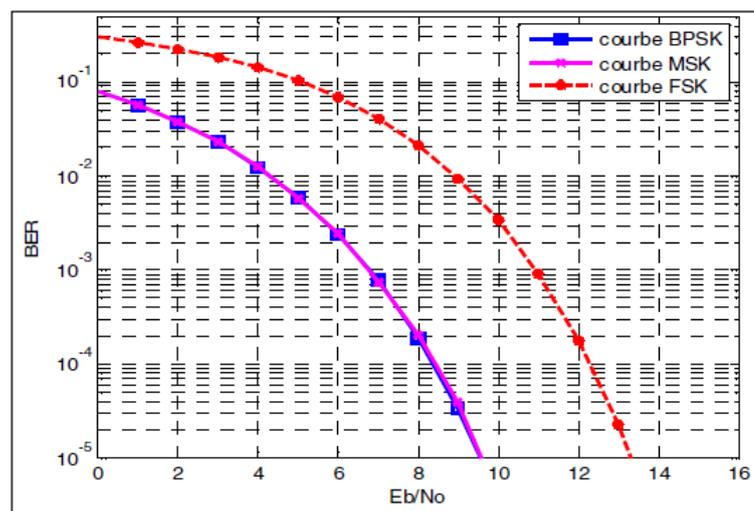


Figure 4-14 : Performances BER de différents techniques de modulation [84]

5.8. Bilan de la liaison montante (station sol vers segment spatial)

Angle d'élévation (°)	5	5
Altitude (Km)	650	650
Fréquence (MHz)	437	437
Station sol		
Puissance de transmission (W)	75	75
Puissance de transmission (dBW)	18.8	18.8
Pertes de lignes de transmission (dB)	2.7	2.7
Gain de l'antenne (dBi)	16	16
PIRE de la station (dB)	32.1	32.1
Canal		
Pertes de pointages de la station (dB)	1	1
Pertes de polarisation (dB)	3	3
Pertes à air libre (dB)	153	154
Pertes atmosphériques (dB)	0.5	0.5
Puissance Isotropique reçue (dBW)	-125.4	-126.4
Cubesat		
Pertes de pointages Cubesat (dB)	4.4	4.4
Gain de l'antenne dipôle (dBi)	2	2
Pertes de ligne du Cubesat (dB)	0.2	0.2
Température équivalente du Cubesat (K)	474	474
Facteur de mérite du Cubesat (G/T) (dB/K)	-24.9	-24.9
Puissance reçue à l'entrée du récepteur (dBW)	-128	-129
Débit binaire (bps)	1200	9600
Puissance du bruit N (dBW)	-171.1	-162
SNR (dBW)	43.1	33
Technique de modulation utilisée	FSK (G3RUH)	MSK
SNR requis pour un BER = 10 ⁻⁵	18	9.5
Marge de liaison (dB)	25.1	23.5

Tableau 4.2 : Bilan de la liaison montante [84]

5.9. Bilan de la liaison descendante (Segment spatial vers station sol)

Angle d'élévation (°)	5	5
Altitude (Km)	650	650
Fréquence (MHz)	435	435
Cubesat		
Puissance de transmission (W)	1	1
Puissance de transmission (dBW)	0	0
Pertes de lignes de transmission (dB)	0.2	0.2
Gain de l'antenne (dBi)	2	2
PIRE du Cubesat (dB)	1.8	1.8
Canal		
Pertes de pointages Cubesat (dB)	4.4	4.4
Pertes de polarisation (dB)	3	3
Pertes à air libre (dB)	153	154
Pertes atmosphériques (dB)	0.55	0.55
Puissance Isotropique reçue (dBW)	-159.15	-160.15
Station sol		
Pertes de pointages de la station (dB)	1	1
Gain de l'antenne Yagi (dBi)	16	16
Pertes de ligne de la station (dB)	2.7	2.7
Température équivalente de la station (K)	1077	1077
Facteur de mérite de la station (G/T) (dB/K)	-17.02	-18.02
Puissance reçue à l'entrée du récepteur (dBW)	-146.85	-147.85
Débit binaire (bps)	1200	9600
Puissance du bruit N (dBW)	-167.5	-158.5
SNR (dBW)	20.65	10.65
Technique de modulation utilisée	FSK (G3RUH)	MSK
SNR requis pour un BER = 10 ⁻⁵	18	9.5
Marge de liaison (dB)	2.65	1.15

Tableau 4-3 : Bilan de la liaison descendante

5.10. Interprétation des résultats

- Le SNR obtenu pour les techniques de modulation choisies (FSK et GMSK) peut être amélioré, en introduisant des techniques de codage et de correction d'erreur.
- Les marges de liaison calculées pour la liaison montante et descendante montrent qu'il est possible d'établir un lien de communication même dans les pires des cas d'altitude, d'angle d'élévation et de bruit thermique.

6. CHOIX DE LA SOLUTION TECHNOLOGIQUE

Le choix s'est basé sur l'architecture adoptée, ainsi que sur des solutions technologiques possédant un héritage de vol, c'est-à-dire qui ont déjà été utilisés et testés dans d'autres missions Cubesats. Le *tableau 4-6* résume les solutions éprouvées et qui semblent particulièrement adaptées à notre mission :

<i>Carte COM</i>	<i>Puissance Tx consommée</i>	<i>Modulation Débit max</i>	<i>Sensibilité Récepteur</i>	<i>AX25</i>	<i>Interfaces I2C- PC104</i>
ISIS TRXVU	< 1,7 W	AFSK 9600 bps	-100 dbm	X	X - X
CLYDE SPACE UTRX	4 W - 10 W	GMSK	-120 dbm	X	X - X
ASTRODEV Li-1	< 10 W	FSK/GMSK 9600 bps	---	X	/
GOMSPACE NanoCom U482C	< 5,5 W	MSK	-126 dbm	/	/
GOMSPACE NanoCom AX100	< 2,64 W	FSK/MSK/GFSK/ GMSK 115.2 kbps	-137 dbm	X	X - /

Tableau 4-4 : Caractéristiques des modules COM des différents fournisseurs

Le choix s'est ainsi porté sur le module GomSpace NanoCom AX100 qui satisfait la plupart des exigences et se démarque des autres modules par :

- Une consommation d'énergie en mode transmission faible.
- Un débit maximal largement supérieur aux autres solutions.
- Une très grande flexibilité au niveau du paramétrage et de la reconfiguration du module : type de modulation, fréquences d'émission et de transmission, débit, intervalle balise, etc.
- Compatibilité avec les solutions GOMSpace adoptées pour les sous-systèmes EdB et OdB.



Figure 4-15 : Module GOMSpace NanoCom AX100 [89]

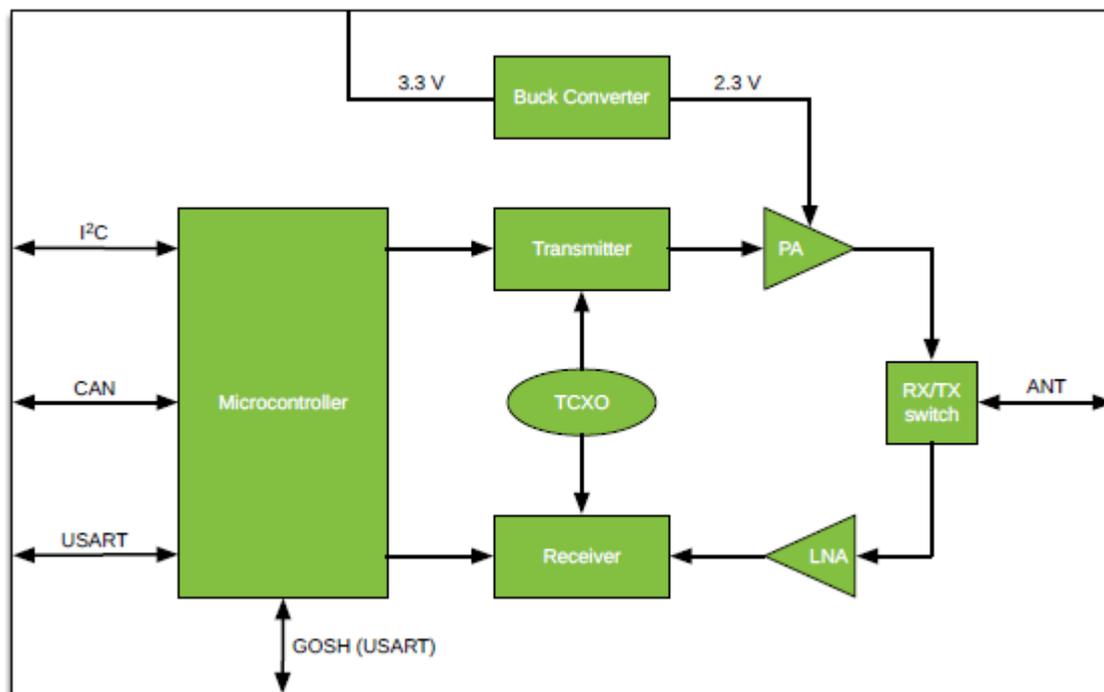
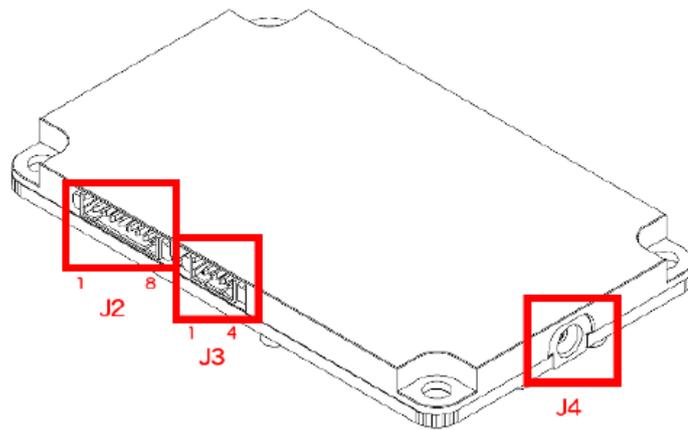


Figure 4-16 : Diagramme fonctionnel du module NanoCom AX100 [89]

Le module NanoCom AX100 est un émetteur-récepteur radio half duplex, dont les paramètres RF sont totalement configurables au sol et sur orbite. Il possède, entre autres, les fonctionnalités et caractéristiques suivantes :

- Fréquence reconfigurable dans la plage UHF : 430-440 MHz
- Débit binaire reconfigurable entre 0.1 kbps et 115.2 kbps
- Type de modulation : FSK/MSK/GFSK/GMSK
- Transmetteur avec une puissance de sortie réglable de 24 à 30 dBm
- Type d'encapsulation de la trame :
 - AX25 - HDLC – En-tête 32 bit ASM – codage viterbi ($r \frac{1}{2} k=7$)
- 32 kB de mémoire FRAM (Ferroelectric RAM) pour le stockage de la configuration courante.
- Système de protection ajustable contre la surchauffe de l'amplificateur de puissance
- Interface de communication I2C.

Pour l'interfaçage, la carte est équipée des connecteurs suivants :



- **J2** : Interface JTAG pour téléchargement du logiciel d'usine uniquement.
- **J3** : Interface série USART pour débogage
- **J4** : Connecteur RF MCX 50Ω pour l'antenne UHF

Figure 4-17 : Connecteurs de la carte NanoCom AX100 [89]

Cependant, l'adaptation de ce module au bus PC-104 de la plateforme nécessite l'utilisation d'une carte mère NanoDock DMC-3 qui sera décrite au chapitre 5.

Chapitre 5

ETUDE ET DESCRIPTION DES SOUS-SYSTEMES ORDINATEUR DE BORD, STRUCTURE ET CONTROLE D'ATTITUDE

1. ETUDE ET DESCRIPTION DU SOUS-SYSTEME ORDINATEUR DE BORD

Dans la plateforme d'un nanosatellite, l'OdB constitue le système principal de traitement des données et des télécommandes. Il doit garantir un état opérationnel stable du segment spatial en mode de fonctionnement nominal ; tout en assurant toute les tâches de supervision et de contrôle du nanosatellite. L'OdB doit ainsi gère trois types de flux de données numériques distincts : les données de télémétrie (TM) et de débogage, les données de la charge utile et les télécommandes (TC) provenant de la station au sol.

Plus précisément, les tâches que l'OdB doit réaliser sont les suivantes :

- Fournir la base de temps du segment spatial.
- Fournir le signal de commande pour le déploiement des antennes.
- Contrôler le bus de communication I2C (en mode multi-maitre) pour assurer l'échange de données et l'interfaçage entre les sous-systèmes du segment spatial.
- Superviser le fonctionnement de tous les sous-systèmes du segment spatial en recevant des informations de débogage de chacun d'eux.
- Superviser et stocker les mesures de télémétrie réalisées à bord du segment spatial entre les séquences de communication avec la station au sol.
- Répondre, pendant les séquences de communication, aux demandes de télémétrie provenant de la station au sol.
- Gérer les TC qui proviennent de la station sol et stocker celles à exécution différée.
- Gérer l'énergie électrique (mode nominal) en activant/désactivant les sous-systèmes selon les scénarios opérationnels et les conditions prédéfinies pour la disponibilité de l'énergie électrique.

1.1. Exigences du sous-système OdB pour la mission MASAT1

[DES] FUNC-0600

L'OdB sera en charge de la supervision globale, la commande et le contrôle de toutes les opérations du nanosatellite et de la charge utile.

[DES] FUNC-0610

L'OdB doit fournir toutes les fonctionnalités nécessaires pour l'acquisition, le stockage, le formatage et la transmission des informations de TM vers le segment au sol.

[DES] FUNC-0620

L'OdB doit fournir toutes les fonctionnalités nécessaires pour la validation, le décodage et le traitement des TC qui peuvent être à exécution directe ou reportée.

[DES] FUNC-0630

L'OdB doit être en charge de la détection, l'isolation et la récupération des défaillances (DIRD) de tous les sous-systèmes et composants qui ne sont pas pourvus de DIRD autonome.

[DES] FUNC-0640

L'OdB doit fournir la possibilité, le cas échéant, de transférer des données entre les sous-systèmes du nanosatellite et les instruments de mesure.

[DES] OPER-0650

Un profil des opérations doit être défini pour couvrir toutes les phases de la mission à savoir du lancement jusqu'à la fin du cycle de vie.

[DES] OPER-0660

L'OdB doit fonctionner en mode autonome avec une intervention minimale du segment au sol, y compris pour sa propre initialisation.

[DES] OPER-0670

L'autonomie des opérations de l'OdB doit être assurée par les logiciels embarqués et non par les commandes stockées en mémoire.

[DES] OPER-0680

L'OdB doit considérer les TC provenant de la station sol comme prioritaires par rapport aux fonctions d'autonomie des logiciels embarqués

[DES] OPER-0700

L'OdB doit être capable d'authentifier la source de toutes les TC reçues.

[DES] OPER-0710

L'OdB ne doit exécuter que les TC dont la source a été authentifiée.

[DES] OPER-0720

L'OdB doit prévoir un contournement de l'authentification des télécommandes.

[DES] OPER-0730

L'OdB doit rejeter les TC invalides ou mal construites pour éviter l'exécution d'actions entraînant des conséquences catastrophiques, critiques ou majeurs.

[DES] OPER-0740

L'OdB doit pouvoir stocker les TC à exécution reportée et les exécuter en fonction de leur temps absolu.

[DES] OPER-0750

L'OdB doit exécuter les TC à exécution directe avant les TC stockées en mémoire.

[DES] OPER-0760

L'OdB doit pouvoir activer, désactiver ou annuler une séquence de commandes (prédéfinies ou dynamiques) stockées en mémoire via le segment au sol.

[DES] OPER-0770

L'OdB doit rapporter toute reconfiguration matérielle ou réinitialisation autonome dans les données de TM.

[DES] OPER-0790

L'OdB doit signaler l'identification, l'acceptation ou le rejet d'une TC dans les données de TM.

[DES] OPER-0810

L'OdB doit être capable de changer la cadence concernant l'acquisition des données de TM.

[DES] OPER-0820

Le segment spatial doit être capable de modifier le contenu des données TM.

[DES] CONC-0920

La conception de l'OdB doit se faire à base de composants COTS (Commercial Off The Shelf) et fournir un degré élevé de tolérance de pannes.

[DES] CONC-0950

L'OdB doit produire la base de temps à bord du segment spatial et distribuer les signaux d'horloge et de synchronisation aux différents sous-systèmes et composants du nanosatellite.

[DES] CONC-0960

Une marge de 20% (à confirmer) pour la puissance de calcul, la mémoire de masse et le débit des liaisons doit être appliquée.

[DES] CONC-0970

La conception de l'OdB doit prévoir la dégradation progressive de la capacité de stockage des données Cette dernière doit être conçue avec une marge de 10% à la fin de cycle de vie.

[DES] CONC-0980

L'OdB doit fournir suffisamment de capacité mémoire pour le stockage des données générées à bord et qui peuvent concerner la TM, le traitement embarqué et les instruments de mesures.

[DES] CONC-0990

Le logiciel de vol embarqué et exécutant les procédures opérationnelles vitales, y compris la procédure d'initialisation, doit être stocké dans une mémoire non volatile de telle sorte qu'une configuration par défaut soit toujours disponible en cas d'anomalies.

[DES] CONC-1000

Les logiciels embarqués doivent permettre le contrôle en temps réel.

[DES] CONC-1020

Le logiciel de vol doit être structuré de façon modulaire en utilisant un langage de haut niveau.

[DES] CONC-1030

La modification ou la mise à jour des logiciels embarqués doit être possible en orbite.

[DES] CONC-1040

La conception de l'OdB doit réduire au maximum le nombre d'interface.

[DES] CONC-1050

L'architecture modulaire du logiciel de vol doit couvrir le contrôle de plusieurs niveaux d'entrées/sorties :

- *Contrôleur mémoire et protocoles de communication interne.*
- *Surveillance de l'attitude du satellite.*
- *Commande et supervision du sous-système Edb.*
- *Contrôle de la charge utile ;*
- *Routines de DIRD ;*

[DES] CONC-1070

L'OdB doit implémenter un système matériel de surveillance indépendant du processeur (comme un chien de garde) pour interrompre chaque tâche ou fonction embarquée bloquée ou tournant en boucle infinie.

[DES] FMST-1090

Aucun mode de fonctionnement critique ne doit être activé si la perte de communication avec le segment sol ne dépasse pas 24 heures (à confirmer). Cela inclut la gestion de simples défaillances à bord.

[DES] FMST-1120

Éviter la défaillance doit être préférée à la tolérance aux pannes chaque fois que possible.

[DES] FMST-1130

Aucun point de défaillance ne peut être autorisé dans les fonctions vitales de l'OdB. Cela doit être démontré par un outil de sûreté de fonctionnement et de gestion de la qualité comme l'Analyse des modes de défaillance, de leurs effets et de leur criticité (AMDEC).

1.2. Spécifications matérielles d'un OdB

1.2.1. Puissance de calcul

La puissance de calcul est un critère nécessaire pour dimensionner le processeur de l'OdB. Sa détermination nécessite le calcul de [8] :

- La taille du logiciel embarqué mesurée par le nombre de mots en mémoire. Elle dépendra des tâches logicielles qui seront effectuées par l'OdB et du choix de l'utilisation ou non d'un système d'exploitation temps réel. Il est à noter que pour estimer le coût du logiciel embarqué, on utilise généralement le nombre de lignes du code source (SLOC : Source Line Of Code). La relation entre SLOC et le nombre de mots en mémoire dépend alors du compilateur utilisé. Ainsi, pour le langage C, il y aura 7 instructions assembleur par SLOC et 42 octets par SLOC pour un processeur 32 bits.

- Le temps de traitement mesuré par le calcul du nombre d'instructions par seconde (KIPS : Kilo of Instructions Per Second ou MIPS : Millions of Instructions Per Second). Ce calcul dépendra du jeu d'instruction et de la vitesse de l'horloge. Si par exemple, le programme contient deux types d'instructions, 60% d'instructions type A nécessitant 2 cycles d'horloge et 40% d'instructions type B nécessitant 7 cycles, le temps de traitement avec une horloge de 10 MHz est égal à :

$$T_{\text{traitement}} = \frac{T_H}{(0,6 * 2 + 0,4 * 7)} = 2,5 \text{ MIPS} \quad (5.1)$$

Pour déterminer la configuration matérielle requise, la règle de base est de multiplier par quatre les estimations des calculs concernant la taille des logiciels embarqués et le temps de calcul [8]. Cette majoration permettra de compenser les changements et mise à jour réalisés pendant les phases tardives du développement du segment spatial.

Le **tableau 5-1** donne une estimation concernant la taille du logiciel (en termes de mémoire) et le temps de traitement concernant les tâches communes pour un logiciel embarqué. Les valeurs sont basées sur une architecture RISC MIL-STD-1750A avec un jeu d'instructions 16 bits et un logiciel développé dans un langage haut niveau.

Fonction	Taille (K mots*)		Temps traitement typique (KIPS)	Fréquence d'exécution typique (Hz)
	Code	Données		
Communication				
▪ Traitement commandes	1.0	4.0	7.0	10.0
▪ Traitement télémétrie	1.0	2.5	3.0	10.0
Autonomie				
▪ Autonomie simple	2.0	1.0	1.0	1.0
▪ Autonomie complexe	15.0	10.0	20.0	10.0
Détection d'erreurs				
▪ Contrôle	4.0	1.0	15.0	5.0
▪ Correction	2.0	10.0	5.0	5.0
Autres fonctions				
▪ Gestion de l'énergie	1.2	0.5	5.0	1.0
▪ Contrôle thermique	0.8	1.5	3.0	0.1

* 2K mots de mémoire = 2048 mots = 2,048 x 16 bits = 32,768 bits = 32 Kbits

Tableau 5-1 : Estimation de la taille du logiciel embarqué et du temps de traitement [8]

1.2.2. Consommation d'énergie

La puissance disponible à bord du segment spatial est extrêmement limitée (typiquement de l'ordre de 1W à 3W pour tous les sous-systèmes, y compris pour les émetteurs radio). Il est donc très important de prendre la consommation d'énergie en compte lors de la sélection des composants ou des protocoles de communication.

Sachant que le sous-système OdB doit rester sous tension aussi longtemps que possible pendant toute la durée de vie de la mission, il est important qu'il consomme le moins d'énergie possible. D'un autre côté, puisque la tension d'alimentation sera située entre 3.3V et 5V, tous les composants utilisés doivent être capables de fonctionner dans cette plage de tension.

1.2.3. Génération de la référence temporelle

L'OdB doit générer une référence temporelle qui doit être compatible avec le format : année, mois, jour, heure, minutes et secondes. Deux approches peuvent alors être utilisées :

- Module "Horloge logiciel" qui utilise un temporisateur 32 bits pour maintenir une référence temporelle avec une précision d'une seconde, et la rendre disponible pour tous les autres sous-systèmes. Le module sera implémenté directement dans la routine d'interruption du temporisateur dont l'horloge utilisera un quartz de 32768 KHz. Cela donnera, après configuration, l'exécution périodique de 1Hz d'un code d'interruption qui consistera simplement à incrémenter un compteur. On aura une capacité de comptage allant jusqu'à 2^{32} secondes soit environ 136 ans.
- Horloge temps réel externe (RTC : Real Time Clock) commandée par un quartz de 32768 KHz et permettant de fournir un "tick" contrôlant la fréquence d'exécution d'une interruption RTC. Pendant cette interruption RTC, la référence temporelle du segment spatial pourra être mise à jour via des compteurs 32 bits.

1.2.4. Mémoire de stockage

Il est nécessaire d'avoir un moyen de stockage des données à bord du segment spatial. Cette mémoire sera utilisée uniquement pour stocker les mesures de télémétrie, et les informations de débogage sur le comportement des différents sous-systèmes. Par conséquent, il n'y a pas besoin d'une grande capacité de stockage ou d'un taux de transfert important.

Pour les mesures de télémétrie, l'allocation mémoire peut se faire :

- *Statiquement*, s'il est prévu à l'avance dans le logiciel embarqué d'exécuter périodiquement une série de mesures.
- *Dynamiquement*, si l'on envoie depuis la station sol une commande au segment spatial pour lui faire effectuer une série de mesures choisies.

La taille de la mémoire occupée par les mesures de télémétrie dépendra alors de :

- La place occupée par une mesure (son format de stockage).
- La fréquence à laquelle on désire connaître la mesure
- La durée pendant laquelle on désire conserver la mesure.

Le **tableau 5-2** donne une première estimation de l'allocation mémoire statique nécessaire pour les données de télémétrie de la mission MASAT1 sachant que :

- La durée de conservation des données est de 24 h (à confirmer).
- Données de télémétrie codée sur 16 bits.
- La sauvegarde de l'heure temporelle du segment spatial (32bits) est prévue.

Mesures	Variable/taille	Cadence	Nombre d'octets	Nombre d'octets pour l'heure	Total
Tensions entrées Convertisseurs PV	vboost[3] 16 bits	1 fois toutes les 10 minutes	864	(Toutes les 20 min) 288	1152
Courants entrées Convertisseurs PV	curin[3] 16 bits	1 fois toutes les 10 minutes	864	(Toutes les 20 min) 288	1152
Courant sortie Convertisseur PV	cursum 16 bits	1 fois toutes les 20 s	8640	(Toutes les 2 min) 2880	11520
Tension sortie batterie Courant sortie batterie	Vbat cursys 16bits	1 fois toutes les 20 s	17280	(Toutes les 2 min) 2880	20160
Courants sorties Canaux d'alimentation Etat sorties Canaux d'alimentation	curout[4] 16 bits outout[6] 8 bits	1 fois toutes les 10 minutes	2016	(Toutes les 20 min) 288	2304
Températures Convertisseurs PV Températures Batteries	temp[4] 16 bits	1 fois toutes les 5 minutes	2304	(Toutes les 10 min) 576	2880
Mode fonctionnement batterie Mode suivi PPI	batt_mod ppt_mode 8 bits	1 fois toutes les 10 minutes	576	(Toutes les 20 min) 288	864
Nombre de redémarrage EdB Nombre de redémarrage COM	count_boot_E count_boot_C 16 bits	1 fois toutes les 5 minutes	1152	(Toutes les 10 min) 576	1720
Cause redémarrage EdB Cause redémarrage COM	boot_cause_E boot_cause_C 32 bits	1 redémarrage / 24 heures (Estimation)	8	8	16
Températures panneaux PV	temp_PV[5] 16 bits	1 fois toutes les 2 minutes	7200	(Toutes les 2 min) 2880	10080
Température Module OdB Température Module COM Température Ampli COM	temp_brd_O temp_brd_C temp_pa 16 bits	1 fois toutes les 2 minutes	4320	(Toutes les 2 min) 2880	7200
Nombre de paquets reçu et transmis par la COM depuis dernier redémarrage	tx_count rx_count 32 bits	10 fois toutes les 24 heures	80	(10 fois toutes les 24 heures) 80	160
Configuration active du module COM	active_conf 8 bits	1 fois toutes les 10 minutes	144	(Toutes les 20 min) 288	432
Total					59640

Tableau 5-2 : Estimation de la taille mémoire statique pour les données de TM

1.3. Architecture des OdBs utilisés dans d'autres missions Cubesat

Les critères qui influencent le choix de la solution technologique pour le sous-système OdB sont généralement le type de microcontrôleur, l'architecture et le bus de communication. Les *figures 5-1, 5-2* et *5-3* résument les résultats du benchmark fonctionnel réalisé en *ANNEXE C* et qui concerne 39 missions de nanosatellites lancés depuis l'année 2003 :

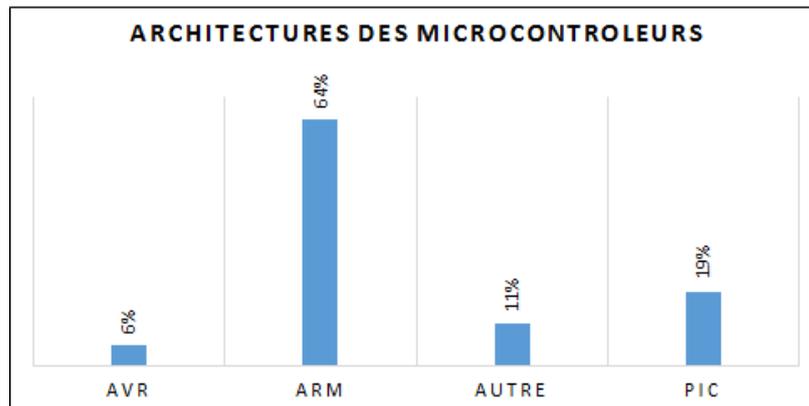


Figure 5-1 : Architectures des microcontrôleurs utilisées dans les missions Cubesat

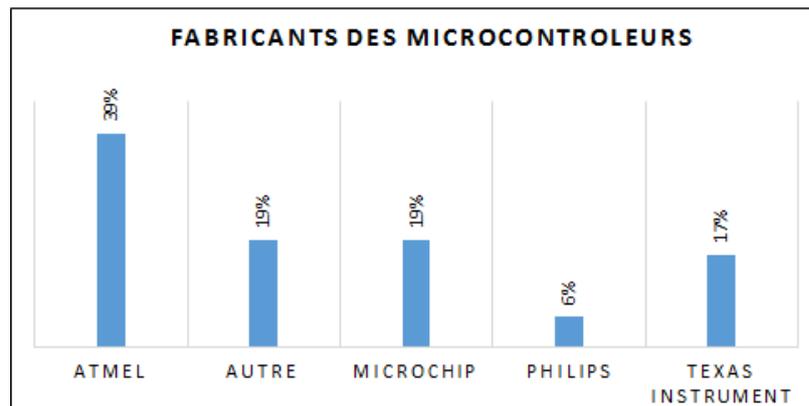


Figure 5-2 : Fabricants des microcontrôleurs utilisées dans les missions Cubesat

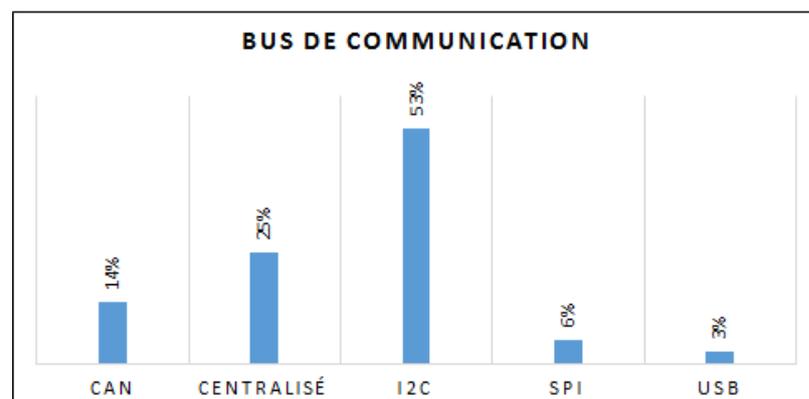


Figure 5-3 : Bus de communication utilisées dans les missions Cubesat

Les conclusions de cette étude statistiques sont les suivantes :

- L'architecture ARM est la plus adoptée comparaison à l'architecture PIC.
- Les sociétés ATMEL, MICROSHIP et TEXAS INSTRUMENT sont celles qui ont obtenu la confiance de la communauté CubeSat, avec un avantage pour ATMEL.
- La plupart des missions ont adoptées une architecture distribuée avec principalement le bus I2C comme bus de communication série.
- Il y a eu 2 migrations vers l'architecture ARM :
 - Les missions AAUSAT qui ont migré de l'architecture Siemens (AAUSAT-1) vers l'architecture ARM (AAUSAAT-2 et AAUSAT-3) ;
 - Les missions PolySat qui ont migrés de l'architecture PIC (PolySat CP1 et CP2) vers l'architecture ARM (PolySat CP3 et CP4).

Ces constatations favorisent l'utilisation d'une architecture ARM au détriment de l'architecture PIC, et d'un bus de communication série I2C au détriment du bus CAN.

1.4. Interfaces électriques

Les interfaces électriques du sous-système OdB sont résumées par la *figure 5-4*. On distingue :

- Des entrées analogiques pour l'acquisition des mesures de télémétrie.
- Les signaux DEPLOI_ANT1 et DEPLOI_ANT2 pour la commande du système de déploiement des antennes.
- L'entrée d'alimentation *Vcc* provenant du sous-système EdB.
- Les interfaces SPI et I2C (bus de communication), et les interfaces JTAG et série pour respectivement la configuration et le débogage.

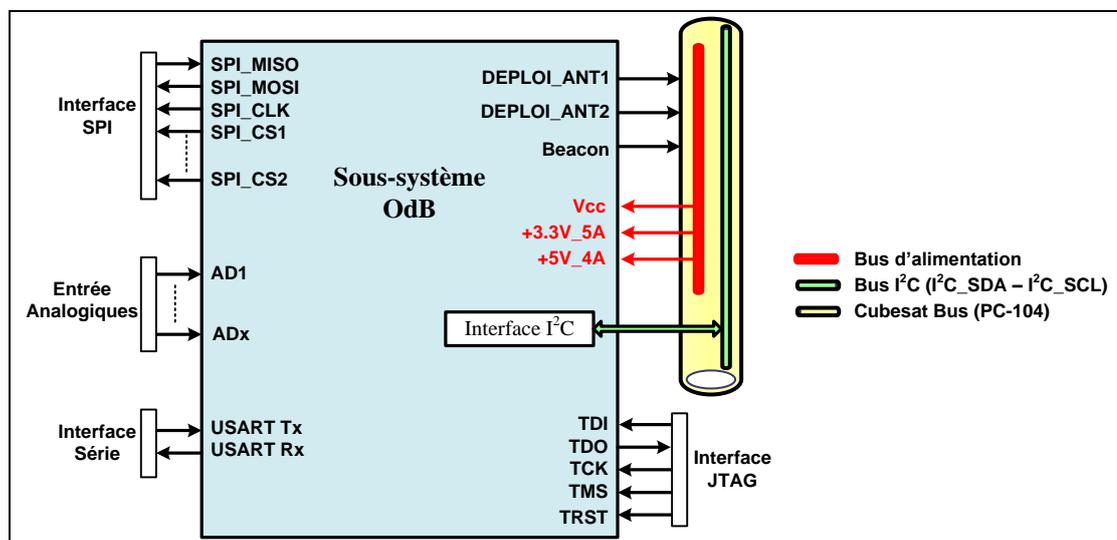


Figure 5-4 : Interfaces électriques du sous-système OdB

1.5. Architecture du sous-système OdB de la mission MASAT1

Conformément aux exigences de notre OdB, aux conclusions du benchmark fonctionnel réalisé et aux définitions des solutions de conception de MASAT1 (Chapitre 2 § 3.1 et § 3.2), nous avons dressé l'architecture fonctionnelle de *figure 5-5* :

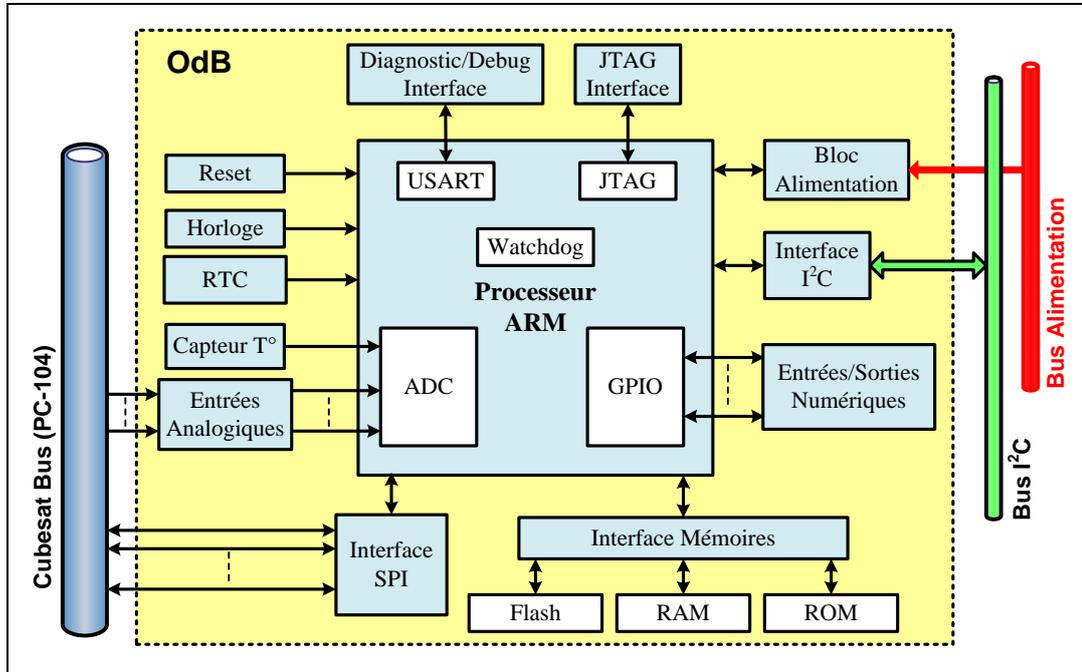


Figure 5-5 : Schéma bloc fonctionnel de l'OdB de MASAT1

Le schéma bloc fonctionnel prévoit l'utilisation d'un chien de garde logiciel pour la remise à zéro du système en cas de dysfonctionnement software. Ce dernier peut aussi être réalisé matériellement via un circuit externe au processeur.

On distingue aussi l'utilisation de trois types de mémoire :

- **ROM/Flash** : Mémoire résistante aux radiations qui joue le rôle de mémoire de masse (disque dur) pour le stockage du système d'exploitation et des logiciels embarqués. Elle contiendra ainsi le "bootloader" qui sera lancé à chaque remise à zéro ou redémarrage de l'OdB.
- **Flash** : Mémoire pour la sauvegarde des données applicatives permanentes et de l'historique concernant les données de télémétrie.
- **SRAM** : Mémoire vive de travail de l'OdB permettant le stockage des données temporaires applicatives ainsi que les valeurs temporaires de télémétrie. Le choix d'une SRAM à la place d'une SDRAM a été dicté par les avantages qu'elle offre à savoir : pas de rafraîchissement, mémoire à accès direct, consommation d'énergie faible et stabilité. Comme les mémoires RAM sont sensibles aux effets singuliers dues aux radiations, on doit prévoir une détection/correction d'erreur EDAC (Error Detection And Correction).

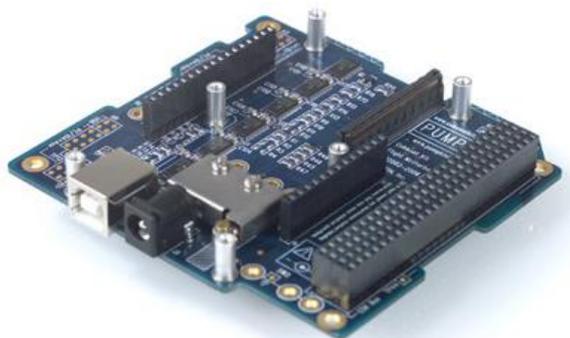
Une horloge temps réel (RTC) a été prévue pour fournir une référence temporelle permettant la synchronisation et la planification de tâches. Mais si le processeur choisi dispose de suffisamment de temporisateurs, la référence temporelle peut être générée par logiciel comme c'est décrit dans § 1.2.3.

1.6. Choix de la solution technologique

Le choix s'est basé sur l'architecture adoptée, ainsi que sur des solutions technologiques possédant un héritage de vol, c'est-à-dire qui ont déjà été utilisés et testés dans d'autres missions Cubesats. Parmi les Ordinateurs de Bords éprouvés, trois solutions semblent particulièrement adaptées à notre mission :

1.6.1. Carte mère CubeSat Kit de Pumpkin avec processeur MSP430F2618 :

C'est une solution basée sur le microcontrôleur MSP430 (16-Bit ARM RISC CPU) de Texas Instrument. C'est un processeur ARM RISC à faible consommation avec une fréquence d'horloge pouvant atteindre 25MHz. Son architecture est compatible avec l'architecture de Von Neumann, c'est-à-dire le programme et les données sont stockés dans la même mémoire (auto-programmation possible). La version actuelle du Kit Pumpkin utilise le processeur MSP430F2618.



(a) Carte mère



(b) Processeur MSP430 enfilable

Figure 5-6 : Cubesat Kit MSP430 de la société Pumpkin [90]

L'étude de cette solution [90] a démontré qu'il existe deux inconvénients concernant son utilisation et son interfaçage avec la solution matérielle adoptée pour le sous-système EdB :

- Le kit MSP430 de Pumpkin n'offre pas de mémoires externes autres qu'une mémoire carte SD qui est gourmande en énergie et qui a une capacité qui dépasse largement nos besoins.
- Le kit MSP430 ne possède pas d'interfaces SPI et il n'est pas compatible avec le système d'exploitation temps réel FreeRTOS.

- L'interfaçage avec la carte adoptée pour le sous-système Edb (NanoPower P31U de GOMSpace) n'est pas pleinement compatible au niveau de l'alimentation. En effet, le Kit MSP430 devra être alimenté via l'alimentation permanente 3,3V @ 5A (connecteurs H2.27 et H2.28) de l'EdB et non via l'un des 6 canaux d'alimentation de la carte EdB, qui sont dédiés à l'alimentation des sous-systèmes du segment spatial et qui ont l'avantage d'être configurable, contrôlable via le bus I2C et protégés contre les effets "latch-up".

1.6.2. Module NanoMind A712D de GOMSpace

C'est une solution basée sur le microcontrôleur ATMEL AT91SAM7A1 (32-Bits ARM7TDMI RISC CPU) qui est un processeur RISC à faible consommation, pouvant fonctionner avec une fréquence d'horloge allant de 8MHz à 40MHz.



Figure 5-7 : Carte NanoMind A712D de GOMSpace [91]

L'étude de cette solution [91] a démontré qu'elle est adaptée avec les objectifs de la mission et avec la solution matérielle adoptée pour le sous-système EdB :

- La cartographie mémoire du système NanoMind A712D offre plus de flexibilité et de possibilités avec une mémoire RAM externe, une première mémoire de masse (Flash1) pour le stockage des données et une deuxième mémoire de masse (Flash2) pour le logiciel embarqué.
- L'interfaçage avec la carte adoptée pour le sous-système Edb (NanoPower P31U de GOMSpace) est compatible et peut utiliser n'importe lequel des 6 canaux dédiés à l'alimentation des sous-systèmes.

- La carte NanoMind A712D possède des interfaces SPI dédiées à la communication avec les capteurs de température des cellules solaires des 6 faces.
- La documentation concernant la carte NanoMind A712D ne spécifie pas l'affectation des broches H2.45 et H2.46, sachant que la carte EdB donne un accès direct (à travers ces mêmes broches) à la tension des batteries V_{BAT} .

1.6.3. Module NanoMind A3200 de GOMSpace

C'est une solution basée sur le microcontrôleur Atmel AT32UC3C512C MCU avec fonctions avancées d'économie d'énergie et fonctionnant avec une fréquence d'horloge allant de 8MHz à 40MHz :



Figure 5-8 : Carte NanoMind A3200 de GOMSpace [92]

Le microcontrôleur Atmel AT32UC3C512C est basée le nouvelle architecture AVR32 RISC 32 bits, conçue pour les applications embarquées sensibles à la consommation d'énergie. Il offre une variété de microarchitectures, permettant au processeur d'être mis en œuvre sous la forme de processeur à faible, moyenne ou haute performance.

Ce processeur est caractérisé, entre autres, par :

- Puissance de calcul de 1,49 DMIPS/MHz (Dhrystone MIPS)
- Plage de température : -40°C à +85°C.
- Consommation en courant très faible :
 - Active mode : 512 μ A/MHz
 - Idle mode : 258 μ A/MHz
 - Frozen mode : 106 μ A/MHz
 - Standby Mode : 48 μ A/MHz
- 64 KO de mémoire SRAM haute vitesse interne et 512 KO de mémoire Flash haute vitesse

- Contrôleur de vecteurs d'interruption à faible latence avec priorité programmable.
- Un convertisseur analogique-digital 12 bits à 16 canaux.
- Chien de garde (Watchdog) programmable et six temporisateurs 16-bits
- 4 canaux 20 bits pour modulation de largeur d'impulsion (PWM)
- 1 Contrôleur de bus CAN 2 canaux (CAN 2.0A et 2.0B)
- Cinq interfaces de communication série (USART)
- 2 Interfaces de communication série SPI synchrone
- 2 convertisseurs digital-analogiques 12 bits, avec système d'échantillonnage à double sortie
- 3 interfaces I2C maître/esclave 400KHz
- 4 comparateurs analogiques

Concernant les fonctionnalités de la carte NanoMind A3200, elles sont résumées par le schéma fonctionnel de la *figure 5-9*. On distingue [92] :

- Large plage de fréquence de fonctionnement (via PLL intégrée).
- Horloge temps réel RTC.
- Capteurs de température embarqués.
- 4 types de mémoires externes : 32MO SDRAM – 128 MO Flash – 32kO FRAM
- Interfaces séries compatibles avec le protocole de communication CSP (CubeSat Space Protocol) : I2C – CAN – USART.
- Interface SPI externe avec 3 broches de sélection.
- Système de remise à zéro au démarrage et de supervision d'alimentation.
- Système de commande pour le contrôle d'attitude.
- Capteur magnéto-résistif à 3 axes.
- Gyroscope à 3 axes.
- 3 sorties PWM bidirectionnelles avec mesure de courant.
- Compatibilité avec le système d'exploitation FreeRTOS. En effet, une plateforme logicielle FreeRTOS, est fournie pour permettre un démarrage rapide du développement du logiciel embarqué. Cette plateforme, basée sur Eclipse, comprend une bibliothèque avec les pilotes de périphériques, et un outil de débogage et de programmation.

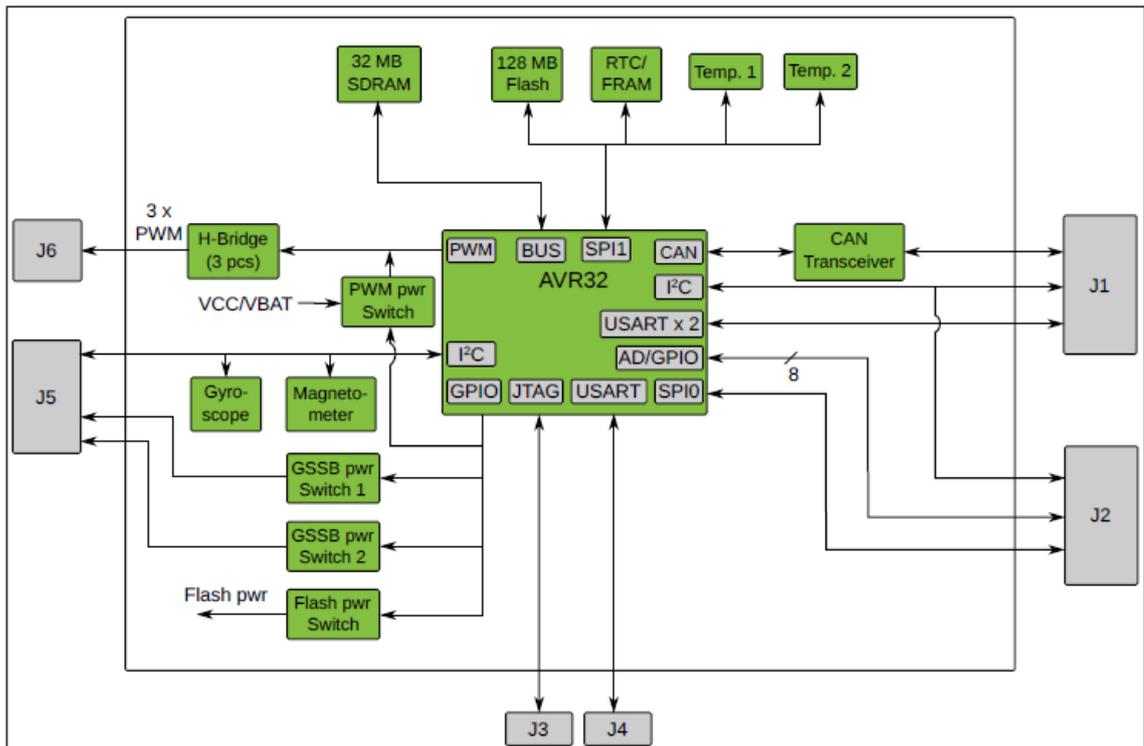
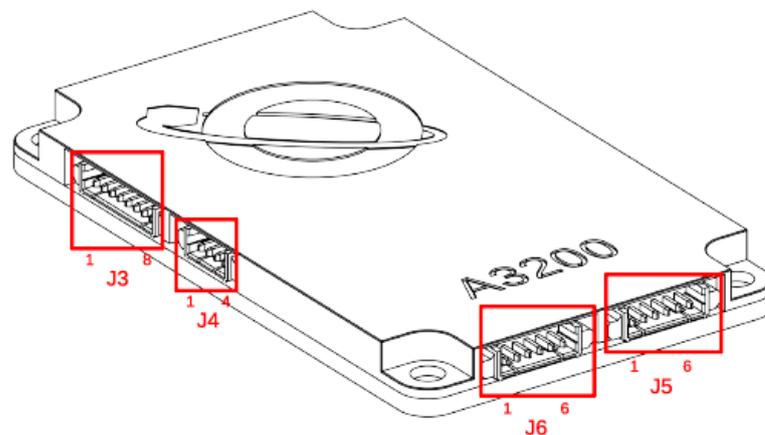


Figure 5-9 : Schéma bloc fonctionnel de la Carte NanoMind A3200 de GOMSpace [92]

Pour l'interfaçage, la carte NanoMind A3200 est équipée des connecteurs représentés par la *figure 5-10* :



- **J3** : Interface de programmation JTAG
- **J4** : Interface série USART de débogage
- **J5** : Connecteur I2C et VBAT permettant la communication avec le gyroscope le magnétomètre, ou tout autre dispositif externe.
- **J6** : Connecteur pour les sorties à modulation de largeur d'impulsion (PWM)

Figure 5-10 : Connecteurs de la carte NanoMind A3200 de GOMSpace [92]

L'étude de cette solution a démontré qu'elle est plus adaptée avec les objectifs de la mission et avec la solution matérielle adoptée pour le sous-système EdB :

- La cartographie mémoire du système NanoMind A3200 offre une plus grande capacité avec plus de flexibilité et de possibilités.

- L'interfaçage avec la carte Edb (GS-DS-NANOPOWER-P31U de GOMSpace) est compatible et peut utiliser n'importe lequel des 6 canaux dédiés à l'alimentation des sous-systèmes.
- La présence d'une interface SPI non dédiée, en plus de 8 entrées analogiques qui peuvent fonctionner comme E/S digitales.
- Une consommation d'énergie moyenne plus faible que la carte NanoMind A712D, avec une puissance de calcul plus importante.
- Un module de contrôle d'attitude actif qui ouvre plus d'opportunités pour les expériences scientifiques à effectuer en orbite.

Cependant, l'adaptation de ce module au bus PC-104 du segment spatial nécessite l'utilisation de la carte mère NanoDock DMC-3 pour réaliser l'interfaçage. C'est la même carte utilisée par le module NanoCom AX100.

2. DESCRIPTION DE LA CARTE MERE NANODOCK DMC-3 DE GOMSPACE

2.1. Présentation

La carte mère NanoDock DMC-3 est conçue pour permettre d'interfacer jusqu'à quatre cartes filles GOMSpace avec le bus PC104, tout en fournissant le montage et l'interfaçage d'un récepteur GPS. Chaque connecteur de carte fille possède des interfaces de communication (I2C et CAN) et des lignes d'alimentation configurables acheminées vers le connecteur d'empilement PC104, et permet ainsi à la carte mère d'héberger efficacement quatre sous-systèmes.

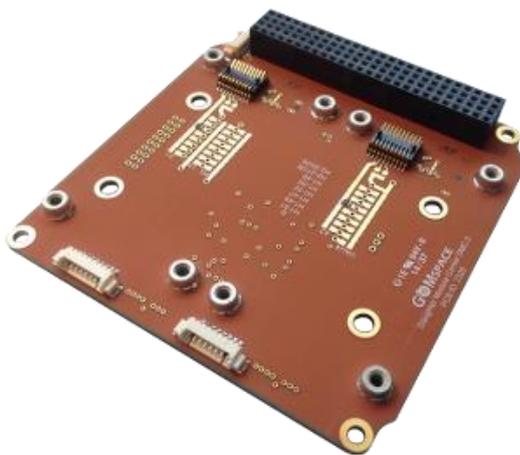


Figure 5-11 : Carte mère NanoDock DMC-3 de GOMSpace [93]

Pour faciliter l'accès au débogage des cartes filles, une interface USB peut être montée sur la carte pour permettre d'accéder à l'interface UART0 de chaque carte fille via USB.

2.2. Fonctionnalités matérielles

Comme le montre les *figures 5-12 et 5-13*, la carte mère NanoDock DC-3 est un circuit passif qui fournit une plate-forme physique pour les cartes-filles GOMSpace et les connexions électriques nécessaires avec le bus PC-104 du segment spatial. On distingue [93] :

- Les connecteurs X1 à X4 type SAMTEC-FSI pour cartes empilables, permettant l'alimentation et la communication avec les cartes filles.
- Les connecteurs H1/H2 type SAMTEC-SSQ permettant aux cartes filles d'accéder au bus PC-104 de la plateforme du segment spatial.
- Les connecteurs P1 à P10 permettant d'accéder aux entrées/sorties auxiliaires des cartes mères et à leurs interfaces UART.
- L'adaptateur USB/Serial qui est le seul circuit électronique actif de la carte mère et il est alimenté par le port USB. Il fournit une connexion série de débogage vers les cartes filles.

Pour l'architecture matérielle du segment spatial MASAT1, la carte mère NanoDock DMC-3 a permis d'héberger deux sous-systèmes : le module OdB NanoMind AX3200 et le module COM NanoCom AX100.

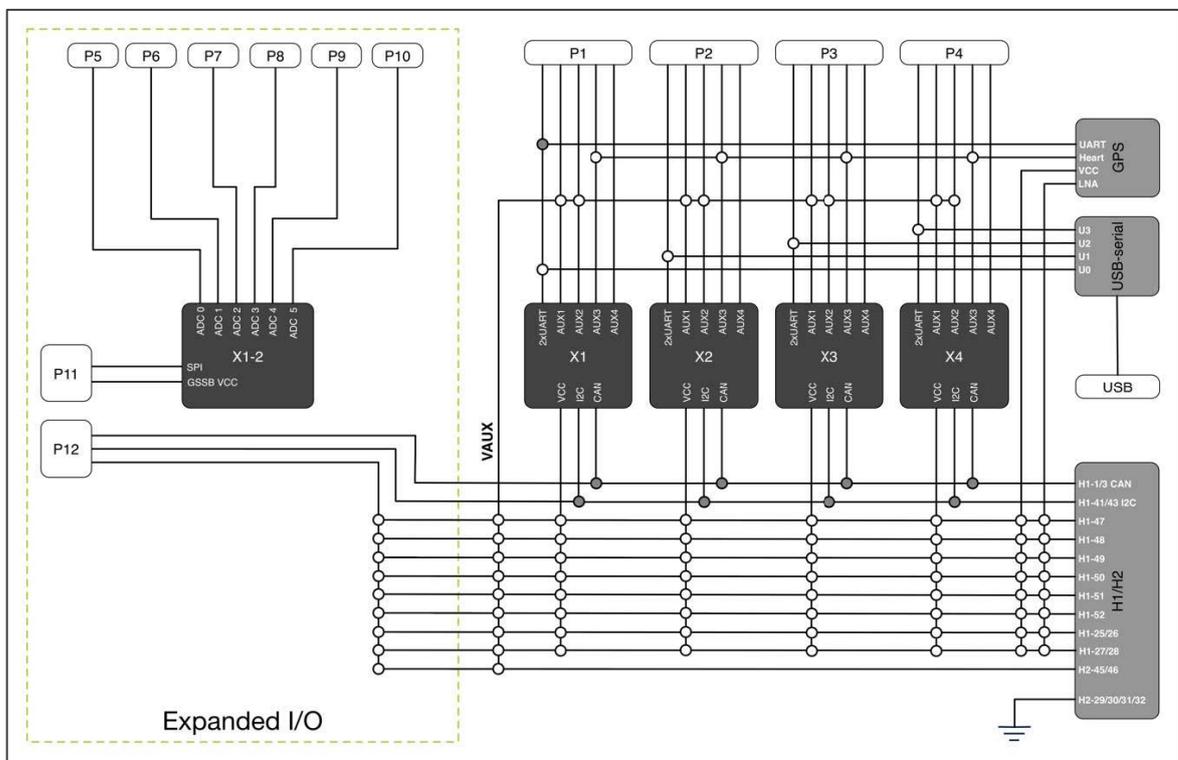


Figure 5-12 : Diagramme bloc de la carte mère Nano Dock DMC-3 [93]

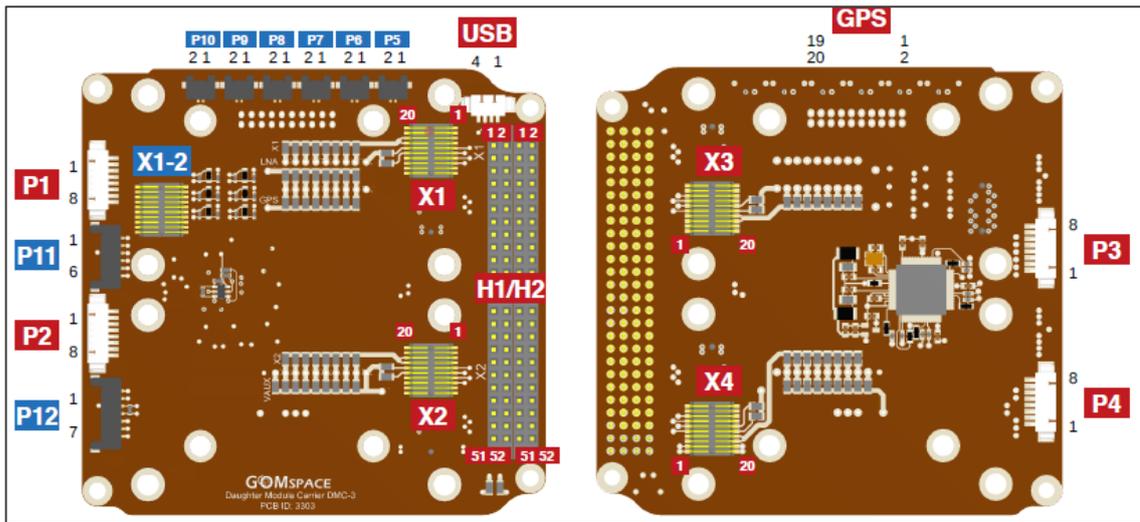


Figure 5-13 : Emplacement des connecteurs dans le carte mère Nano Dock DMC-3 [93]

2.2.1. Interfaçage du module OdB NanoMind AX3200

Le module OdB sera branché avec la carte mère au niveau des connecteurs :

- X1 pour l'alimentation, la communication via le bus I2C et le débogage via l'interface UART. Il permet aux autres sous-systèmes de la plateforme d'accéder à l'OdB via le connecteur H1/H2.
- X1-2 pour l'accès aux entrées Analogiques (P5 à P10) et l'interface SPI (P11).

Du point de vue plateforme du segment spatial, l'OdB NanoMind A3200 peut être accessible depuis le bus PC-104 via les broches du connecteur H1/H2 décrites par le *tableau 5-3* [107 -108] :

Broche	Description
H1.47 – H1.48 – H1.49 H1.50 – H1.51 – H1.52	Entrées d'alimentation configurables et commandées depuis la carte NanoPower P31u
H2.27 + H2.28	Tension d'alimentation permanente Vcc (+3.3V)
H2.25 + H2.26	Tension d'alimentation permanente Vcc (+5V)
H2.29 + H2.30 + H2.31	Masse
H1.1	CAN-LOW
H1.3	CAN-HIGH
H1.41	I2C SDA
H1.43	I2C SCL

Tableau 5-3 : Broches du bus PC-104 permettant d'accéder au sous-système OdB

2.2.2. Interfaçage du module COM NanoCom AX100

Le module COM sera branché avec la carte mère au niveau du connecteur X2 pour l'alimentation, la communication via le bus I2C et le débogage via l'interface UART.

Du point de vue plateforme du segment spatial, le module NanoCom AX100 peut être accessible depuis le bus PC-A04 via les broches du connecteur H1/H2 décrites par le *tableau 5-4* [91] [93] :

Broche	Description
H1.47 – H1.48 – H1.49 H1.50 – H1.51 – H1.52	Entrées d'alimentation configurables et commandées depuis la carte NanoPower P31u
H2.27 + H2.28	Tension d'alimentation permanente Vcc (+3.3V)
H2.25 + H2.26	Tension d'alimentation permanente Vcc (+5V)
H2.29 + H2.30 + H2.31	Masse
H1.1	CAN-LOW
H1.3	CAN-HIGH
H1.41	I2C SDA
H1.43	I2C SCL

Tableau 5-4 : Broches du bus PC-104 permettant d'accéder au sous-système COM

La *figure 5-14* représente le montage des cartes OdB et COM sur la carte mère, réalisé à l'aide du logiciel SOLIDWORKS :

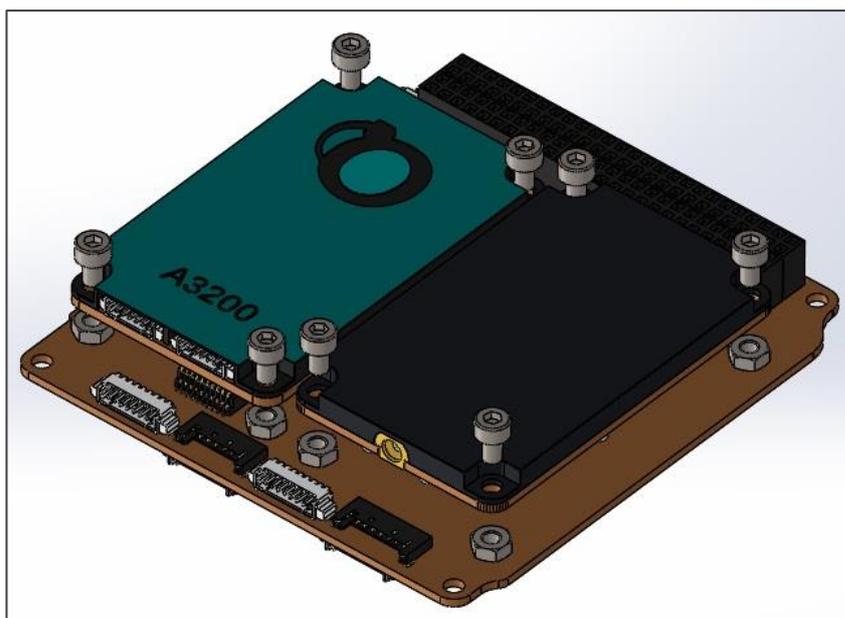


Figure 5-14 : Vue 3D des cartes OdB et COM montée sur la carte mère [SOLIDWORKS]

2.2.3. Diagramme bloc électrique de la plateforme MASAT1

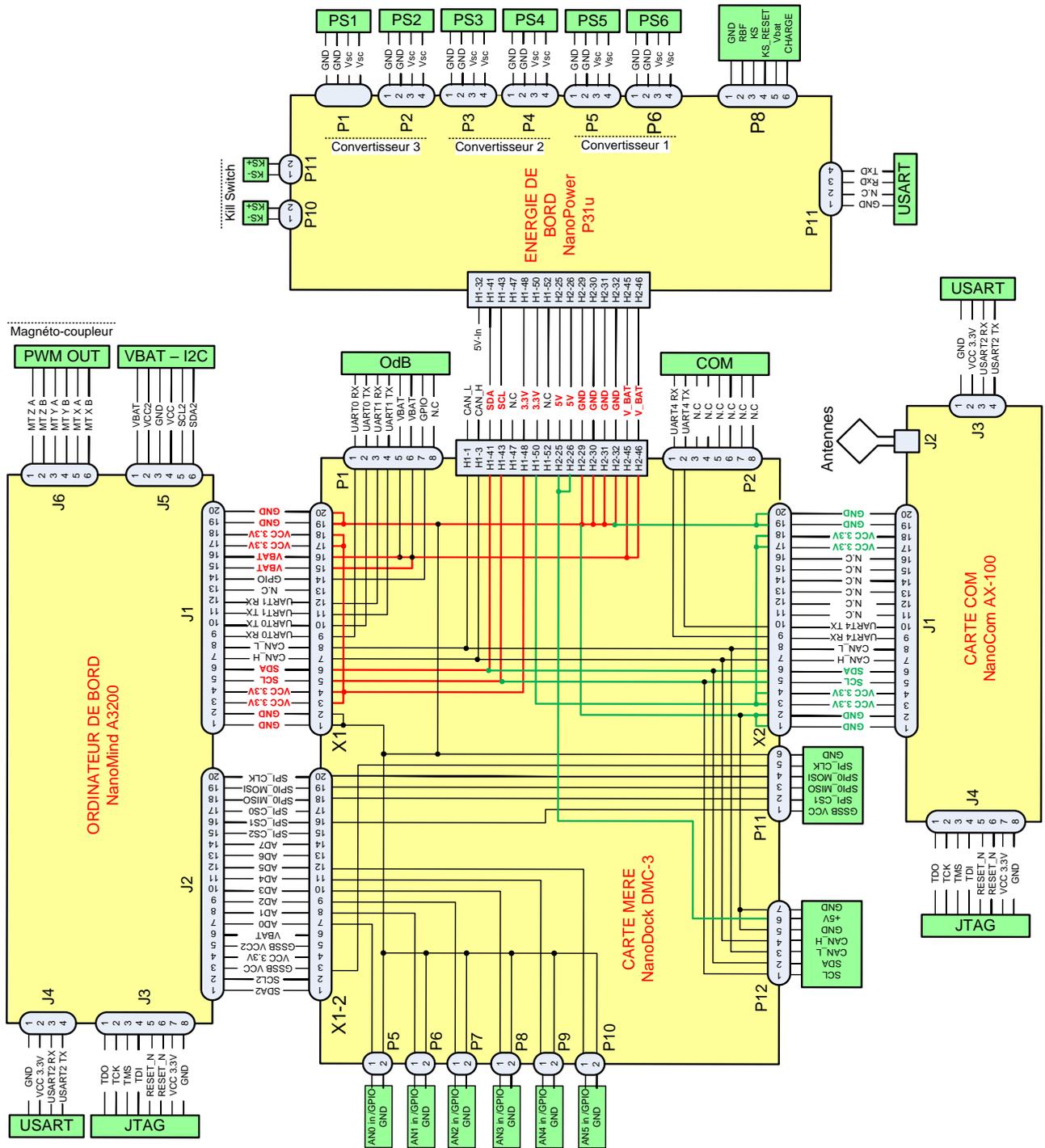


Figure 5-15 : Diagramme bloc électrique de la plateforme MASAT1

3. ETUDE ET DESCRIPTION DU SOUS-SYSTEME STRUCTURE (STR)

3.1. Fonctionnalité de la structure d'un Cubesat

L'étude de la structure du segment spatial a pour but de fournir une structure solide, légère et simple permettant de :

- Interfacer mécaniquement le CubeSat et le lanceur via le système de déploiement Poly Picosatellite Orbital Deployer (P-POD) décrit au § 3.3 de l'introduction générale. Ce dernier se compose d'un conteneur rectangulaire pouvant contenir trois Cubesats empilés les uns sur les autres et la structure de chaque CubeSat doit obligatoirement comporter à sa base des ressorts de séparation, afin de garantir la séparation des trois cubesats après l'éjection comme le montre la *figure 2.72*.



Figure 5-16 : Vue perspective du P-POD et des ressorts de séparation [10]

- Survivre aux contraintes de lancement à savoir les charges statiques et dynamiques, les vibrations durant le vol et les chocs lors de l'allumage et la séparation.
- Assurer un environnement approprié pour le bon fonctionnement de tous les sous-systèmes en les protégeant contre l'environnement spatial hostile, à savoir les radiations, les débris et les variations thermiques.
- Permettre d'insérer et d'assembler facilement les cartes qui équipent le segment spatial.
- Permettre un accès facile aux données et au bus d'alimentation pendant le test et le débogage.

Comme le montre la *figure 5-17*, La structure peut ainsi être divisée en trois blocs fonctionnels :

- La structure primaire qui est constituée du cadre cubique extérieur.
- La structure secondaire qui est constituée de toute partie ne supportant pas les charges structurelles dues au lanceur (panneaux solaires, vis sans fin, etc.).
- L'interface avec le P-POD qui comprend des éléments comme les rails, les pieds, etc.

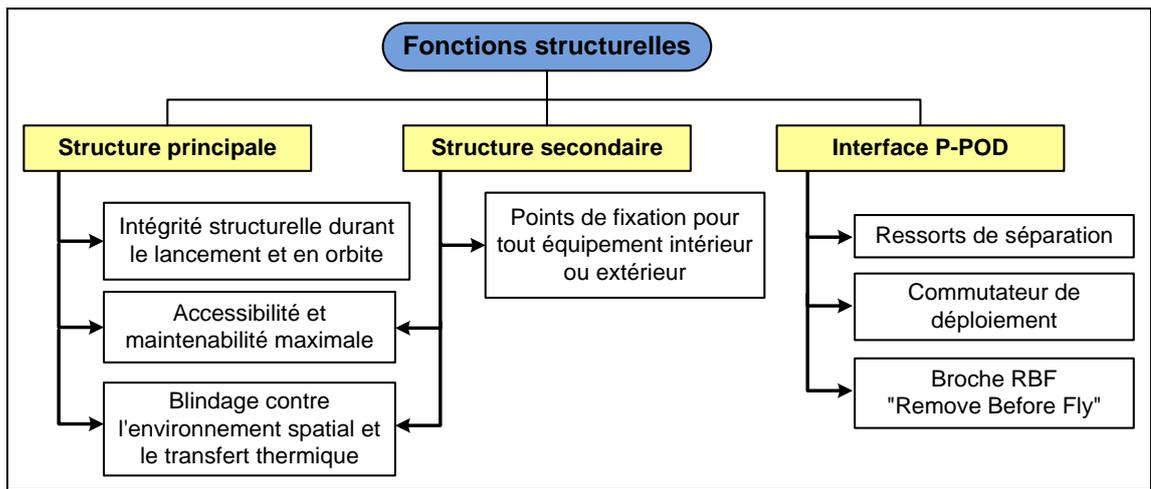


Figure 5-17 : Diagramme fonctionnel de la structure d'un Cubesat [62]

3.2. Exigences du sous-système STR pour la mission MASAT1

[DES] FONC-1405

Assurer un environnement sécurisé et approprié pour le bon fonctionnement de tous les sous-systèmes

[DES] FONC-1406

La structure doit permettre un accès facile pendant l'intégration des sous-systèmes et le contrôle du segment spatial au sol.

[DES] CONC-1410

Être conforme à la dernière version des spécifications relatives aux cubesat (V13)

[DES] CONC-1415

Assurer un interfaçage mécanique entre le CubeSat et le lanceur via le système de déploiement appelé Poly Picosatellite Orbital Deployer (P-POD), ou KiboCUBE de la station spatiale internationale ISS.

[DES] CONC-1420

Le CubeSat doit utiliser un système de coordonnées tel que défini dans les spécifications (La face -Z sera insérée en premier dans le système de déploiement)

[DES] CONC-1430

Les rails doivent avoir une largeur minimale de 8,5 mm.

[DES] CONC-1440

Les rails ne doivent pas avoir une rugosité de surface supérieure à 1,6 μm .

[DES] CONC-1450

Les bords des rails doivent être arrondis à un rayon inférieur à 1 mm.

[DES] CONC-1460

Les extrémités des rails sur la face +Z doivent avoir une surface minimale de 6,5 mm x 6,5 mm comme zone de contact avec les rails voisin du CubeSat

[DES] CONC-1470

Au moins 75% des rails doit être en contact avec les rails P-POD. 25% des rails peut être en retrait et aucune partie des rails ne doit dépasser la spécification Cubesat.

[DES] CONC-1480

Le centre de gravité du CubeSat doit être situé dans une sphère de 2 cm de son centre géométrique.

[DES] CONC-1490

La structure doit être composée d'aluminium 7075 ou 6061.

[DES] CONC-1500

Une broche "Supprimer avant vol" RBF (Remove Before Flight) doit être incorporée.

[DES] CONC-1510

Au moins un commutateur de déploiement indiquant le déploiement du Cubesat doit être incorporé à la structure.

[DES] CONT-1530

La structure doit survivre dans l'environnement de lancement et en orbite.

3.3. Choix de la solution technologique

Le choix de la structure de base doit respecter les exigences de la mission et les choix réalisés au niveau des solutions de conception de MASAT1, à savoir une disposition en couche pour la configuration du segment spatial (Chapitre 2 § 3.4.2).

Pour réduire le temps de développement global du projet, deux alternatives pour le choix de la structure de base ont été identifiées (voir **figure 5-18**) :

- Structure pour Cubesat 1U fournit par le fabricant Pumpkin
- Structure pour Cubesat 1U fournit par le fabricant ISIS

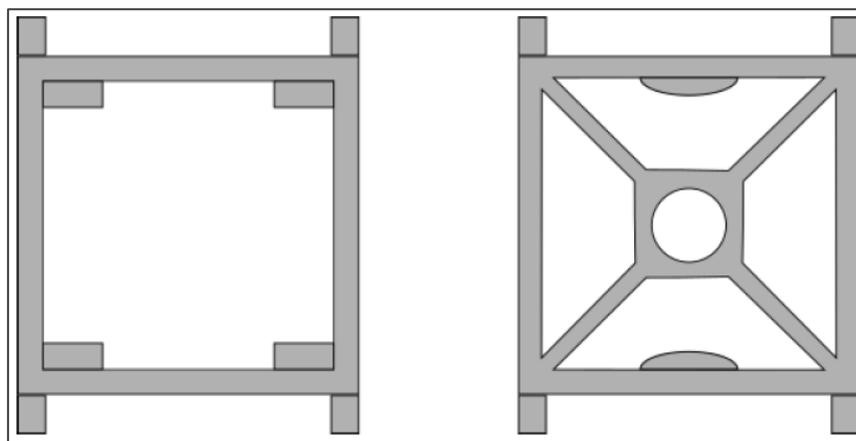


Figure 5-18 : Structure ISIS CubeSat (à gauche) et Structure Pumpkin Cubesat (à droite)

On a décidé comme première approche d'utiliser le Kit Cubesat de Pumpkin comme structure mécanique principale puisque cette solution répond aux fonctionnalités recherchées. La structure offre en plus la possibilité de fixer les panneaux solaires et le système permettant le déploiement des antennes sur n'importe quelle face du cube.

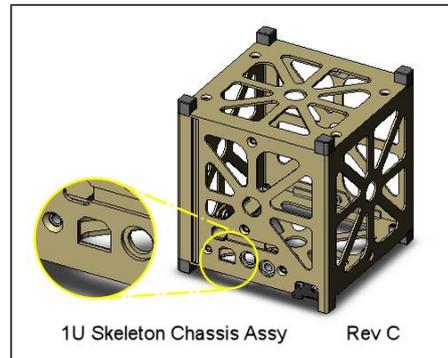


Figure 5-19 : Vue perspective de la structure Cubesat 1U Pumpkin [145]

Le *figure 5-20* montre l'intégration prévue des sous-systèmes de la plateforme du segment spatial MASAT1 à l'intérieur de la structure Pumpkin :

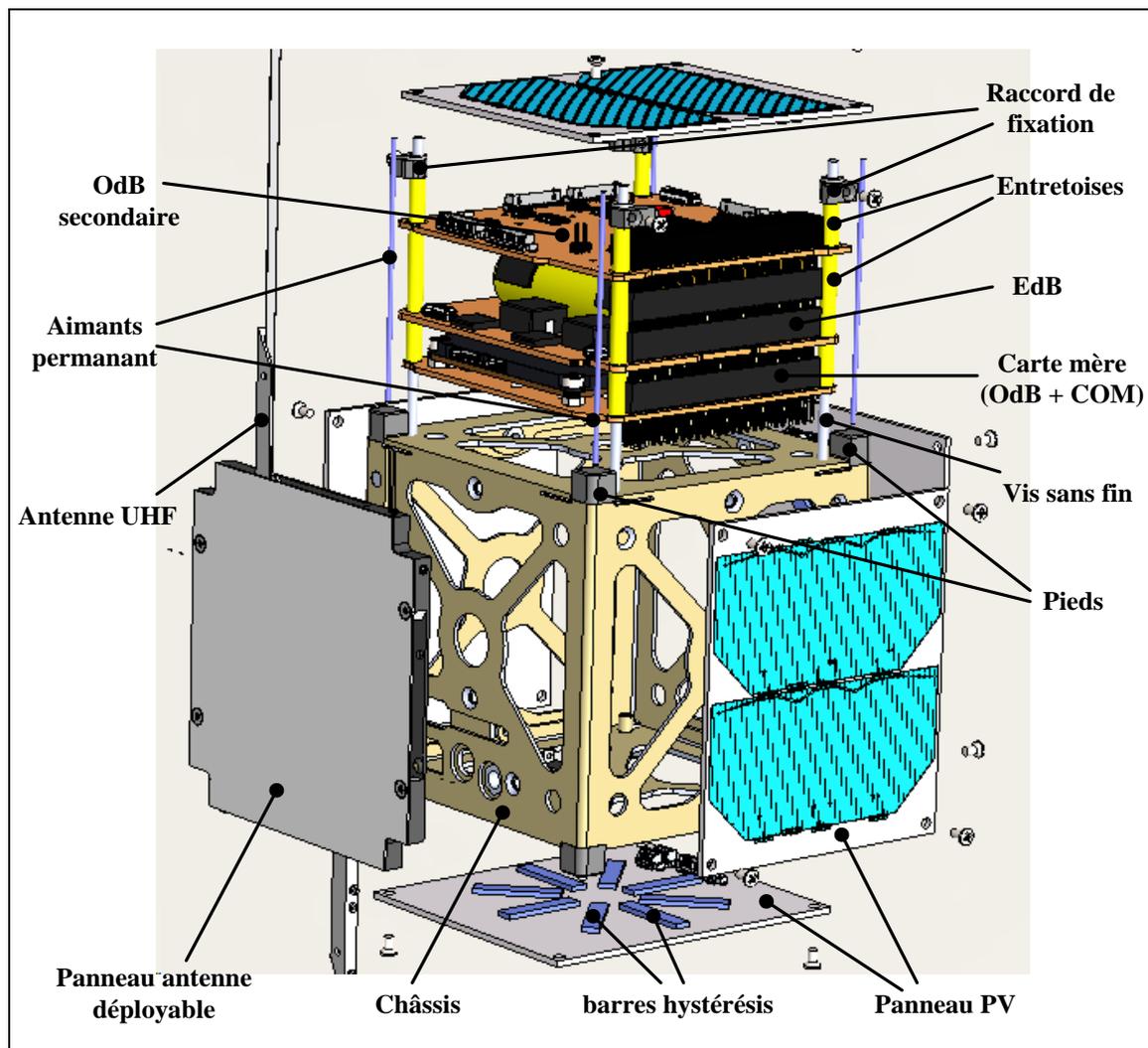


Figure 5-20 : Vue 3D de la configuration du segment spatial MASAT1 [SOLIDWORKS]

3.4. Analyse de la structure

3.4.1. Bilan de masse

En raison des spécifications de la norme CubeSat, le segment spatial doit peser moins de 1,33 kg avec obligatoirement une marge de 10%. Le *tableau 2-31* résume le bilan de masse préliminaire du segment spatial MASAT1 :

Assemblages	Parties	Nombre	Poids (gr)
Face supérieure	Panneau solaire ISIS (Substrat aluminium 1,8 mm)	1	50
	Plaque de recouvrement	1	26,3
	Pied	4	5,3
	Rondelles M3	4	0,2
	Vis M3	4	1,5
	Coins	4	0,6
	Total		
Face inférieure	Antenne déplorable ISIS	1	50
	Panneau solaire ISIS (Substrat aluminium 1,8 mm)	1	50
	Plaque de base	1	35,9
	Pied	1	1,3
	Pied foré	3	3,6
	Vis à ressort	2	1,2
	Commutateur de déploiement	1	2,4
	Boulons M3	2	1,1
	Rondelles	2	0,1
	Vis M3	1	0,4
	Vis M1	1	0,4
	Coins	4	0,6
Total			147
Faces latérales	Panneau solaire ISIS (Substrat aluminium 2,5 mm)	4	200
Total			200
Structure	Structure	1	63,3
	Vis sans fin	4	14,6
	Vis M3	10	3,2
Total			81,1
Energie de bord	Module NanoPower p31u + Batteries	1	200
	Entretoises 25mm	4	1,9
Total			201,9
Ordinateur de bord	Module NanoMind a3200	1	24
	Vis M3	4	1,5
Total			25,5
Communication	Module NanoCom ax100	1	24,5
	Vis M3	4	1,5
Total			26
Carte mère	Module NanoDock dmc3	1	51
	Entretoises 15mm	4	1,1
Total			52,1
CUT (OdB secondaire)	Module à base de FPGA Spartan 6	1	60
	Entretoises 15mm	4	1,1
Total			61,1
Contrôle d'attitude	Aimant	4	4
	Barres hystérisis	16	1,5
	Fixation	3	7
Total			12,5
General	Boulons M3	4	2,2
	Câbles	//	30
Total			32,2
Masse Totale + 10%			1015,63

Tableau 5-5 : Bilan de masse préliminaire de la mission MASAT1

3.4.2. Centre de gravité et centre d'inertie

Le Centre de gravité (CdG) doit respecter l'exigence qui impose son emplacement à moins de 2 cm du centre géométrique du CubeSat [71]. Le modèle SOLIDWORKS utilisé repose sur l'hypothèse d'une densité uniforme pour les pièces métalliques comme les panneaux PV ou le châssis, et la répartition des composants est supposée uniforme sur toute la surface des circuits imprimés des cartes électroniques.

La *figure 5-21* donne les résultats concernant la position du centre de gravité du segment spatial MASAT1 avec une modélisation sous SOLIDWORKS :

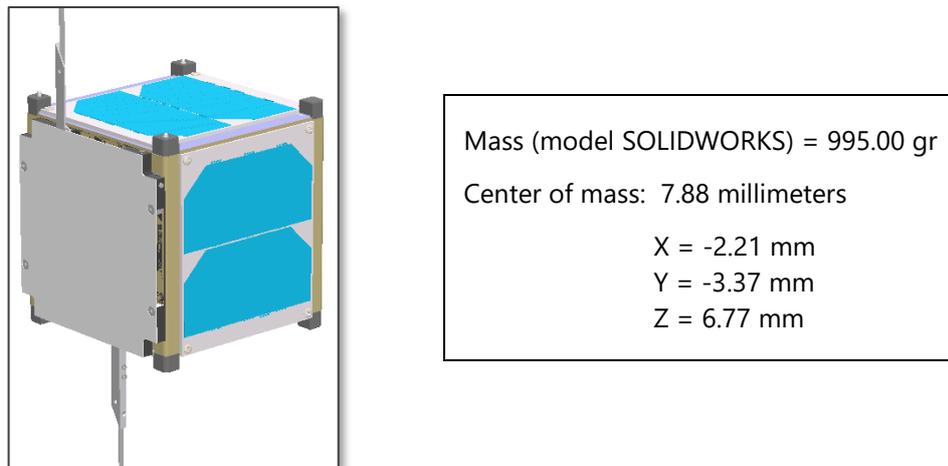


Figure 5-21 : Centre de masse du segment spatial MASAT1 [SOLIDWORKS]

L'emplacement du CdG le long de l'axe Y s'explique par les propriétés asymétriques des cartes électroniques au niveau du connecteur PC-104. L'écart le long de l'axe Z peut être minimisé en déplaçant la carte EdB qui contient les batteries. Enfin, le léger écart le long de l'axe X s'explique par la différence de masse entre le panneau PV et le système de déploiement des antennes.

À partir de cette simulation préliminaire sous SOLIDWORKS, nous pouvons conclure que l'exigence sur le positionnement du CdG est largement remplie.

La même modélisation a permis de calculer le point d'inertie qui a été défini à partir du CdG en alignement avec le centre géométrique du segment spatial :

$$\begin{pmatrix} L_{xx} = +2.235 & L_{xy} = +2.464 & L_{xz} = -2.471 \\ L_{yx} = +2.464 & L_{yy} = +2.118 & L_{yz} = -2.109 \\ L_{zx} = -2.471 & L_{zy} = -2.109 & L_{zz} = +2.304 \end{pmatrix} 10^6 \cdot g \cdot mm^2$$

Les propriétés d'inertie étant une caractéristique fondamentale d'un corps tridimensionnel, elles seront nécessaires dans le dimensionnement du contrôle d'attitude passif de la mission MASAT1.

4. DESCRIPTION DU SOUS-SYSTEME CONTROLE D'ATTITUDE (ATT)

4.1. Présentation du contrôle d'attitude passif

Le sous-système de contrôle d'attitude permet d'assurer une orientation optimale du segment spatial en fonction des objectifs de la mission. Puisque la charge utile de la mission MASAT1 n'impose pas une orientation du segment spatial vers une direction précise, on a opté pour un contrôle d'attitude passif par stabilisation magnétique.

En effet, et comme le montre la *figure 5-22*, la stabilisation magnétique passive reste une technique populaire et particulièrement adaptée pour les Cubesat au format 1U [34] :

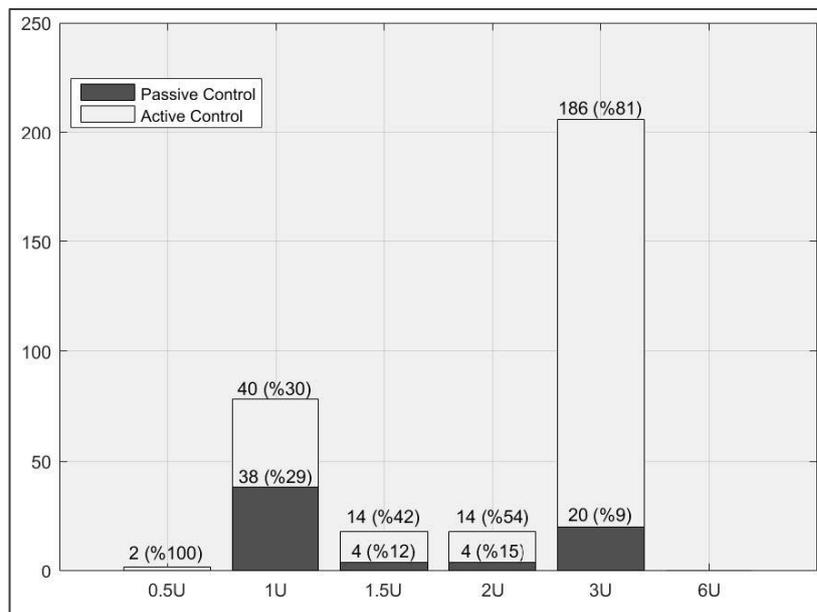


Figure 5-22 : Méthodes de contrôle d'attitude en fonction de la taille du Cubesat [34]

Le principe de la stabilisation magnétique passive consiste à utiliser un aimant permanent pour aligner le satellite avec le vecteur du champ magnétique terrestre local et contrôler donc le satellite dans deux directions autres que l'axe magnétique qui reste incontrôlé, comme le montre la *figure 5-23*.

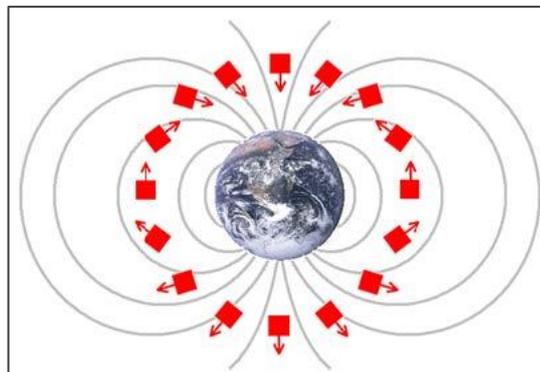


Figure 5-23 : Alignement d'un cubesat sur le champ magnétique terrestre [Internet]

Même si cette technique de contrôle d'attitude ne consomme pas de puissance et ne nécessite pas de traitement logiciel, elle présente deux sources d'erreurs dans le contrôle de l'attitude du satellite [14] :

- L'aimant permanent induit un mouvement oscillatoire indésirable du satellite autour des lignes de champs car le champ magnétique terrestre change à mesure que le cubesat se déplace, ce qui provoque un retard dans l'alignement. L'utilisation de barres d'hystérésis permet d'amortir ces oscillations.
- Le moment magnétique introduit par les barres d'hystérésis entraîne une erreur d'alignement stationnaire du moment magnétique total le long de l'axe magnétique de Cubesat, ce qui diminue la précision du pointage. Cette erreur ne peut être annulée en changeant l'orientation des barres d'hystérésis vu que leur polarité commute constamment.

Une bonne conception d'un sous-système de contrôle d'attitude passif par stabilisation magnétique consiste à trouver un équilibre entre ces deux effets indésirables lors du dimensionnement des barres d'hystérésis et de l'aimant permanent.

4.2. Etapes pour la conception du sous-système ATT

4.2.1. Détermination des couples perturbateurs agissant sur le CubeSat

La première étape cruciale dans la conception d'une stabilisation magnétique passive est le calcul des couples qui agissent sur le Cubesat. En effet, le couple total $\vec{\Gamma}_t$ exercé sur un satellite orbitant autour de la terre, est composé des couples perturbateurs $\vec{\Gamma}_p$ et du couple de contrôle $\vec{\Gamma}_c$ généré par les barres d'hystérésis et l'aimant permanent :

$$\vec{\Gamma}_t = \vec{\Gamma}_p + \vec{\Gamma}_c \quad (5.2)$$

- Les couples perturbateurs produits par l'environnement spatial vont empêcher le Cubesat de rester dans une orientation donnée par rapport à un référentiel galiléen. Les principales sources de ces couples sont : le gradient de gravité de la terre, la trainée atmosphérique, la pression des radiations solaires et les dipôles magnétiques résiduels.
- Le couple de contrôle généré par les barres d'hystérésis et de l'aimant permanent doit être dimensionner afin de compenser les couples perturbateurs et relativement contrôler l'attitude du Cubesat.

Comme le montre le **tableau 5-6**, La magnitude des couples perturbateurs reste petite en comparaison avec le standard terrestre (de l'ordre de 10^{-8} N.m) mais ils doivent être pris en compte dans l'espace, où il n'y a pas de friction [14] [94] :

Couple perturbateur	Valeur (N.m)
Dipôle magnétique résiduel	$4.00 \cdot 10^{-7}$
Gradient de gravité	$1.75 \cdot 10^{-9}$
Pression des radiations solaires	$1.64 \cdot 10^{-9}$
Trainée atmosphérique	$6.97 \cdot 10^{-9}$

Tableau 5-6 : Ordre de grandeurs des couples perturbateurs pour les missions CubeSat 1U MCubed-2, AAU Sat et CalPoly PolySat [94]

4.2.2. Choix et dimensionnement des aimants permanents

La première étape consiste à évaluer le couple de contrôle des aimants permanents, en commençant par l'expression de l'induction électromagnétique $\vec{B}(\mathbf{T})$ en fonction de l'excitation magnétique \vec{H} et la magnétisation \vec{M} (A/m) :

$$\vec{B} = \mu_0(\vec{H} + \vec{M}) \quad (5.3) \quad \text{avec } \mu_0 \text{ est la perméabilité du vide}$$

On définit ensuite l'induction rémanente d'un aimant permanent ($\vec{H} = \mathbf{0}$) :

$$\vec{B}_{r(\vec{H}=\mathbf{0})} = \mu_0 \vec{M} \quad (5.4)$$

A partir de sa magnétisation \vec{M} , on peut ensuite calculer l'intensité de l'aimant permanent à l'aide de son moment magnétique défini par :

$$\vec{m} = \vec{M} \cdot V \quad (5.5) \quad \text{avec } V \text{ le volume de l'aimant}$$

Vu que l'aimant a tendance à s'aligner avec le champ magnétique terrestre, un couple va se produire et ce dernier peut être calculé en appliquant la formule suivante :

$$\vec{\Gamma} = \vec{m} \wedge \vec{B}_{earth} \Rightarrow \|\vec{\Gamma}\| = V \cdot \|\vec{M}\| \cdot \|\vec{B}_{earth}\| \cdot \sin(\theta) \quad (5.6)$$

$$B_{earth} = \frac{\mu_0 \cdot M_e}{4\pi r^3} \sqrt{1 + \sin(\lambda)^2} \quad (5.7)$$

Avec :

- B_{earth} : Induction du champ magnétique terrestre
- θ : angle entre le vecteur champ magnétique terrestre et celui de l'aimant.
- V : Volume de l'aimant permanent
- $\frac{\mu_0 \cdot M_e}{4\pi}$: Moment magnétique de la terre
- R : la distance entre le cubesat et au centre de la terre
- λ : la latitude magnétique

Il est à noter que les ingénieurs de la mission UNISAT-4 ont recommandé dans leurs travaux [14] [94], une formule générale pour le calcul du moment magnétique des aimants permanents dans une stabilisation magnétique passive :

$$m \leq 10 \cdot \frac{T_{RMS}}{B_{min} \cdot \sin(\beta_{max})} \quad (5.8)$$

Avec :

- m : moment magnétique de l'aimant.
- T_{RMS} : moyenne quadratique des couples perturbateurs.
- B_{min} : Induction magnétique minimale du champ à 600 km.
- β_{max} : Précision de pointage désirée.

Le **tableau 5-7** résume les caractéristiques des deux types d'aimants généralement utilisés dans les missions Cubesat :

Type d'aimant	$\rho [g/cm^3]$	$H_c [A/m]$	$B_r [T]$
AlNiCo-5	7.3	$5.09 \cdot 10^4$	1.25
Neodyme Iron bore (NdFeB)	7.3	$1.50 \cdot 10^6$	1.3

Tableau 5-7 : Caractéristiques des aimants utilisés dans les mission Cubesat [94]

Le choix de l'aimant permanent approprié est généralement basé sur un processus itératif dont les principaux critères sont [94] :

- L'aimant permanent doit être adapté aux contraintes d'utilisation dans un cubesat, avec une masse et une taille réduite.
- L'induction rémanente (B_r), qui mesure la force de l'aimant, doit être suffisamment grand pour vaincre les couples perturbateurs et appliquer le couple nécessaire au contrôle du cubesat.
- Le champ coercitif (H_c), qui mesure la résistance du matériau à la démagnétisation, doit être suffisamment grand pour que le champ magnétique externe élevée n'affecte pas l'aimantation de l'aimant.
- La stabilité thermique de l'aimant doit rester stable dans la gamme de température opérationnelle du cubesat.
- La résistance à la corrosion en protégeant la surface de l'aimant par des revêtements à haute résistance à la corrosion comme l'acier inoxydable.

4.2.3. Choix et dimensionnement des barres hystérésis

L'utilisation de matériau magnétique hystérésis est une solution passive et simple pour le problème d'oscillation. Elle permet de fournir l'amortissement nécessaire pour limiter et éliminer la vitesse angulaire excessive du segment spatial après sa séparation du lanceur.

Les matériaux magnétiques hystérésis se comportent comme des aimants permanents sauf qu'ils ont une perméabilité très haute (ferromagnétisme). Alors, lorsque le Cubesat tourne en orbite LEO en subissant des oscillations induites par l'aimant permanent, les barres hystérésis connaissent une variation du champ magnétique terrestre et ont tendance à varier l'alignement de leurs dipôles micro-magnétiques. Leur aimantation va changer selon un cycle d'hystérésis qui produit une dissipation frictionnelle d'énergie perdue à l'échelle moléculaire sous forme de chaleur [95].

La **figure 5-24** représente la réponse d'un matériel ferromagnétique à une excitation magnétique sinusoïdale, avec la zone hachurée entre les courbes ascendante et descendante qui représente l'énergie perdue sous forme de chaleur :

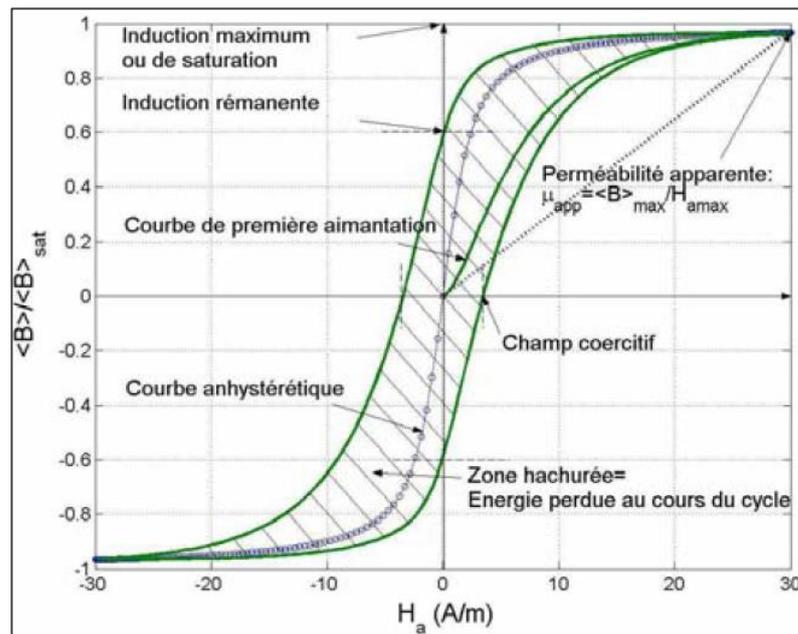


Figure 5-24 : Cycle d'hystérésis typique [95]

En retardant le changement de leurs polarités, en réponse au changement du champ magnétique terrestre, les barres hystérésis permettent d'amortir le mouvement de rotation de cubesat en convertissant son énergie en énergie thermique. Pour maximiser cet amortissement, les barres hystérésis sont généralement montées par paires orthogonales à l'aimant permanent [14].

Les dimensions et le nombre de barres hystérésis sont déterminés en fonction des paramètres d'hystérésis du matériau utilisé, à savoir : le champ coercitif H_c , l'induction rémanente B_r et l'induction de saturation B_s .

L'approche analytique que nous avons adoptée pour déterminer ces paramètres repose sur les travaux de l'université du Kentucky concernant la stabilisation magnétique passive [96]. La détermination du couple généré par les barres d'hystérésis se fait en cinq étapes :

1. On détermine l'induction du champ magnétique terrestre $B_{earth\ ECI}$, au niveau du Cubesat, en le modélisant comme un dipôle magnétique au niveau du noyau de la terre (modèle L-Shell). D'autres modèles plus précis existent mais nécessitent des ressources informatiques plus importantes. Puis on passe du référentiel géocentrique ECI vers le référentiel fixe centré sur le centre de masse du cubesat, en utilisant la transformation suivante :

$$B_{earth\ body} = C_{bi} \cdot B_{earth\ ECI} \quad (5.9)$$

Où C_{bi} est la matrice de rotation du référentiel ECI vers le référentiel qui va décrire l'attitude du cubesat.

2. On calcule ensuite l'excitation magnétique correspondante :

$$H_{earth\ body} = \frac{B_{earth\ body}}{\mu_0} \quad (5.10)$$

3. L'induction magnétique induite dans la barres d'hystérésis B_{hyst} est ensuite calculée en utilisant un modèle approximatif pour le cycle d'hystérésis comme le montre la *figure 5-25* :

$$B_{hyst} = Courbe_{hysteresis}(H_{earth\ body}) \quad (5.11)$$

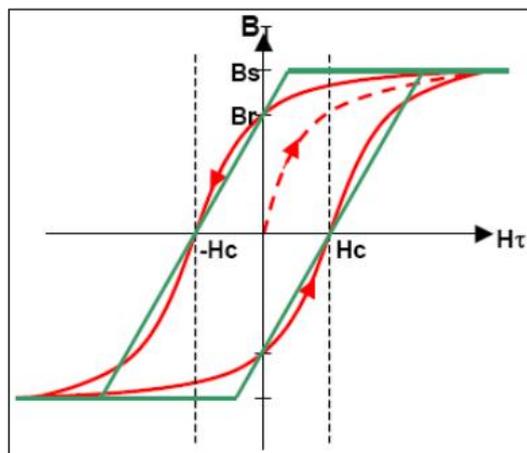


Figure 5-25 : Modèle approximatif du cycle d'hystérésis (en vert) [96]

4. L'étape suivante consiste à déterminer le moment magnétique \mathbf{m}_{hyst} fourni par la barre d'hystérésis en utilisant la formule :

$$\mathbf{m}_{hyst} = \frac{\mathbf{B}_{hyst} \cdot V_{hyst}}{\mu_0} \quad (5.12)$$

Où V_{hyst} représente le volume de la barre d'hystérésis.

5. Enfin, le couple généré par les barres d'hystérésis est calculé avec le produit vectoriel entre le moment magnétique et l'induction du champ magnétique terrestre :

$$\vec{\Gamma}_{hyst} = \vec{\mathbf{m}}_{hyst} \wedge \vec{\mathbf{B}}_{earth\ body} \quad (5.13)$$

Le **tableau 5-8** résume les caractéristiques des deux types de matériaux hystérésis généralement utilisés dans les missions Cubesat :

Matériaux hystérésis	$\rho [g/cm^3]$	$H_c [A/m]$	$B_r [T]$	$B_s [T]$
Hy-Mu-80	8.747	1.59	0.35	0.73
Permenorm	8.25	5	0.755	1.55

Tableau 5-8 : Matériaux hystérésis utilisés dans les mission Cubesat [94]

Le choix et le dimensionnement approprié des barres d'hystérésis sont généralement basés sur un processus itératif dont les principaux critères sont [94] :

- La fréquence des oscillations autour des lignes de champ magnétique terrestre augmente avec l'augmentation du couple générée par les aimants permanents.
- Le surdimensionnement des matériaux hystérésis diminue la précision de pointage de la stabilisation magnétique passive.
- Les matériaux utilisés dans la structure du satellite peuvent le protéger contre le champ magnétique terrestre et rendre la stabilisation magnétique passive moins efficace.

Chapitre 6

DESCRIPTION DE L'ORGANISATION LOGICIELLE

1. VUE D'ENSEMBLE

Pour répondre aux exigences de la mission MASAT1, il a été décidé que le segment spatial aura une architecture distribuée permettant un développement plus facile de chaque sous-système et une simplification des procédures d'essais dont beaucoup peuvent être réalisées avant l'intégration des sous-systèmes. Cette approche permettra de développer les sous-systèmes de façon indépendante.

Ainsi, chaque sous-système (COM, EdB, OdB et CUT) a son propre microcontrôleur et sera relié aux autres sous-systèmes via un bus de données I2C. Le logiciel de vol embarqué sera réalisé autour des bibliothèques logicielles de GOMSpace et de modules et drivers (langage C) exécutés au sein du système d'exploitation FreeRTOS. La communication interne entre les différents sous-systèmes se fera via le protocole couche réseau CSP (Cubesat Space Protocol) conçu spécialement pour les cubesats.

2. LES MODES OPERATIONNELS

Les modes opérationnels permettront de définir à tout moment les ressources actives du segment spatial ainsi que les sous-systèmes alimentés par le sous-système EdB. Les transitions entre les modes peuvent être effectuées via des télécommandes provenant de la station au sol, mais peuvent également, dans certaines situations, être déclenchées par le logiciel de vol embarqué.

2.1. Description

Le *tableau 6-1* donne une description des différents modes opérationnels qui ont été défini dans le ConOps de la mission MASAT1 (Chapitre 2 § 3.7) :

Mode	Acronyme	Numéro	Description
Mode Séparation	MS	0	<ul style="list-style-type: none"> ▪ Utilisé juste après la séparation du P-POD. ▪ Activation du sous-système Edb. ▪ Déploiement des antennes.
	MI	1	<ul style="list-style-type: none"> ▪ Utilisé après le mode séparation, lorsque le niveau batteries est critique (<10%) ou après une défaillance majeure. ▪ Consommation minimale de puissance. ▪ Récupération des données de télémétrie. ▪ Vérification des sous-systèmes. ▪ Envoie signal balise. ▪ Recharge batterie.
Mode Sans Echec	MSE	2	<ul style="list-style-type: none"> ▪ Utilisé au démarrage du système, après une défaillance critique ou lorsque le niveau batteries est faible (<20%). ▪ Récupération des données de télémétrie. ▪ Envoie signal balise. ▪ Recharge batterie. ▪ Ecoute des télécommandes provenant de la station sol.
Mode Nominal	MN_nTnCU	3	<ul style="list-style-type: none"> ▪ Utilisé lorsque le segment spatial est en mode nominal (niveau batteries > 20%) mais sans charge utile ni transmission. ▪ Récupération des données de télémétrie. ▪ Envoie signal balise. ▪ Recharge batterie. ▪ Ecoute des télécommandes provenant de la station sol.
	MN_TnCU	4	<ul style="list-style-type: none"> ▪ Utilisé lorsque le segment spatial est en mode nominal (niveau batteries > 20%). ▪ Transmission vers la station sol. ▪ Ecoute des télécommandes provenant de la station sol.
	MN_nJCU	5	<ul style="list-style-type: none"> ▪ Utilisé lorsque le segment spatial est en mode nominal (niveau batteries > 20%). ▪ Activation de l'ordinateur de bord secondaire (charge utile). ▪ Envoie signal balise. ▪ Ecoute des télécommandes provenant de la station sol.

Tableau 6-1 : Description des modes opérationnels de MASAT1

2.2. Transitions entre les modes opérationnels

Les transitions entre les modes de fonctionnement du segment spatial MASAT1 peuvent être effectuées depuis la station au sol via des TC, mais peuvent également être déclenchées par le logiciel embarqué en fonction de certaines données de télémétries mesurées à bord (niveau charge batterie, température).

La **figure 6-1** donne le diagramme préliminaire des transitions entre mode opérationnels, prévu pour la mission MASAT1 :

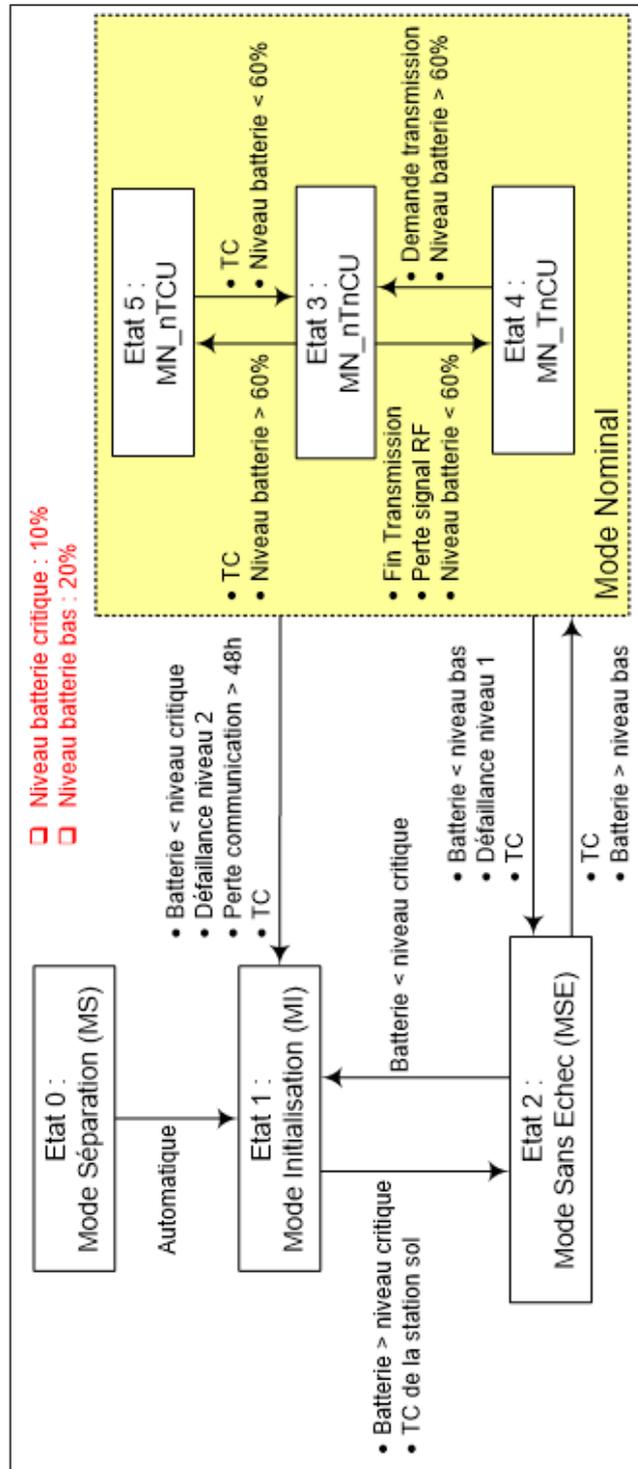


Figure 6-1 : Diagramme préliminaire des transitions entre les modes opérationnels

3. ALGORITHMES PRELIMINAIRES DES MODES FONCTIONNELS

3.1. Mode Séparation

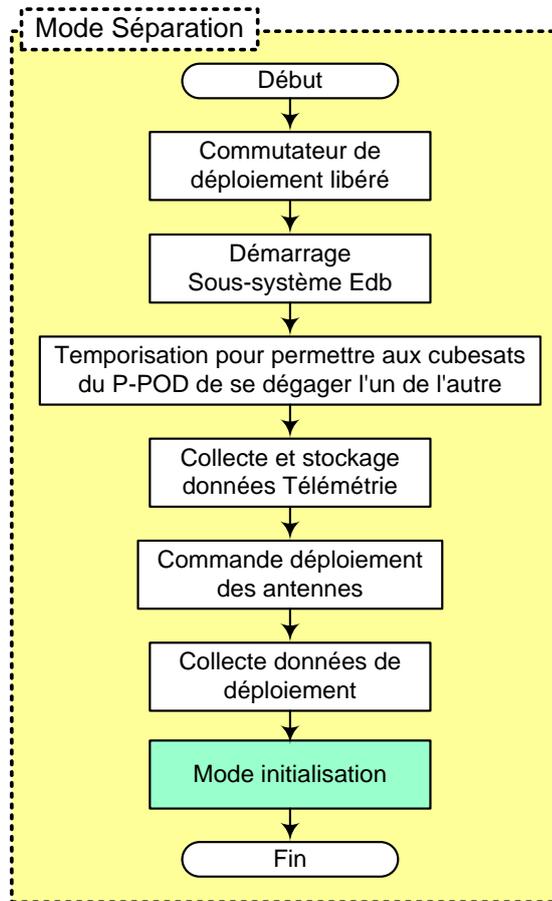


Figure 6-2 : Algorithme des opérations en mode séparation

3.2. Mode Initialisation

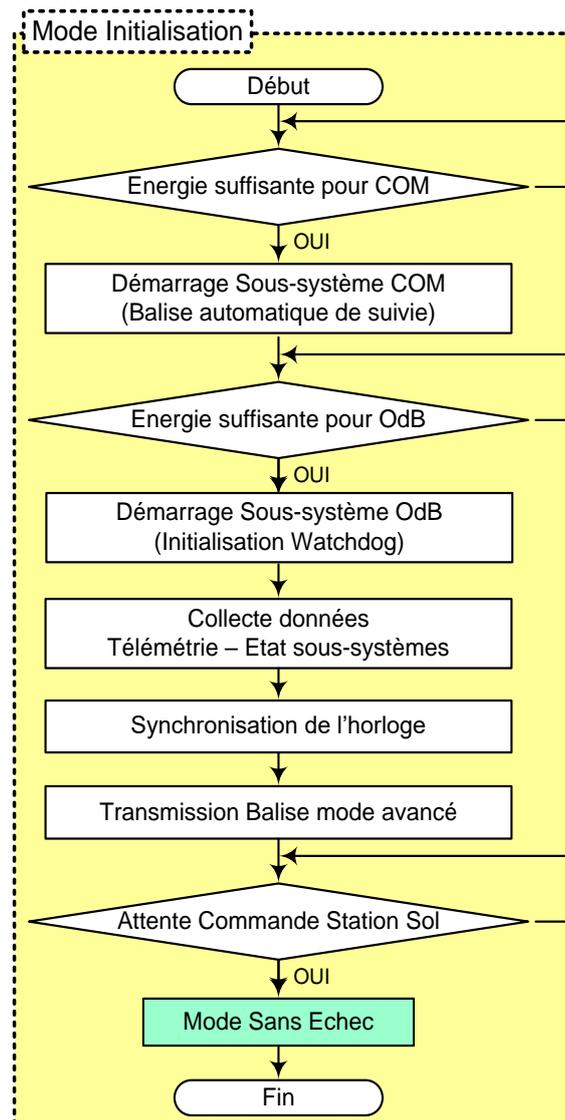


Figure 6-3 : Algorithme des opérations en mode Initialisation

3.3. Mode Sans Echec

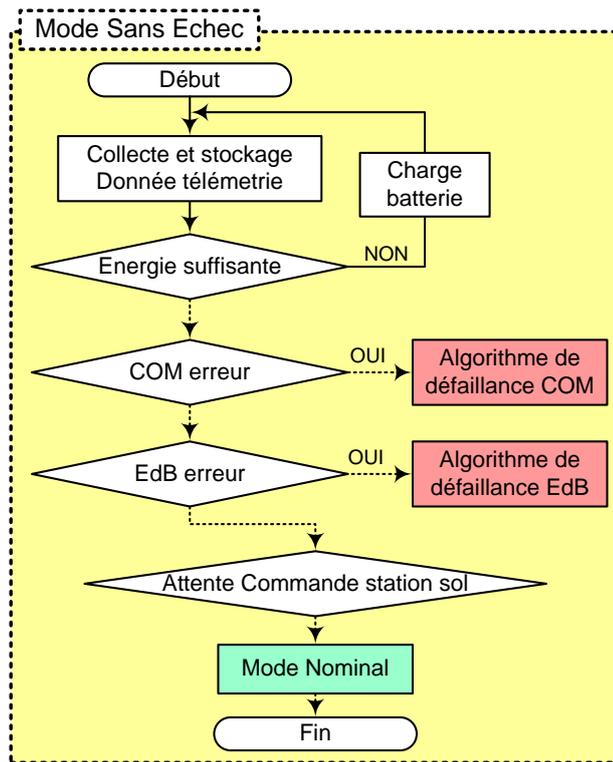


Figure 6-4 : Algorithme des opérations en mode Sans Echec

3.4. Mode Nominal

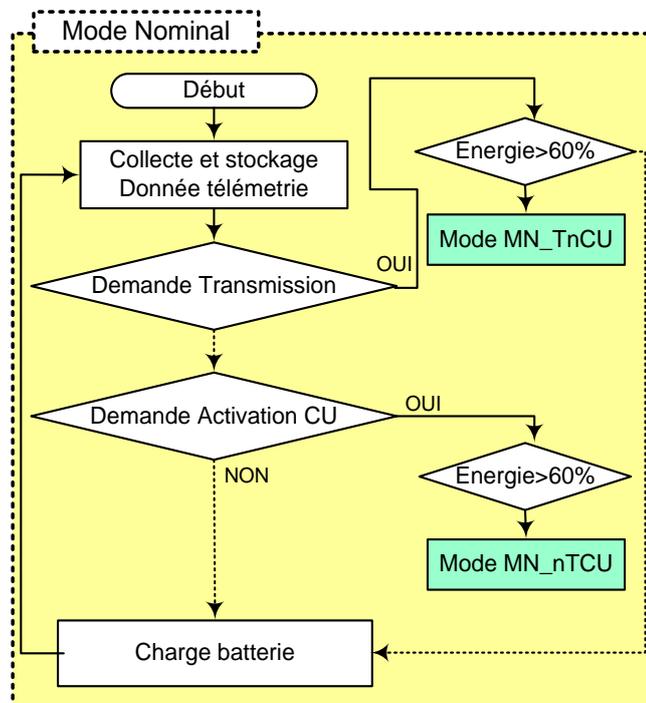


Figure 6-5 : Algorithme des opérations en mode Nominal

4. LA REFERENCE TEMPORELLE

La base de temps absolue de la mission MASAT1 est le temps universel coordonné UTC (Coordinated Universal Time). La donnée du temps sera une donnée 32 bits signée basée sur la convention d'horodatage et sera donc sous forme du nombre de secondes écoulées depuis 00:00:00 1 janvier 1970 UTC.

La référence temporelle est fournie par une horloge temps réel (RTC) intégrée à l'OdB et commandée par un quartz de 32768 KHz. Cette dernière fournira un "tick" contrôlant la fréquence d'exécution d'une interruption RTC qui mettra à jour la référence temporelle du segment spatial via des compteurs 32 bits.

Cette horloge RTC à base de quartz introduira au fur et à mesure un décalage entre la référence temporelle du segment spatial et celle au sol. Par conséquent, une TC de la station au sol est prévue pour synchroniser et mettre à jour l'horodatage du segment spatial à intervalles réguliers.

5. ARCHITECTURE DE COMMUNICATION (PROTOCOLES CSP)

Le CubeSat Space Protocol (CSP) est un protocole de couches réseau/transport visant à simplifier la connectivité entre les sous-systèmes embarqués d'une architecture distribuée, en permettant d'avoir une topologie de réseau orientée services [97]. Le protocole a été conçu à l'origine par des étudiants de l'Université d'Aalborg au Danemark et remis ensuite à la société danoise GOMSpace qui l'a exploité pour développer des solutions logicielles fiables et économiques destinées au cubesats.

L'implémentation de ses fonctionnalités de base est publiée par GOMSpace en tant que bibliothèque logicielle (libcsp.org) sous Licence LGPL (GNU Lesser General Public Licence). L'implémentation est écrite en GNU C, et a été portée sur plusieurs systèmes d'exploitation différents, y compris FreeRTOS le RTOS adopté pour notre mission.

La bibliothèque logicielle du protocole CSP peut être configurée pour utiliser différents pilotes pour la couche physique (I2C, CAN, FM Radio, etc.), ainsi que différentes techniques d'authentification et de contrôle de fiabilité.

5.1. Topologie

Le protocole CSP utilise une topologie et une terminologie orientée réseau similaire à la pile de protocole réseau TCP/IP [97]. La topologie utilisée pour la mission MASAT1

consiste à créer deux segments : l'un pour le segment spatial et l'autre pour la station au sol comme le montre la *figure 6-6* :

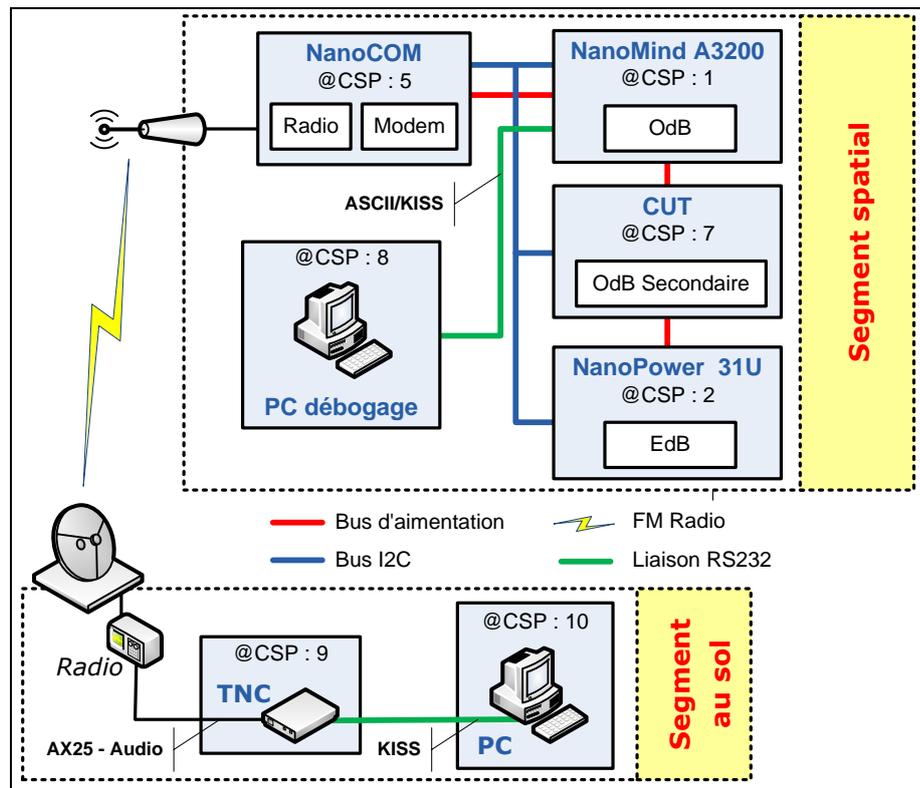


Figure 6-6 : Topologie réseau utilisée pour la mission MASAT1

La plage d'adressage allant de 0 à 15, a été segmentée en deux segments de taille égale avec toutes les adresses ayant le bit 1 comme poids fort (8 à 15) sont assignées au segment au sol, et toutes celles ayant le bit 0 comme poids faible (0 à 7) sont assignées au segment spatial. Il est à noter qu'à partir de la version CSP 1.0, l'espace d'adressage a été porté à 32 adresses.

Dans cette topologie, le nœud N°8 est différent des autres nœuds du segment au sol. En effet, chaque sous-système du segment spatial est configuré pour utiliser le sous-système COM pour contacter les nœuds allant de 9-15 et pour utiliser son propre port RS232 pour contacter le nœud 8. Cela signifie que le terminal de débogage peut être connecté directement à tout sous-système et utiliser le protocole CSP pour la communication.

Le réseau est configuré en utilisant des tables de routage statiques programmées dans le code source des différents sous-systèmes. Cela permet de tester et de développer le logiciel et le protocole de communication indépendamment pour chaque sous-système sans intégrer ce dernier avec les autres sous-systèmes et sans une liaison radio opérationnelle.

5.2. Architecture logicielle

L'architecture de la pile de protocole CSP est constituée de quatre couches avec des extensions avancées introduites par GOMSpace comme le montre la *figure 6-7* :

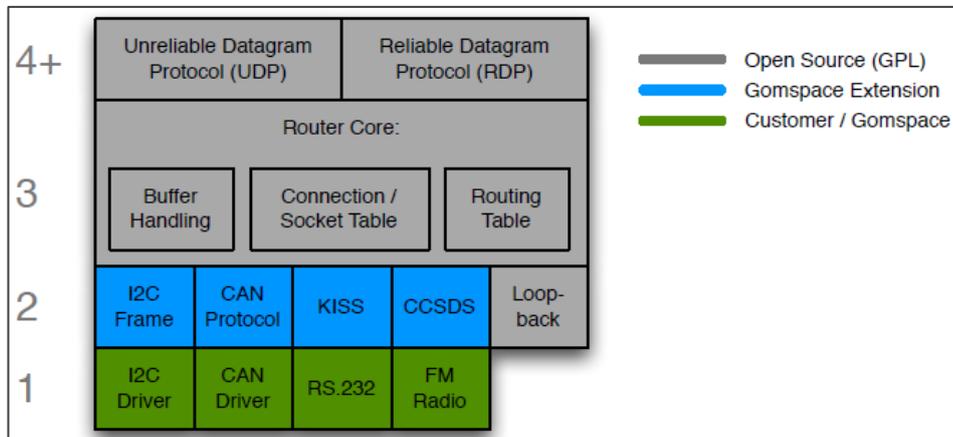


Figure 6-7 : La pile de protocole CSP [97]

5.2.1. Couche N°4 : Transport

Par similarité avec les modèles TCP/IP et OSI, cette couche permet la gestion de la transmission de paquets. Elle met en œuvre deux protocoles :

- UDP (Unreliable Datagram Protocol) qui fournit un service simple et peu fiable vu qu'il n'utilise pas d'accusées de réception pour garantir la fiabilité de la transmission des données. Les applications qui ont des contraintes temporelles (système temps réel par exemple) utilisent souvent UDP.
- RDP (Reliable Datagram Protocol) qui contrôle l'intégrité des données et la fiabilité de la transmission en ajoutant par exemple la retransmission automatique de paquets.

5.2.2. Couche N°3 : Noyau de routage

Le noyau de routage est l'épine dorsale de l'implémentation du protocole CSP. Non seulement il gère le routage, mais il permet aussi la gestion des tampons utilisés par tous les pilotes. Le routage fonctionne à l'aide de l'entête 32 bits de la trame, décrite à la *figure 6-8*, qui contient l'adresse destination (5 bits), l'adresse source (5 bits) et les numéros de port (6 bits) nécessaires pour la connexion.

Le routage n'est pas dynamique mais statique avec des tables de routage qui sont préprogrammées dans les sous-systèmes.

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Priority	Source				Destination				Destination port				Source port				Reserved	HMACE	XTEA	RDP	CRC										
32	Data (0-65536)																															

Figure 6-8 : Structure de l'entête CSP dans les trames de communication [97]

Les tables de routage sont au format CIDR (Classless InterDomain Routing) et chaque ligne doit respecter la syntaxe suivante :

<adresse>/<masque de sous réseau> <nom_inteface> <prochain saut>

Par exemple, si on considère la table de routage suivante du sous-système COM (@CSP = 5) de la plateforme MASAT1 :

```
nanocom-ax # route
5/5 LOOP
0/0 AX100
0/2 I2C
8/5 KISS
```

Le routage est organisé comme suit :

- 5/5 LOOP : boucle locale.
- 0/2 I2C : le trafic concernant les adresses 0 à 7 est acheminé vers I2C.
- 8/5 KISS : le trafic concernant l'adresse 8 est acheminé vers l'interface KISS.
- 0/0 AX100 : route par défaut vers l'interface radio AX100.

5.2.3. Couche N°2 : Accès au médium MAC (Medium Access Control)

Cette couche de liaison définit un format de trame qui convient au support et au protocole de transmission, dans le but :

- Soit de décoder correctement chaque trame reçue pour transmettre les données qu'elle contient au noyau de routage.
- Soit de coder et d'encapsuler les données provenant de la couche réseau dans un format de trame correspondant à la couche physique qui s'occupera de les transmettre.

Pour la mission MASAT1, on utilise trois types de formats :

- **Protocole AX25** pour les liaisons radio montantes et descendante. C'est un protocole de la couche liaison défini et utilisé par la communauté des radioamateurs, pour empaqueter aussi bien les TC que les données de TM.
- **Protocole I2C** adopté pour les communications entre les sous-systèmes de l'architecture distribué du segment spatial.
- **Protocole KISS** adopté pour l'interface RS 232 qui peut servir aussi bien pour le débogage que pour la communication avec le TNC.

5.2.4. Couche N°1 : Couche physique

Les drivers de la couche physique permettent de prendre en charge tous les périphériques réseau utilisant des protocoles de communication I2C, CAN ou RS232. Tous les pilotes utilisent, dans la mesure du possible, l'accès direct à la mémoire DMA (Direct Memory Access) et un gestionnaire d'interruption permettant d'obtenir une bonne stabilité et de bonnes performances.

6. PROTOCOLES POUR LA COMMUNICATION RADIO

Le segment spatial est censé communiquer avec la station au sol dans la bande radio UHF que ce soit pour les liens montants ou descendants. Les protocoles AX25 et PUS (Packet Utilisation Standard) ont été adoptés pour cette communication radio afin de définir le format des TC à envoyer par la station sol, ainsi que celui des TM à recevoir.

6.1. Protocole AX25

Ce protocole spécifie les couches liaison et physique du protocole CSP pour les communications Radio. Il prend en charge le transfert des données de TM et les TC en mode connecté et non connecté. Il est caractérisé principalement par [98] :

- Format et codage des trames : HDLC (High-Level Data Link Control).
- Communication en mode connecté (Trames I) qui peut être utilisée quand la station au sol initie une connexion.
- Communication en mode non connecté (Trames UI) qui peut être utilisée pour le fonctionnement du segment spatial en mode balise (communication à sens unique sans accusé de réception).
- Détection d'erreur : 16 bits (CRC HDLC).
- Algorithme d'accès au support du type CSMA p-persistent.

La structure d'une trame AX25 est donnée par la *figure 6-9* suivante :

Flag	Adresses		Contrôle	PID	Informations	FCS	Flag
01111110	Destination Callsign	Source Callsign	0x03	0xF0	DATA	CRC-CCITT	01111110
1 Octet	7 Octets	7 Octets	1 Octet	1 Octet	≤ 256 Octets	2 Octets	1 Octet

Figure 6-9 : Trame du protocole AX25 [98]

- Champs "**Flag**" qui permettent de délimiter la trame. Deux trames peuvent partager un flag qui indique la fin de la première trame, et le début de la suivante.
- Champ "**Address**" qui contient les "**Callsigns**" (6 octets) de la source et de la destination de la trame. Chaque "Callsign" est complété, sur 1 octet, par son Secondary Station Identifier (SSID) qui désigne le type de station émettrice (caractère '0' pour une station fixe). Il est important de noter que le Callsign et le SSID ne sont pas envoyés en clair, mais chaque octet subit un décalage d'un bit vers la gauche.
- Champ "**Protocol Identifier**" (**PID**) qui identifie le type de protocole utilisé pour la couche 3 (Network Layer).
- Champ "**Informations**" (**DATA**) contient les données utiles qui sont dans notre cas une télécommande ou de la télémétrie qui peut être encapsulée dans le *protocole PUS*.
- Champ "**Frame-Check Sequence**" (**FCS**) qui permet de vérifier l'intégrité des données lors de leur réception.

6.2. Protocole PUS (Packet Utilisation Standard)

Le protocole PUS est l'un des standards ECSS et il constitue un complément des recommandations CCSDS (Consultative Committee for Space Data Systems) permettant de définir une interface au niveau de la couche application. Il est applicable à n'importe quel type de mission spatiale en offrant un ensemble de concepts opérationnels et de services qui s'adaptent avec le scénario des opérations :

- Distribution et vérification des TC.
- Transfert des données de TM.
- Gestion des logiciels embarqués.
- Surveillance et planification des opérations à bord du segment spatial.
- Déclaration des évènements à bord du segment spatial.

Le format général de la trame PUS se base sur le protocole SPP (Space Packet Protocol).

La *figure 6-10* représente les différents champs de la trame PUS [99] :

SPP Primary Header	PUS Data			
	PUS Secondary Header	Data	Spare	Error Control
6 Octets	Taille variable			

Figure 6-10 : Trame du protocole PUS selon la norme ECSS-E-70-41A [99]

Le protocole PUS doit être adapté aux exigences de la mission MASAT1 en conservant l'entête principale du protocole PPS (SPP Primary Header) et en modifiant la partie Data [99] [100] :

▪ **Primary Header SPP**

Il est identique qu'il s'agisse d'une télécommande ou d'une télémétrie :

SPP Primary Header						
Packet Identification				Sequence Control		Data Length
Version Number (=0)	Packet Type	Secondary Header Flag (=1)	APID	Sequence Flag	Sequence Count	
3 bits	1 bit	1 bit	11 bits	2 bits	14 bits	16 bits
2 Octets				2 Octets		2 Octets

Figure 6-11 : En tête principale de la trame du protocole PUS [99]

– **Packet Identification**

- **Version Number** : La version du protocole utilisée étant la version 1, la valeur de ce champ vaut par définition '000'.
- **Packet Type** : S'il s'agit d'une télécommande, ce bit vaudra '1', s'il s'agit d'une télémétrie, il vaudra '0'.
- **Secondary Header Flag** : Ce bit sert à indiquer la présence du Secondary Header.
- **Application Process Identifier (APID)** : utilisé pour identifier notre OdB et lui permettra de rejeter les TC qui ne lui sont pas destinés.

– **Sequence Control** : Ce champ permet de segmenter l'information en plusieurs trames.

– **Data Length** : Ce champ contient la taille en octets de la partie Data de la trame PUS (y compris le Secondary Header).

▪ **Secondary Header**

Il est différent selon qu'il s'agisse d'une télécommande ou d'une télémétrie :

⇒ **Télécommande :**

PUS Secondary Header (TC)						
CCSDS Secondary Header Flag (=0)	Version Number (=1)	ACSK	Service Type	Service Sub-type	Source ID (Optionnel)	Spare (Optionnel)
1 bit	3 bits	4 bits	8 bits	8 bits	n bits	n bits
1 Octet			1 Octet	1 Octet	n Octets	

Figure 6-12 : En tête secondaire de la trame du protocole PUS pour les TC [99]

- **CCSDS Flag** : Ce champ reste à '0' pour indiquer qu'il s'agit d'un Secondary Header non défini par la norme CCSDS.
- **Version Number** : La version actuelle étant 1, ce champ vaudra donc '001'.
- **Acknowledgement (ACK)** : Ce champ spécifie le type d'acquiescement que la station sol désire recevoir du satellite.
- **Service Type** : Permet de définir le type de service demandé. Les nombres compris entre 0 et 127 sont réservés pour les services standards, tandis que les nombres de 128 à 255 peuvent être utilisés pour un service spécifique à la mission.
- **Service Subtype** : Permet de définir un sous-type de service.
- **Source ID** : Permet d'identifier la source de la trame et peut être ignoré s'il n'existe qu'une seule source et aussi parce que la trame PUS est encapsulée dans une trame AX.25 où la source est déjà présente sous la forme d'un Callsign
- **Spare** : permet d'ajouter des bits afin d'obtenir un nombre entier d'octets.

⇒ **Télémetrie :**

PUS Secondary Header (TM)								
Spare	Version Number (=1)	Spare	Service Type	Service Sub-type	Packet Subcounter	Destination ID (Optionnel)	Time	Spare (Opti.)
1 bit	3 bits	4 bits	8 bits	8 bits	16 bits	n bits	n bits	n bits
1 Octet			1 Octet	1 Octet	2 Octets	n Octets		

Figure 6-13 : En tête secondaire de la trame du protocole PUS pour la TM [99]

- **Packet Subcounter** : Il s'agit d'un nombre incrémenté à chaque envoi d'une télémetrie d'un certain type et sous-type. Ainsi, si la station sol détecte qu'un compteur pour un certain type et sous-type saute une valeur, elle peut déterminer le type et le sous-type de la télémetrie manquante.
- **Time** : Contient la référence temporelle du satellite au moment de l'envoi de la télémetrie pour permettre à la station sol d'être synchrone avec le satellite.

7. SYSTEME D'EXPLOITATION TEMPS REEL FREERTOS

Le logiciel de vol embarqué de la mission MASAT1 présente les caractéristiques d'un système en temps réel, en raison du fait qu'il doit gérer divers événements externes et effectuer des tâches avec des contraintes temporelles. L'utilisation d'un système d'exploitation temps réel (RTOS) permet de répondre à cette exigence en donnant la possibilité d'avoir un code multitâches, modulaire et plus fiable afin de garantir le bon fonctionnement des tâches essentielles.

D'après le benchmark fonctionnel concernant les architectures des ODBs utilisés dans d'autres missions Cubesat (ANNEXE C), il y a quatre RTOS potentiellement utilisables et dont les caractéristiques sont résumées dans le *tableau 6-2* :

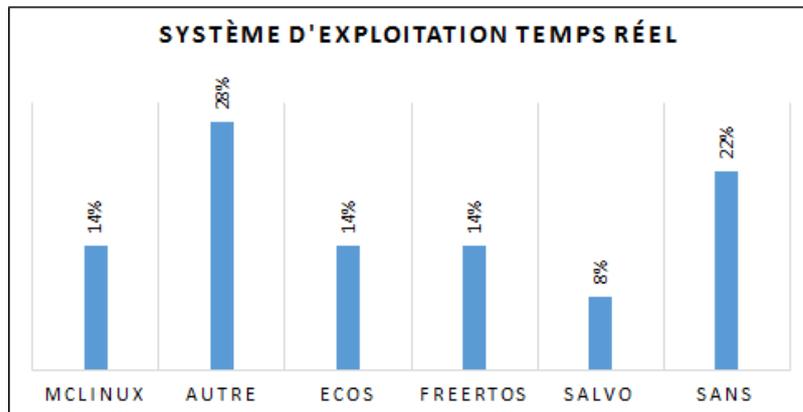


Figure 6-14 : Systèmes d'exploitation temps réel adoptées pour les missions Cubesat

RTOS	Processeurs supportés	Mémoire requises
 <ul style="list-style-type: none"> Free open source RTOS. Préemptif. POSIX compatible API. ISO C and math librairies. SPI, I2C, ADC drivers. TCP/IP networking stacks. 	<ul style="list-style-type: none"> 68K/ColdFire ARM (ARM7TDMI, ARM9TDMI, Cortex-M) CalmRISC16 and CalmRISC32 (RedBoot) Fujitsu FR-V et FR30 Hitachi H8/300 Intel x86 Matsushita AM3x PowerPC SPARC SuperH 	<p>Noyau, support de périphériques et mémoire : 10 à centaines de KO</p>
 <ul style="list-style-type: none"> Free open source RTOS. Préemptif. POSIX compatible API. Standard C libraries. TCP/IP networking stacks. 	<ul style="list-style-type: none"> Altera Nios/Nios II ADI Blackfin ARM (ARM7TDMI, ARM Cortex) Freescale m68k, Motorola 68000 Fujitsu FR-V Hitachi H8 Intel i960 Freescale and Motorola Coldfire Xilinx MicroBlaze 	<p>Noyau : Taille < 512 KO</p> <p>Noyau + Commandes Linux : Taille < 900 KO</p>
 <ul style="list-style-type: none"> Free open source RTOS. Préemptif POSIX compatible API. Standard C libraries. TCP/IP networking stacks. 	<ul style="list-style-type: none"> Altera Nios II ARM (ARM7, ARM9, Cortex) Atmel (AVR32, SAM3-4-7-9-D20) Xilinx (MicroBlaze, Zynq-7000) PIC microcontroller (PIC18, 24, 32) Intel (x86 – 8052) Texas Instruments MSP430 STMicroelectronics (STM32, STR7) 	<p>Entre 300 Octets et 9KO pour l'image binaire du noyau</p>
 <ul style="list-style-type: none"> Commercial RTOS. Coopératif POSIX compatible API. 	<ul style="list-style-type: none"> 8051 family ARM (ARM7TDMI, and CortexM3) Atmel (AVR, MegaAVR) Motorola M68HC11 TI (MSP430, DSP TMS320C2000) Microchip PIC12 14000 16 17 18 24 32 dsPIC 	<p>Typiquement 1KO pour l'image binaire du noyau</p>

Tableau 6-2 : Description des RTOS potentiellement utilisables pour MASAT1

Cette étude comparative préliminaire a permis d'écarter deux systèmes d'exploitation :

- Salvo parce qu'il est payant et non préemptif.
- μ CLinux parce qu'il a une empreinte mémoire beaucoup plus importante.

Nous avons finalement choisi le système d'exploitation temps réel FreeRTOS pour les raisons suivantes :

- Faciliter le développement du logiciel embarqué vue qu'une plateforme logicielle FreeRTOS est fournie avec la module OdB NanoMind A3200. Cette plateforme, basée sur Eclipse, comprend une bibliothèque avec les pilotes de périphériques, et les outils de débogage et de programmation.
- Le noyau de FreeRTOS est très léger (seulement 5 fichiers en langage C) et son image binaire ne dépasse pas 10KO [101].
- C'est un RTOS préemptif basé sur la priorité des tâches. Le système permet aussi la gestion des tâches de même priorité à l'aide de l'algorithme circulaire Round-Robin qui est adapté au système en temps partagé.

7.1. Noyau FreeRTOS

Le noyau FreeRTOS est indépendant du processeur cible. Il est distribué comme un module indépendant qui peut être ajouté à tout projet standard en utilisant l'assistant ASF (Atmel Software Framework) disponible sous Atmel Studio :

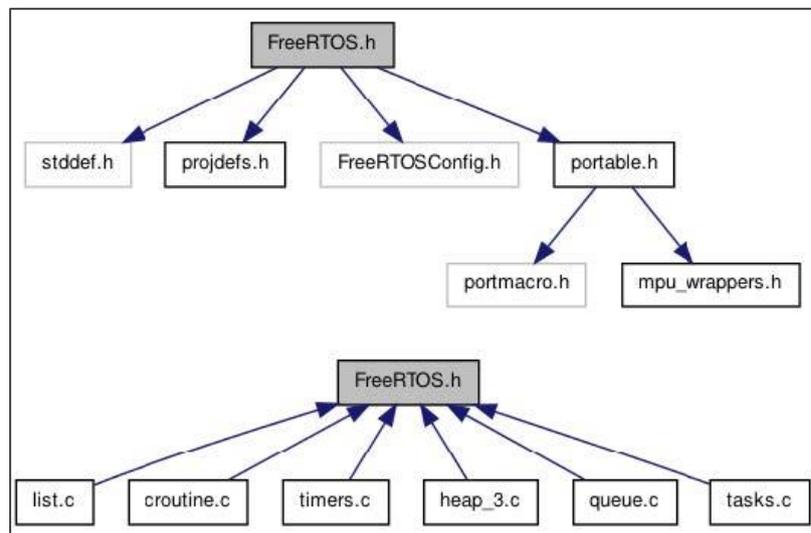


Figure 6-15 : Fichiers sources constituant le module FreeRTOS [101]

Les fonctionnalités standards incluses dans le module portable FreeRTOS sont :

- Affectation très flexible de la priorité d'une tâche.
- Temporisateurs logiciels.
- Files d'attente et Sémaphores (binaires – à compte – récursives - mutex)

La configuration de FreeRTOS se fait dans le fichier *FreeRTOSConfig.h*, et définit notamment :

- Les fonctionnalités utilisées (préemption, sémaphores, files d'attente, priorités dynamiques ou non, délais relatifs, délais absolus, etc.).
- Le nombre de niveau de priorités utilisées, et la plage réservée à l'exécutif de FreeRTOS et aux interruptions.
- La taille de pile minimale pour chaque tâche, la taille de la mémoire disponible pour le tas.
- La fréquence du tick système, qui représente la granularité des délais offerts au programmeur et la fréquence du changement de tâche active entre tâches de même priorité.

7.2. Mécanisme de gestion des tâches

FreeRTOS permet d'exécuter un nombre de tâches limité uniquement par les ressources matériels (mémoire) du système embarqué. Une tâche peut, au cours de son cycle de vie, se trouver dans un des quatre états résumés par la *figure 6-16* :

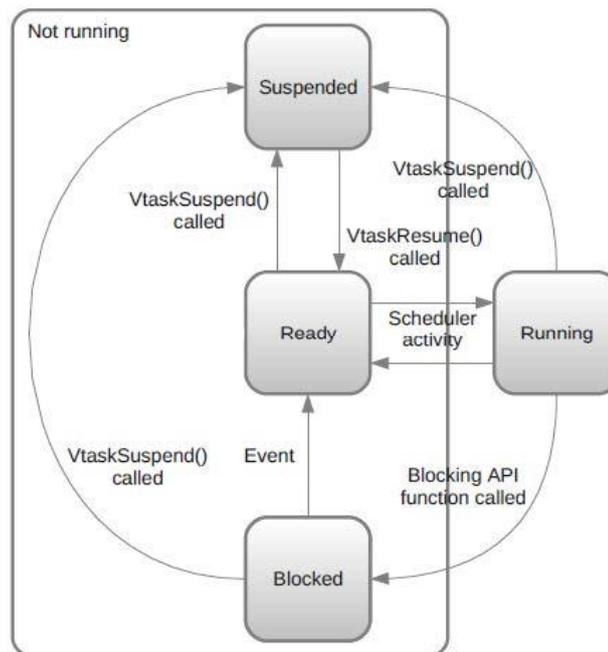


Figure 6-16 : Machine à états des tâches pour FreeRTOS [101]

Puisque le microcontrôleur doit toujours avoir quelque chose à exécuter, FreeRTOS créé une tâche de priorité minimale nommée *Idle* au lancement de l'ordonnanceur. Cette dernière utilise des fonctions de support comme la libération de l'espace occupé par des tâches supprimées, la mesure du taux d'utilisation du processeur ou la mise en veille du microcontrôleur lorsqu'aucune tâche applicative n'est pas en exécution.

7.3. Gestion de la mémoire

Dans un processus, toutes les tâches partagent un même espace mémoire d'adressage qu'on peut décomposer en trois parties principales :

- **Code** : contenant les instructions du programme [fonction `main()` et les tâches].
- **Data** : contenant les variables globales et les variables allouées dynamiquement.
- **Heap** : La pile, où les variables locales des tâches, ainsi que diverses informations temporaires (ayant une durée de vie égale à la tâche) sont stockées.

Le noyau RTOS alloue dynamiquement de la mémoire RAM (pile mémoire) à chaque fois qu'une tâche ou un objet du noyau (sémaphore ou file d'attente) est créé. La taille de cette pile est configurable lors de la création de la tâche et elle réserve deux espaces mémoires libres pour chaque tâche :

- Une structure de donnée **TCB (Task-Control Block)** permettant au noyau de gérer la tâche.
- Un fichier de la tâche (**Task file**) contenant les variables locales et temporaires de la tâche.

La constante `configTOTAL_HEAP_SIZE`, dans le fichier `FreeRTOSConfig.h`, définit la quantité de mémoire disponible pour la zone Heap. La gestion de cette zone peut être réalisée de trois manières différentes :

- **1ère implémentation** : valable pour un petit système embarqué, elle ne permet pas de libérer la mémoire une fois attribuée à une tâche ou à un objet du noyau.

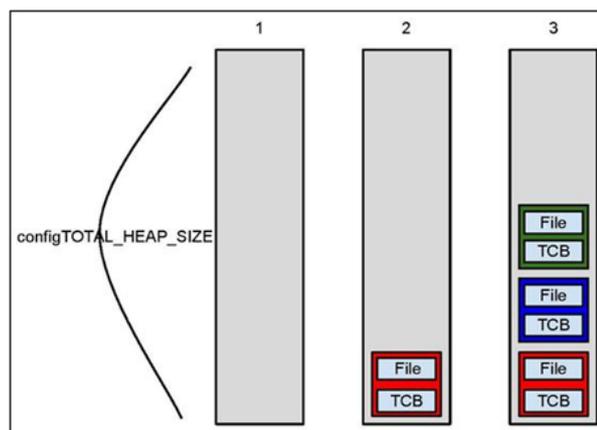


Figure 6-17 : 1^{ère} implémentation de la gestion mémoire dans FreeRTOS [101]

- **2ème implémentation** : dispose d'une méthode permettant de libérer la mémoire allouée pour une tâche. On peut ensuite l'allouer de nouveau à une autre tâche à condition que la taille du bloc mémoire de la nouvelle tâche soit au maximum égale à la taille du bloc de l'ancienne tâche.

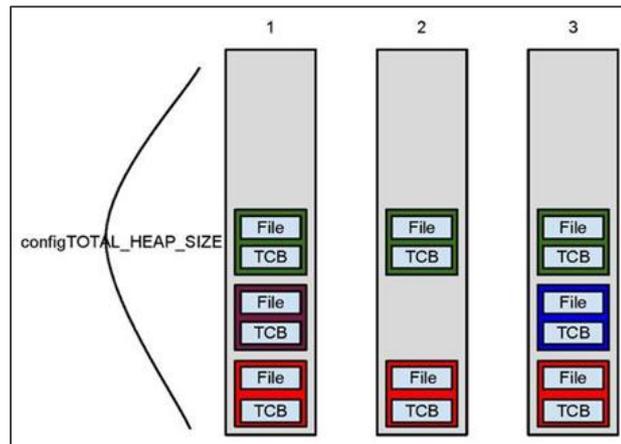


Figure 6-18 : 2^{ème} implémentation de la gestion mémoire dans FreeRTOS [101]

- **3^{ème} implémentation** : Il s'agit de la simple mise en œuvre d'une interface pour les bibliothèques C standard malloc() et free() qui sont généralement fournis avec le compilateur utilisé. L'interface ajoute de la sécurité en suspendant toutes les tâches pendant la durée des opérations sur la mémoire.

8. IDENTIFICATION DES FONCTIONNALITES LOGICIELLES DE L'ODB

L'adoption d'une architecture distribuée utilisant le bus I2C comme protocole de communication, signifie que chaque sous-système du segment spatial contiendra un logiciel embarqué lui permettant d'exécuter ces tâches d'une manière autonome. En mode de fonctionnement nominal, l'ODB sera en charge de contrôler le bus de communication, de superviser le fonctionnement de l'ensemble des sous-systèmes en collectant les informations de débogage et les données de télémétrie et d'exécuter les télécommandes provenant de la station au sol.

A ce stade de la conception du segment spatial, les fonctionnalités logicielles ont été définies de façon abstraite pour permettre une flexibilité en fonction des changements qui seront effectués dans les étapes suivantes du projet. Ces fonctionnalités sont listées ci-dessous :

- Effectuer les opérations initiales du satellite selon une séquence prédéfinie (déploiement antenne, la première activation des autres sous-systèmes, etc.).
- Fournir la base de temps du segment spatial.
- Contrôler le bus de communication I2C (en mode nominal) pour assurer l'échange de données et l'interfaçage entre les sous-systèmes du segment spatial.
- Superviser le fonctionnement de tous les sous-systèmes du segment spatial en recevant des informations de débogage de chacun d'eux et en tenant un journal des événements significatifs qui se déroulent à bord du segment spatial.

- Superviser, acquérir et stocker les mesures de télémétrie réalisées à bord du segment spatial entre les séquences de communication avec la station au sol.
- Répondre, pendant les séquences de communication, aux demandes provenant de la station au sol et qui peuvent concerner les données de télémétrie ou les informations du journal d'évènements à bord.
- Gérer les TC qui proviennent de la station sol et stocker celles à exécution différée.
- Gérer l'énergie électrique (mode nominal) en activant ou en désactivant les sous-systèmes selon les scénarios opérationnels et selon des conditions prédéfinies concernant la disponibilité de l'énergie électrique.

9. ORGANISATION LOGICIELLE EN COUCHES ET MODULES

Le logiciel embarqué de l'OdB est codé en langage C, et exécutée au sein du système d'exploitation temps réel FreeRTOS. Il est ainsi divisé horizontalement en 3 couches principales, comme le montre la *figure 6-19* :

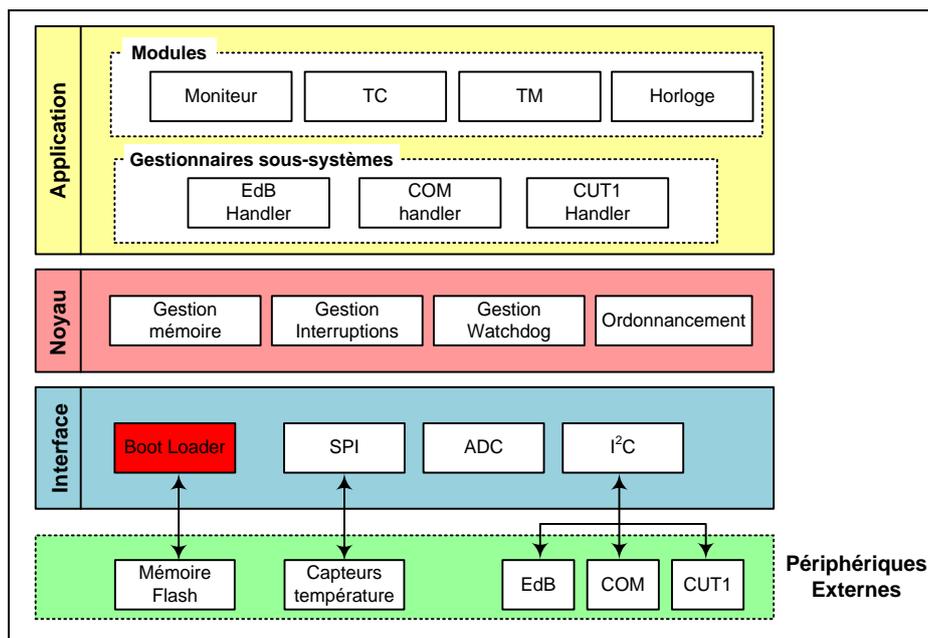


Figure 6-19 : Organisation du logiciel embarqué dans l'OdB de MASAT1

- Une couche supérieur appelée *Couche Application* qui dépendra uniquement des fonctionnalités logicielles en relation avec la mission comme la gestion de l'énergie électrique, la gestion des données de télémétrie et la supervision du système.
- Une couche inférieure appelée *Couche Interface* qui contiendra les drivers qui servent comme couche d'abstraction entre les applications et le matériel. Les drivers se composent de fonctions pouvant être appelées par n'importe quel module de la couche Application pour interagir avec le matériel.

- Une *couche intermédiaire appelé Noyau* qui fournira les services communs du RTOS permettant entre autres la gestion des tâches, des interruptions et de la mémoire commune.

Vu que les solutions technologiques adoptées pour la plateforme matérielle de MASAT1 sont homogènes, cette organisation logicielle en couche permettra la réutilisation d'un maximum de code par les différents sous-systèmes du segment spatial comme les services communs du noyau ou les drivers matériels de la couche Interface.

Le "Bootloader" présent dans la couche interface représente du soft placé physiquement dans la mémoire externe Flash et généralement à l'adresse physique 0x00. Il contient toutes les instructions nécessaires à l'initialisation du processeur et le démarrage du système d'exploitation FreeRTOS ainsi que les couches logicielles construites au-dessus. Cela inclut dans la plupart des cas : une copie du logiciel embarqué de la mémoire Flash vers la RAM, une vérification de la somme de contrôle de l'image de la mémoire, et le démarrage du logiciel embarqué à une adresse spécifiée de la RAM (habituellement à l'adresse 0x00).

La couche Application est constituée de :

- *Handlers* permettant la gestion des sous-systèmes du segment spatial en mode Nominal.
- *Modules* permettant de réaliser les fonctionnalités logicielles de l'ODB. Chaque module sera constitué de plusieurs tâches.

La description des tâches réalisées par chacun des modules de la couche application est proposée dans le *tableau 6-3* suivant :

MODULES	TACHES
Moniteur	Déploiement Antennes Gestion Energie Gestion Erreurs Supervisions sous-systèmes Mise à jour journal évènement
TC	Gestion TC Exécution TC
TM	Prise Mesures Formatage Mesures Stockage Mesures
Horloge	Fournir Base de temps

Tableau 6-3 : Tâches réalisées par les modules de la couche application

Chapitre 7

ETUDE ET CONCEPTION DE L'OdB SECONDAIRE RECONFIGURABLE A BASE DE FPGA

1. DESCRIPTION DE LA CHARGE UTILE DE LA MISSION MASAT1

1.1. Motivation

Comme on l'a vu dans l'introduction générale, le marché des nanosatellites a connu une grande croissance ces dernières années avec l'utilisation de la norme CubeSat :

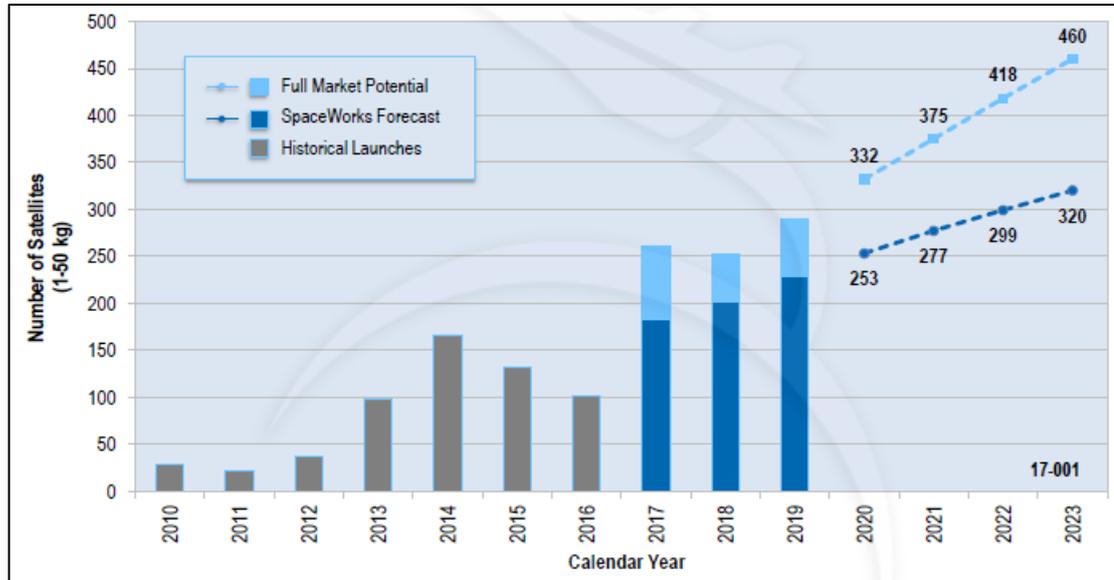


Figure 7-1 : Historique et prédictions des lancements de nano/microsatellites [13]

Cet intérêt croissant pour le développement de missions spatiales type nanosatellite exige de nouvelles approches pour trouver une alternative peu coûteuse, flexible, reconfigurable et fiable, à l'architecture matérielle traditionnelle des OdBs basée sur les microcontrôleurs. L'une des solutions de conception les plus adaptées à ces exigences consiste à adopter une approche System-on-Chip (SoC) hautement intégrée et reconfigurable, se basant sur la technologie des FPGA type SRAM [102].

La charge utile de la mission MASAT1 consiste à concevoir un ordinateur de bord secondaire exploitant les fonctionnalités de reconfiguration Xilinx du FPGA Spartan 6. La finalité étant de :

- Démontrer la faisabilité d'utiliser la technologie FPGA type SRAM comme un moyen viable permettant de diminuer les temps de conception, d'augmenter la capacité de calcul, et de permettre une reconfiguration flexible en orbite des fonctions matérielles et/ou logicielles de la plateforme embarquée pour, par exemple, prendre en charge des exigences de mission dynamiques ou mettre à jour la configuration du FPGA afin de corriger des erreurs de conception.

- Tester et évaluer, en orbite, des techniques d'atténuation pour assurer une bonne tolérance aux pannes des FPGA type SRAM. En effet, les radiations spatiales et leur effet sur ce type de FPGA constituent le principal défi au déploiement des systèmes reconfigurables dans l'espace, et ils nécessitent la mise en œuvre de techniques de détection, d'isolation et de récupération de défaillance (DIRD). La DIRD doit être gérée de façon autonome et le **tableau 7-1** reprend les niveaux de cette autonomie selon la norme ECSS-E-ST-70C [103] :

Niveau	Description	Fonctions
D1 (bas)	Établir une configuration sécurisée du segment spatial suite à une défaillance à bord.	<ul style="list-style-type: none"> ▪ Identifier et signaler les anomalies. ▪ Reconfigurez le système embarqué pour isoler la défaillance. ▪ Placez le segment spatial dans un état sécurisé.
D2 (haut)	Rétablir les opérations nominales de la mission nominales suite à une défaillance à bord.	<ul style="list-style-type: none"> ▪ Reconfigurer le système embarqué pour un mode opérationnel nominale. ▪ Reprise de l'exécution des opérations nominales.

Tableau 7-1 : Niveau d'autonomie de la DIRD selon la norme ECSS-E-ST-70C [103]

1.2. Exigences de l'OdB secondaire de la mission MASAT1

L'OdB secondaire, comme charge utile de la mission MASAT1, doit satisfaire aux exigences préliminaires spécifiées dans le DES [71] :

[DES] FUNC-1600

L'OdB secondaire sera en charge de la supervision partielle, la commande et le contrôle de certaines opérations du nanosatellite.

[DES] FUNC-1610

L'OdB secondaire doit fournir certaines fonctionnalités nécessaires pour l'acquisition, le stockage, le formatage et la transmission des informations de TM vers le segment au sol.

[DES] FUNC-1620

L'OdB secondaire doit fournir toutes les fonctionnalités nécessaires pour la validation, le décodage et le traitement des TC qui peuvent être à exécution directe ou reportée.

[DES] FUNC-1630

L'OdB secondaire doit être pourvu d'une DIRD autonome et supervisée par l'OdB.

[DES] OPER-1640

L'OdB secondaire doit fonctionner en mode autonome avec une intervention minimale de l'OdB et du segment au sol.

[DES] OPER-1650

L'autonomie des opérations de l'OdB secondaire doit être assurée par les logiciels embarqués et non par les commandes stockées en mémoire et qui peuvent être corrompues.

[DES] OPER-1660

L'OdB secondaire doit considérer les TC provenant de la station au sol comme prioritaires par rapport aux fonctions d'autonomie des logiciels embarqués.

[DES] OPER-1670

L'OdB secondaire doit rapporter toute reconfiguration matérielle ou logicielle autonome dans la TM.

[DES] CONC-1680

L'architecture de l'OdB secondaire doit être centralisée en suivant l'approche du System-on-Chip (SoC) et en exploitant les capacités des nouvelles technologies hautes densité des FPGA.

[DES] CONC-1690

La conception de l'OdB secondaire doit se faire à base de composants COTS (Commercial Off The Shelf) et fournir un degré élevé de tolérance de pannes.

[DES] CONC-1700

La conception de l'OdB secondaire doit permettre, en orbite et sur commande, une reconfiguration matérielle totale ou partielle du FPGA pour une mise à jour ou une correction d'une défaillance.

[DES] CONC-1710

Il doit être possible de copier le contenu de la configuration par défaut et celui des mémoires de travail vers le segment au sol.

[DES] CONC-1720

L'OdB secondaire doit fournir suffisamment de capacité mémoire pour le stockage des données de configuration, les données générées à bord et qui peuvent concerner la TM, le traitement embarqué et les instruments de mesures.

[DES] CONC-1730

Les logiciels embarqués dans l'OdB secondaire doivent permettre le contrôle en temps réel.

[DES] CONC-1740

Tout logiciel embarqué doit être identifié.

[DES] CONC-1750

La modification ou la mise à jour des logiciels embarqués doit être possible en orbite.

[DES] CONC-1760

L'OdB secondaire doit implémenter un système matériel de surveillance indépendant du processeur (comme un chien de garde) pour interrompre chaque tâche ou fonction embarquée bloquée ou tournant en boucle infinie.

[DES] CONC-1770

La conception de l'OdB secondaire doit réduire au maximum le nombre d'interface.

2. ETAT DE L'ART DES ARCHITECTURES RECONFIGURABLES

2.1. Introduction aux FPGAs

Vu que le FPGA est le plus dense, le plus avancé et le plus flexible des circuits logiques programmable PLD (Programmable Logic Device), son utilisation permet de concevoir des systèmes performant avec un faible coût de développement, et une grande flexibilité en termes de conception [104].

L'architecture de base d'un FPGA est généralement composée de deux couches. Comme le montre la *figure 7-2*, la première couche est constituée d'un ensemble de blocs logiques configurables (CLB) qui peuvent être interconnectés les uns aux autres ainsi qu'aux blocs d'E/S programmables grâce à une sorte d'architecture de routage programmable. Chaque CLB est composé de blocs générateurs de fonctions logiques ou LUT (Look-Up Table (LUT)), de multiplexeurs (MUX), et de bascules flip-flops et de registres pour la mémorisation. Cette première couche contient aussi des blocs mémoire configurables nommées BRAMs et des blocs embarqués EAB (Embedded Array Bloc) pouvant implémenter des fonctions complexes comme des multiplicateurs, des DSP (Digital Signal Processor) ou des processeurs embarqués.

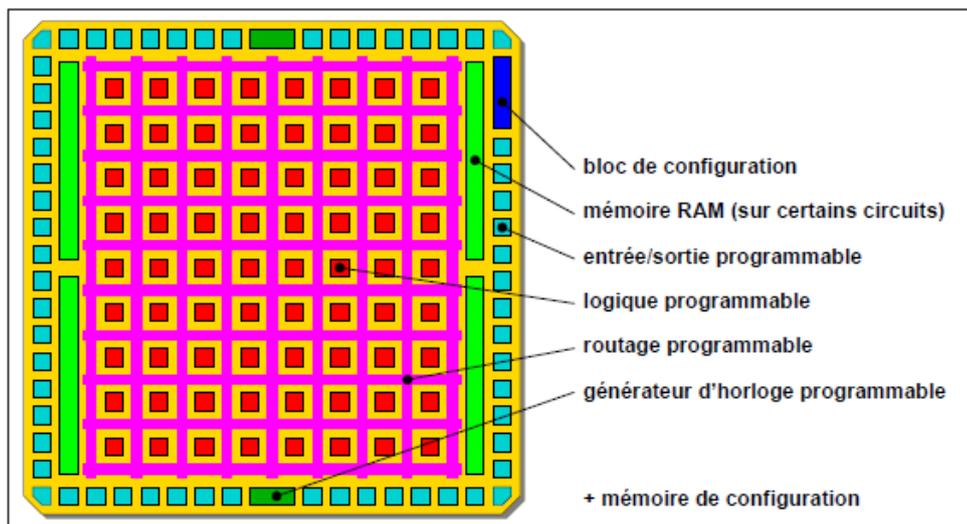


Figure 7-2 : Architecture générale d'un circuit FPGA [105]

La deuxième couche de l'architecture d'un FPGA est constituée de la mémoire de configuration permettant de décrire l'architecture du système se trouvant sur la première couche. Le contenu des CLB est ainsi programmé (configuré) programmé pour contrôler la fonctionnalité des blocs logiques, tandis que les commutateurs de routage (boîtes de commutation) sont programmés pour réaliser les connexions souhaitées entre les blocs logiques.

Pour écrire dans la mémoire de configuration, on utilise des fichiers binaires appelés "bitstreams" qui contiennent les informations de contrôle pour la configuration ainsi que des données de configuration.

2.2. Technologies des FPGAs

Il existe trois grandes catégories de technologies FPGA qui diffèrent essentiellement dans la méthode utilisée pour stocker les informations du fichier de configuration :

- **FPGA à anti-fusibles** qui sont programmables une seule fois. Les anti-fusibles sont des circuits ouverts jusqu'à ce qu'un courant soit appliqué à travers. La **figure 7-3** montre un exemple de structure anti-fusible nommée PLICE (Programmable Low Impedance Circuit Element) du fabricant Actel. Il est constitué de trois couches : deux conducteurs et un isolant diélectrique.

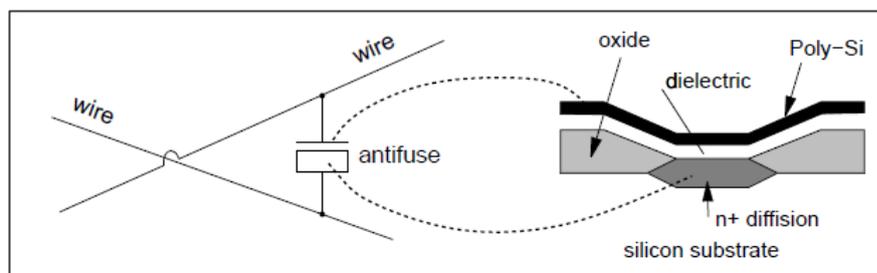


Figure 7-3 : Gestionnaire d'horloge d'un FPGA [106]

Dans un état non programmé, le diélectrique isole les deux couches conductrices, alors que lors d'une programmation, l'application d'une tension élevée fait claquer le diélectrique créant une liaison entre les deux couches conductrices.

Le principal avantage des FPGA anti-fusibles est leur faible sensibilité aux radiations induisant une bonne fiabilité. Par contre, leur principal inconvénient est l'impossibilité de les reconfigurer.

- **FPGA à mémoire SRAM** qui utilisent une mémoire statique à accès aléatoire SRAM (Static Random Access Memory) pour stocker les données de configuration. Comme le montre la **figure 3-4**, on utilise classiquement une cellule 5T-SRAM constituée de deux inverseurs montés tête-bêche pour stocker le bit, d'un transistor d'accès (Write Port) et d'un transistor de remise à zéro lors de la mise sous tension.

Un exemple d'utilisation de la structure de programmation à base de point mémoire SRAM est présenté dans la **figure 7-5** où ces cellules SRAM sont utilisées pour commander des grilles de transistors de réseau d'interconnexion ou pour contrôler des multiplexeurs connectés à des blocs de logique.

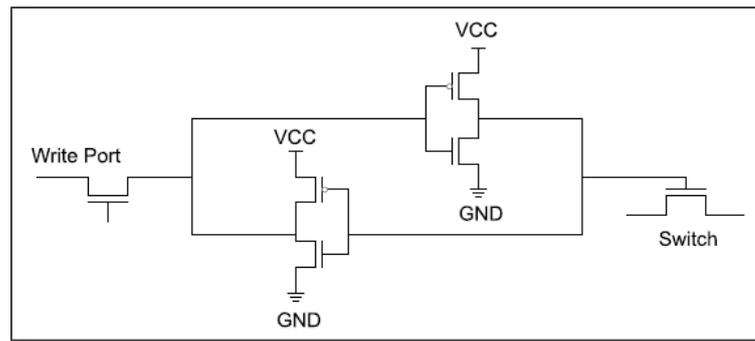


Figure 7-4 : Schéma d'une Cellule SRAM

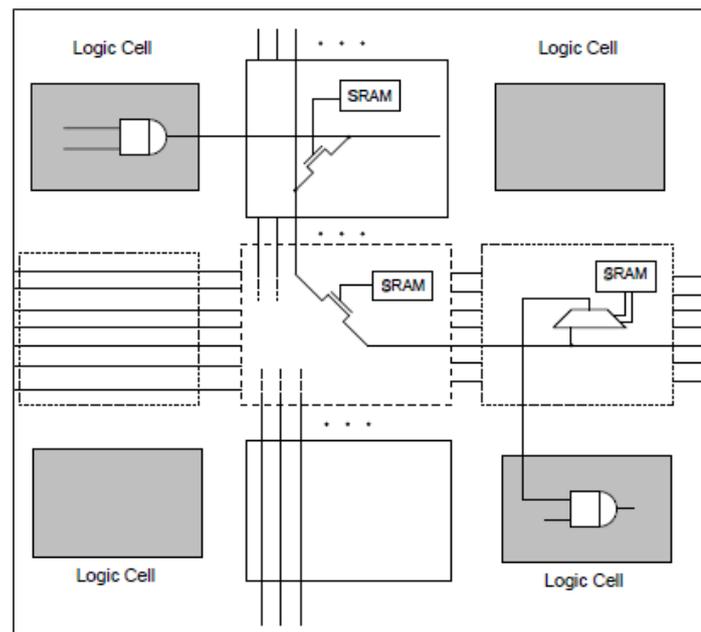


Figure 7-5 : Structure de programmation à base de point mémoire SRAM [106]

Les principaux avantages des FPGA SRAM sont un temps de configuration réduit, un faible coût et une grande flexibilité puisqu'ils permettent une reconfiguration partielle ou totale de leur architecture. Cependant, à cause de leur méthode de stockage volatile des données de configuration, ils sont plus sensibles aux radiations et ils nécessitent une mémoire externe additionnelle non-volatile pour stocker les données de configuration et les charger sur SRAM à chaque mise sous tension.

- **FPGA à mémoire Flash** qui combinent les deux avantages majeurs des FPGA SRAM et anti-fusible, à savoir : possibilité de reprogrammation et non volatile. La structure d'un point mémoire flash est ainsi programmée à partir des techniques des mémoires EPROM (Erasable Programmable ROM) et effacées avec le mécanisme des mémoire EEPROM (Electrically-Erasable Programmable ROM).

Comme le montre la **figure 7-6**, une cellule mémoire flash utilise une charge stockée sur la grille flottante d'un transistor pour réaliser un stockage non volatile :

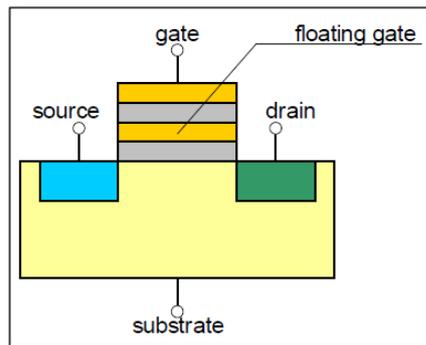


Figure 7-6 : Transistor à grille flottante utilisé dans la mémoire flash

Les principaux avantages de cette technologie Flash est qu'elle est plus résistante aux radiations que la technologie SRAM et elle bénéficie d'un temps de configuration relativement plus court et d'une consommation d'énergie plus faible. Par contre, cette technologie est plus coûteuse, en plus du fait d'avoir une reconfiguration peu flexible vu qu'elle nécessite une alimentation supplémentaire élevée servant à la programmation et à l'effacement de la mémoire.

2.3. Utilisation des FPGAs dans les missions spatiales

En raison des caractéristiques extrêmement attrayantes de la technologie FPGA pour les applications spatiales, la communauté spatiale a activement évalué les effets des radiations pour différents FPGA produits par des fabricants leaders tels que Xilinx, Altera ou Actel [121-123].

Depuis la publication en 1992, par le laboratoire JPL (Jet Propulsion Laboratory) de la NASA, d'un rapport recommandant l'utilisation de plusieurs FPGA anti-fusibles pour les applications spatiales [109], l'implication des FPGA dans les missions spatiales s'est limitée à ce type de FPGA comme pour les Rovers envoyés par la NASA sur Mars [110].

Cependant au cours de ces dernières années, plusieurs missions spatiales ont démontré avec succès l'utilisation de FPGA SRAM, en exploitant leur flexibilité de reconfiguration tout en mettant en œuvre différentes techniques DIRD pour compenser la vulnérabilité de ce type de FPGA aux radiations [111 - 113].

La charge utile scientifique de MASAT1 vise à exploiter le retour sur expérience de ces différentes missions spatiales pour explorer la faisabilité de l'utilisation des FPGA SRAM dans des nanosatellites, avec les défis de la fiabilité et de la puissance électrique limitée.

2.4. Types de défaillances dans les architectures reconfigurables à base de FPGA

L'effet d'une exposition prolongée aux radiations spatiales dépend d'un critère appelé LET (Linear Energy Transfert) définie comme l'énergie transférée par une particule ionisante traversant la matière, par unité de distance et pour une densité de matériaux spécifique (Mev/cm ou MeV·cm²/mg) [114]. Par exemple dans le silicium, une particule ayant un LET de 97 MeV.cm²/mg dépose une charge d'environ 1 pC par micron.

Les dommages causés par ces radiations dépendent de la quantité et de l'emplacement de l'énergie déposée par les particules chargées dans le circuit FPGA. On peut ainsi distinguer deux grandes catégories principales d'effets :

- Les évènements singuliers dont la classification est résumée par la *figure 7-7*.
- Dégradation cumulative à long terme TID qui peut engendrer des défaillances permanentes.

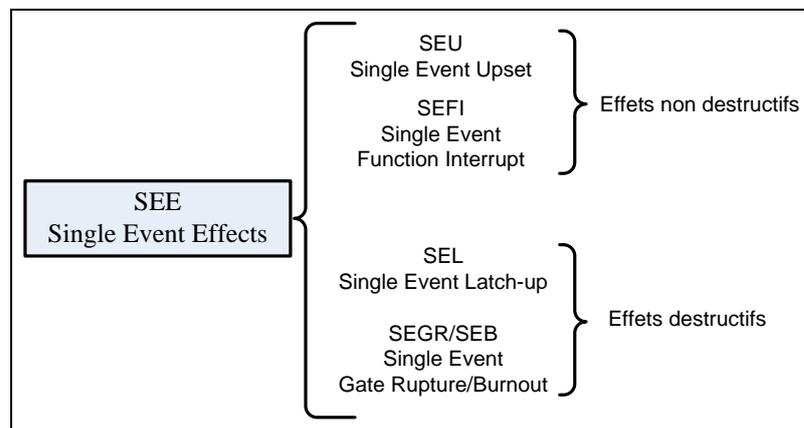


Figure 7-7 : Classification des événements singuliers SEE

2.4.1. Les évènements singuliers

Lorsque la charge, d'un ion lourd qui a traversé la matière, est déposée dans une zone active d'un transistor, elle est collectée par le champ électrique et peut se propager dans le circuit intégré pouvant ainsi provoquer différents effets parasites regroupés sous l'acronyme SEE (Single Event Effects). Ceux-ci peuvent être classés en trois catégories :

- **SEU (Single Effect Upset)** : effet logiciel pouvant affecter la mémoire de configuration d'un FPGA type SRAM, en inversant l'état d'un bit entraînant ainsi la corruption des fonctionnalités de la conception matérielle du FPGA. En effet, comme le montre la *figure 7-8*, les cellules SRAM définissent les états des LUT, des MUX, des bascules Filp-flop et du réseau de commutateurs entre CLB.

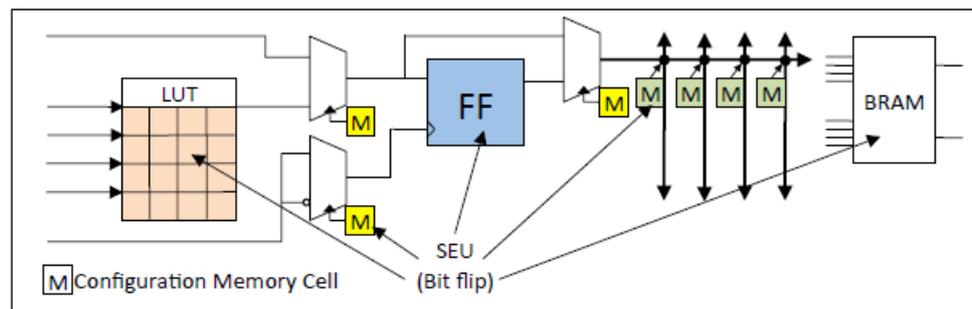


Figure 7-8 : Evènements SEU dans la mémoire de configuration d'un FPGA Xilinx

La mémoire de configuration FPGA étant composée de ces cellules SRAM, lorsqu'une erreur produite par un SEU se produit, elle peut modifier la fonctionnalité du circuit, comme par exemple :

- Modifier le contenu d'un bloc LUT et changer ainsi sa fonction combinatoire.
 - Modifier la configuration d'un bloc E/S en inversant la direction des E/S.
 - Modifier les réseaux de connexion en provoquant des circuits ouverts ou des courts-circuits.
- **SEFI (Single Event Functional Interrupt)** : c'est une anomalie similaire à un événement SEU, mais qui produit un changement temporaire de la fonctionnalité implémentée et qui peut être corrigé par une réinitialisation du système.
 - **SEL (Single Effect latchUp)** : effet destructeur qui entraîne des courts-circuits (qlq ps) entre l'alimentation et la masse du circuit intégré. Cet effet est dû à une énergie transférée suffisante pour activer en activant le transistor parasite présent entre les jonctions N-P des transistors CMOS. Ces effets peuvent être limités en surveillant l'alimentation des puces et prévoyant une mise hors tension rapide en cas de détection de SEL.
 - **SEGR/SEB (Single Event Gate Rupture/Burnout)** : effet se référant à la destruction des transistor MOS, causée par un ion qui traverse la structure du transistor, pour endommager sa grille (augmentation du courant de fuite). Ce phénomène entraîne la destruction du circuit si une énergie de court-circuit suffisante est disponible.

2.4.2. Effets par dose cumulée TID

Il s'agit de l'accumulation, pendant le temps de mission, de charges au niveau des oxydes isolants dans les circuits intégrés. Ces charges cumulatives sont imputables principalement aux particules piégées dans les ceintures de radiations et aux protons issus de l'activité solaire.

L'unité permettant de mesurer la dose de rayonnement ou l'énergie déposée est le "rad" qui est défini comme $100 \text{ rad} = 1 \text{ Gray (Gy)}$, avec le Gy qui est l'unité SI définie par la quantité de 1 Joule d'énergie déposée par kilogramme.

Pour les technologies MOS, cela se traduit par une dérive des paramètres électriques comme par exemple le décalage des niveaux de seuils, l'augmentation des courants de fuite [115]. Les charges étant cumulatives et leur effet essentiellement permanent, ils conduisent à une perte progressive puis totale de la fonctionnalité du dispositif CMOS.

2.5. Moyens de prévention et d'atténuation des effets des radiations

Pour assurer et maintenir le bon fonctionnement des architectures reconfigurables dans un milieu radiatif comme l'espace, plusieurs approches de prévention et d'atténuation des effets des radiations peuvent être appliquées soit au niveau du système, soit au niveau de son architecture.

2.5.1. Approches de préventions

La prévention contre les effets des radiations se réfère à un système ou un circuit conçu pour éviter les occurrences de défaillances. Chacune des techniques décrites, tout en réduisant la probabilité d'une défaillance induite par un rayonnement, va augmenter la complexité du système ou du circuit, ainsi que le coût.

a. Blindage

Le blindage a pour but de protéger les circuits électroniques en utilisant des matériaux impénétrables ou partiellement pénétrables par les rayonnements. Il est utilisé à la fois :

- Au niveau du circuit sous forme de feuilles métalliques légères incorporées au boîtier. Ce type de blindage est efficace contre les particules alpha et les protons à faible énergie [116].
- Au niveau du système en utilisant, pour les missions Cubesats, des feuilles d'aluminium de quelques millimètres d'épaisseurs pour réduire le flux et l'énergie des particules à l'intérieur du segment spatial [112].

En règle générale, le dimensionnement du blindage doit prendre en considération le profil, la durée de vie et le budget de la mission. Une évaluation des effets du blindage au niveau du segment spatial MASAT1 a été réalisée dans l'étude orbitale du chapitre 1 (§ 4.2.1) et les résultats sont résumés à la *figure 1-38*.

b. Resistance aux radiations

La résistance aux radiations se réfère à l'amélioration de la tolérance des circuits microélectroniques à divers types de rayonnements [64]. Les composants résistants aux radiations, qui sont plus coûteux et spécialement conçus pour les applications spatiales, se réfèrent à des circuits qui peuvent tolérer 200 krad ou plus et sont immunisés contre les événements SEL. Cette solution a été écartée pour la mission MASAT1 pour des raisons de budget et d'énergie disponible limitée.

2.5.2. Approches d'atténuation

L'incapacité de garantir l'évitement des défaillances dans la conception du système exige la mise en place de méthodes d'atténuation et de récupération au niveau de l'architecture du système, pour réduire l'impact des défaillances non fatales.

a. Redondance matérielle

En utilisant la redondance matérielle, les blocs cibles sont répliqués afin que le système puisse auto surveiller son état de fonctionnement. Généralement, il existe deux approches :

- **Duplication avec comparaison** où les ressources matérielles sont doublées et les résultats produits par les modules d'origine et les modules répliqués sont comparés pour détecter une défaillance, sans pouvoir identifier le module défectueux (voir *figure 7-9*). Cette approche permet de tolérer les défauts temporaires qui déclenchent quelques tentatives pour répéter la dernière opération. En cas d'échec, une défaillance permanente est déclarée.

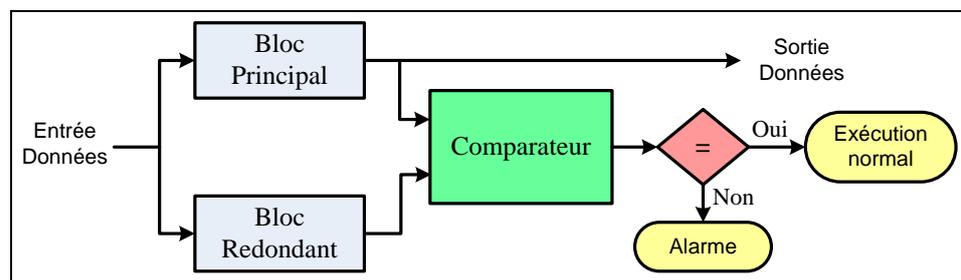


Figure 7-9 : Duplication avec comparaison

On distingue alors l'approche "Lockstep" qui est la mise en œuvre de la duplication avec comparaison au niveau du processeur implémenté dans un FPGA. Elle est prise en charge par certains FPGA Xilinx [117] : deux processeurs identiques reçoivent les mêmes entrées, exécutent simultanément les mêmes instructions, et leurs résultats sont comparés étape par étape à

chaque cycle d'horloge. Ce système est capable de détecter, mais non de corriger l'erreur, car il ne peut pas indiquer le processeur défectueux. En cas d'erreur, l'ensemble du système doit être reconfiguré pour récupérer les fonctionnalités correctes des deux processeurs.

- **Triplification modulaire TMR** (Triple Modular Redundancy) où les ressources matérielles sont triplées, avec une logique de vote ou "voteur qui compare les sorties des modules répliqués (voir *figure 7-10*) :

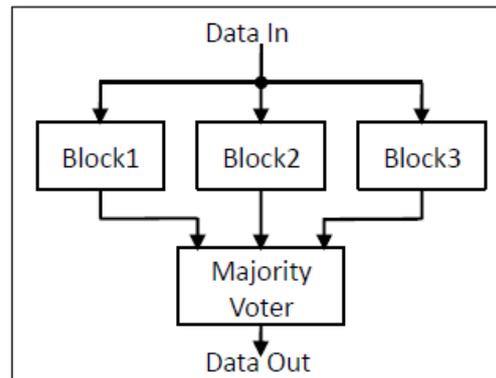


Figure 7-10 : Triplification modulaire TMR

Tous les modules fonctionnent simultanément et la sortie du voteur sera correcte si au moins deux des modules sont non défectueux comme le montre l'exemple de logique de vote de la *figure 7-11* [119]. Cette approche de redondance statique est caractérisée pour être simple, mais elle a un fort surcoût en surface et en consommation, variant entre 300 et 350% par rapport à une conception non-redondante [118].

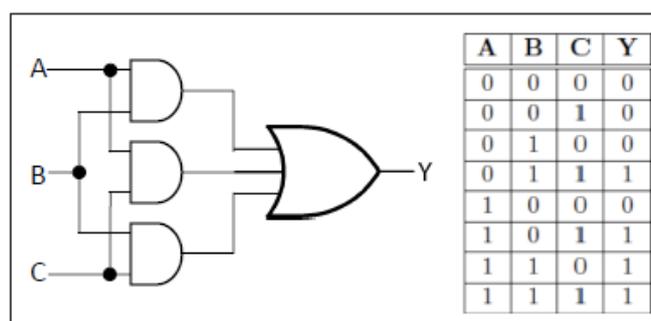


Figure 7-11 : Voteur de la majorité et sa table de vérité [119]

Le point faible de cette méthode c'est le voteur qui, en cas de défaillance, entraîne la défaillance de toute l'architecture TMR.

b. Codes de détection et de correction des erreurs

Les codes correcteurs d'erreurs EDAC (Error Detection And Correction) peuvent représenter une solution moins coûteuse complémentaire aux techniques de redondance. Ils sont couramment utilisés pour protéger les mémoires. Un code EDAC est caractérisé par son nombre de bits supplémentaires, le matériel supplémentaire pour le codage et le décodage et enfin son temps de latence. Par ailleurs, il possède deux paramètres clés à savoir le nombre de bits erronés qui peuvent être détectés et le nombre de bits erronés qui peuvent être corrigés. Le *tableau 7-2* résume les méthodes EDAC couramment utilisées [112] [115] :

Méthode	Description
Contrôle de redondance cyclique	<ul style="list-style-type: none"> ▪ C'est un moyen de contrôle d'intégrité des données puissant et facile à mettre en œuvre avec des polynôme générateurs. ▪ Il représente la principale méthode de détection des erreurs de transmission et de transfert des données.
Codes de hamming	<ul style="list-style-type: none"> ▪ Ce sont les premiers codes linéaires capables de corriger une seule erreur et d'en détecter deux (SEC-DED : Single Error Correction-Double Error Detection). ▪ Ils sont très utiles dans les cas où la probabilité d'erreurs multiples dans une structure de données unique, est faible.
Codes Reed-Solomon	<ul style="list-style-type: none"> ▪ Parmi les codes cycliques les plus puissants pouvant être construits en utilisant des polynômes générateurs appropriés. ▪ Ils sont particulièrement adaptés aux applications où les erreurs se produisent en rafales, mais ils nécessitent un circuit de codage et de décodage complexe.

Tableau 7-2 : Méthodes de correction et de détection des erreurs

c. Le chien de garde

Le chien de garde, ou "watchdog" en anglais, est un mécanisme de surveillance qui peut matériel ou logiciel. Il peut être implanté à différents niveaux (composants, sous-systèmes ou cartes électroniques). Son rôle est de vérifier qu'il reçoit un signal de rafraichissement dans un intervalle de temps donné. En cas de non réception du dit signal, il effectue alors une opération de récupération (remise à zéro, coupure de l'alimentation ou envoi d'un rapport d'erreur) [115].

2.6. Fonctionnalités de reconfiguration des FPGA type SRAM

Depuis leur introduction, les FPGA à mémoire SRAM ont suscité un intérêt croissant en raison de leur potentiel en tant que circuits logiques reconfigurables à volonté. Les fonctionnalités de reconfiguration de ces FPGA peuvent être classées selon certains critères de base tels que l'état du FPGA pendant la reconfiguration, la quantité de ressources reconfigurée ou l'interface de reconfiguration [120].

2.6.1. Activité du FPGA lors de la reconfiguration

Si on se focalise sur l'état du FPGA pendant une reconfiguration, on distingue deux types de reconfiguration :

- **Reconfiguration statique CTR** (Compile-Time Reconfiguration) pour les applications et les FPGA qui ont des modes de fonctionnement et de configuration mutuellement exclusifs. Pour implémenter une quelconque modification, il est nécessaire de recompiler et resynthétiser les blocs modifiés, de régénérer le fichier de configuration pour le système complet et de télécharger cette nouvelle configuration sur le FPGA qui doit être inactif.
- **Reconfiguration dynamique RTR** (Run-Time Reconfiguration) qui offre la possibilité de reconfigurer certains FPGA durant l'exécution d'une application. Ce type de reconfiguration permet d'éviter le surcoût de la désactivation et de la réinitialisation en termes de temps d'exécution.

2.6.2. Quantité reconfigurable des ressources du FPGA

La reconfiguration d'un FPGA à mémoire SRAM consiste à reprogrammer la mémoire de configuration en téléchargeant une séquence de bits connue sous le nom de "bitstream". Ces données définissent la fonctionnalité à traiter par les ressources logiques combinatoires et séquentielles présentes dans le FPGA. En général, deux scénarios différents sont possibles en ce qui concerne la taille du fichier "bitstream" qui sera transféré :

- **Reconfiguration complète** où le fichier "bitstream" est téléchargée en sa totalité pour reprogrammer la mémoire de configuration en entier. Le FPGA est alors reconfiguré dans sa totalité.
- **Reconfiguration partielle** où le fichier "bitstream" téléchargé concerne uniquement la partie du FPGA qui sera reconfigurée. Cela implique la possibilité d'une modification sélective des ressources matérielles du FPGA.

2.6.3. Format du fichier "Bitstream" et mécanisme de téléchargement

Une limitation fondamentale des FPGA commerciaux basées sur mémoire SRAM est leur mécanisme de reconfiguration. Il existe quatre approches primaires adoptées couramment [120] :

- **Configuration avec accès série** : Pendant la configuration, le fichier bitstream est téléchargé en tant que données brutes séquentielles dans la mémoire de configuration FPGA. L'ensemble du FPGA doit être configuré avant qu'une partie puisse être utilisée pour l'exécution de l'application. En raison de ces restrictions architecturales, la reconfiguration partielle n'est pas prise en charge dans ce schéma de configuration (par exemple, famille Altera APEX).
- **Configuration avec accès aléatoire** : Les cellules de reconfiguration sont accessibles de la même manière qu'une RAM standard et les informations de configuration sont lues ou écrites sur les cellules souhaitées à l'aide d'une adresse ligne/colonne. La reconfiguration partielle est supportée avec ce mécanisme d'accès basé sur les adresses, dans une certaine mesure, le temps de configuration est réduit grâce à l'utilisation d'un chemin de données parallèle (par exemple, le bus de données de configuration 32 bits pour la série Xilinx XC6200).
- **Configuration par fenêtrage (windowing)** : le format du fichier bitstream suit un mécanisme de fenêtrage très flexible dans lequel de petites zones du FPGA peuvent être programmées indépendamment l'une de l'autre. Chacune de ces zones est connue sous le nom de "windows". Ce mécanisme est adapté pour spécifier non seulement le bitstream complet, mais aussi les bitstream partiels (par exemple, le FPGA Atmel AT40K).
- **Bitstream basé sur les paquets de bits (frames)** : le fichier bitstream est composé de paquets de bits nommés "frames" qui contiennent des séries de commandes et de données de configuration. Les commandes de configuration englobent la gestion des registres internes de la logique de configuration, tandis que les données de configuration correspondent aux données écrites dans la mémoire de configuration FPGA. Ainsi, l'écriture de données dans la mémoire de configuration nécessite une gestion appropriée des registres de configuration, qui gère en même temps la machine à états finis du moteur de reconfiguration, quand va décrire plus loin dans ce rapport.

Du point de vue du format, les données du fichier bitstream sont encapsulées dans des paquets, où un paquet contient deux sections différentes : l'en-tête et les données. L'en-tête spécifie les registres de configuration concerné (c'est-à-dire la commande de configuration) tandis que les données contiennent la trame de configuration à télécharger (par exemple, les FPGA Spartan 6 et Virtex de Xilinx)

2.6.4. Interface et mode de téléchargement du fichier "bitstream"

Un contrôleur de reconfiguration est nécessaire pour transférer le contenu du fichier bitstream d'un support de stockage, habituellement une mémoire externe non volatile, vers la mémoire de configuration du FPGA. Ce contrôleur peut être directement couplé à la logique de reconfiguration intégrée au FPGA, ou connectée à elle via une interface externe comme le montre la *figure 7-12*.

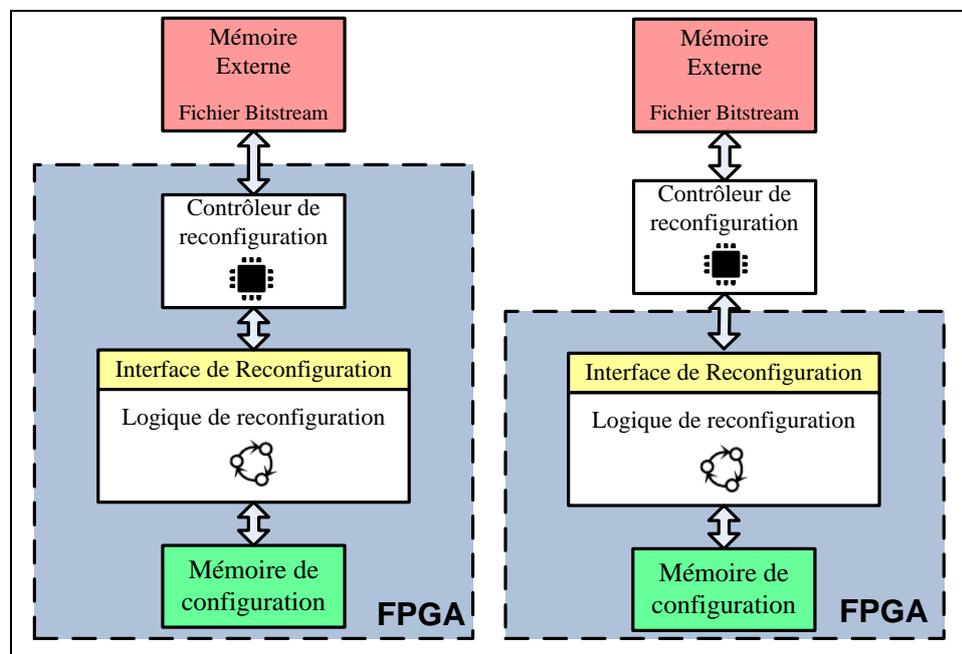


Figure 7-12 : auto reconfiguration (à gauche) et reconfiguration externe (à droite)

- **Interface externe** : un contrôleur intelligent externe, tel qu'un microprocesseur, un processeur DSP ou un microcontrôleur, est généralement utilisé pour transférer de manière synchrone le bitstream en mode maître ou esclave vers FPGA, effectuant ainsi la configuration de la logique programmable de manière séquentielle.
- **Interface interne** : un contrôleur interne embraqué, tel qu'un processeur softcore synthétisé dans le FPGA lui-même, est capable d'accéder à la mémoire de configuration pour prendre en charge le processus de configuration.

Cette solution permet une reconfiguration autonome du FPGA et évite les latences liées à la communication entre le contrôleur externe et le FPGA, permettant donc de diminuer l'impact du temps de reconfiguration sur la performance du système.

3. ARCHITECTURE DES SYSTEMES RECONFIGURABLES DYNAMIQUEMENT

3.1. Description

Pour un système embarqué à base de FPGA à mémoire SRAM, une architecture auto-reconfigurable dynamiquement standard et flexible peut être décomposée en quatre blocs fonctionnels, comme le montre *la figure 7-13* :

- Le processeur softcore ou hardcore reconfigurable qui gère le flux applicatif complet du logiciel embarqué.
- Un support de stockage constitué typiquement par des mémoires volatiles et non volatiles (MNV) à faible coût pour stocker des informations système (Fichiers bitstream, le logiciel embarqué et les données des applications).
- Les interfaces d'entrées/sorties qui relient le système au monde extérieur, composé essentiellement de bus de communication, de capteurs et d'actionneurs locaux.
- Un moteur de reconfiguration, qui prend en charge le processus de reconfiguration.

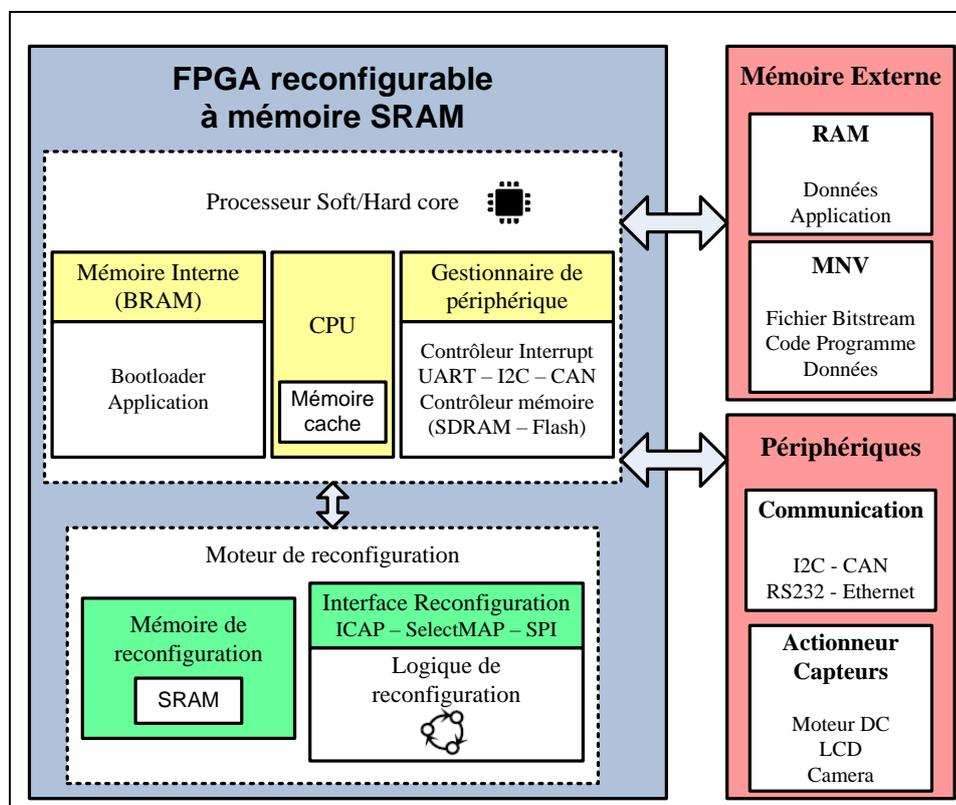


Figure 7-13 : Diagramme fonctionnelle d'une architecture auto-reconfigurable

3.2. Processeurs reconfigurables

Les FPGA modernes offrent aux concepteurs, en plus des ressources reconfigurables à haute densité habituelles, la possibilité de mettre en œuvre des processeurs reconfigurables, disposant des fonctionnalités des composants COTS. Ces processeurs sont chargés de l'exécution d'un logiciel embarqué et permettront la gestion de la mémoire externe, des périphériques et du processus de reconfiguration.

Les FPGA Xilinx permettent d'implémenter deux catégories de processeurs : le processeur hardcore PowerPC [121] et le processeur softcore MicroBlaze [122], et PicoBlaze [123]. Les FPGA Altera fournissent également deux types de processeur : processeur hardcore basé sur ARM Excalibur [124] et le processeur softcore Nios [125].

Les processeurs Hardcore sont intégrés dans le FPGA comme EAB et leur nombre est limité (1, 2, 4 ou aucun processeur hardcore). D'autre part, les processeurs softcore utilisent des ressources reconfigurables, de sorte que le nombre de processeurs effectivement implémentés dépend uniquement de la taille du FPGA.

3.3. Moteurs de reconfiguration

La valeur ajoutée de l'architecture décrite en § 3.1, est le moteur de reconfiguration qui permet d'exploiter d'auto reconfiguration dynamique totale ou partielle des FPGA à mémoire SRAM. La logique programmable introduite par ce moteur doit être conçue de manière à supporter une reconfiguration en temps réel flexible. Dans ce qui suit, on va s'intéresser uniquement à l'approche de conception souple adopté par Xilinx, le fabricant ayant la plus grande expérience dans l'auto configuration temps réel des FPGA à mémoire SRAM [120].

En effet, dans les FPGA Xilinx type SRAM comme Virtex et Spartan 6, le moteur de configuration peut être implémenté comme un IP (Intellectual Property) soft personnalisable pendant la phase de conception matérielle de l'architecture. Les FPGA Xilinx offrent aussi une grande flexibilité concernant les interfaces de configuration disponibles pour accéder, en série ou en parallèle, à la mémoire de configuration. Plus précisément la plupart de ces FPGA sont également équipés d'une interface ICAP (Internal Configuration Access Port) permettant d'accéder directement à la mémoire de configuration.

Le *tableau 7-3* résume les interfaces et certaines fonctionnalités de reconfiguration des FPGA Xilinx à mémoire SRAM :

Famille FPGA	Bande passante maximale reconfiguration (Taille Bus & Fréquence reconfiguration)	Interfaces de reconfiguration	Granularité de reconfiguration
Spartan-3E	400 Mbps (8 bits x 50 MHz)	Série – JTAG – SelectMAP	Colonne complète
Spartan-3A DSP	640 Mbps (8 bits x 80 MHz)	Série – JTAG – SelectMAP SPI – BPI – ICAP	Colonne complète
Virtex-II/ II PRO	400 Mbps (8 bits x 50 MHz)	Série – SelectMAP – ICAP	Frame complète
VIRTEX-4	3.2 Gbps (32 bits x 100 MHz)	Série – JTAG – SelectMAP ICAP	Partie de frame de taille 16 CLB
VIRTEX-5	3.2 Gbps (32 bits x 100 MHz)	Série – JTAG – SPI – BPI SelectMAP – ICAP	Partie de frame de taille 20 CLB
SPARTAN-6	320 Mbps (16 bits x 20 MHz)	Série – JTAG – SPI – BPI SelectMAP – ICAP	Partie de frame de taille 16 CLB

Tableau 7-3 : Fonctionnalités de reconfiguration des FPGA Xilinx [120]

De toutes les interfaces les interfaces de reconfiguration disponibles, la plus attrayante pour la reconfiguration totale ou partielle en temps réel, est la primitive ICAP car elle est accessible depuis l'intérieur du FPGA (via un hard ou soft processeur), et elle offre la bande passante de reconfiguration maximale possible. Comme le montre la *figure 7-14*, l'interface ICAP est lié au contrôleur de reconfiguration par un bus séparé composé de :

- Lignes de commande CLK (horloge), CE (Activation) et WR (write enable).
- Un signal d'état BUSY pour le handshake avec le contrôleur.
- Un bus de données de 8, 16 ou 32 bits selon la famille de FPGA. Ce dernier se divise en un bus IN d'entrée des données IN et un bus OUT de sortie des données.

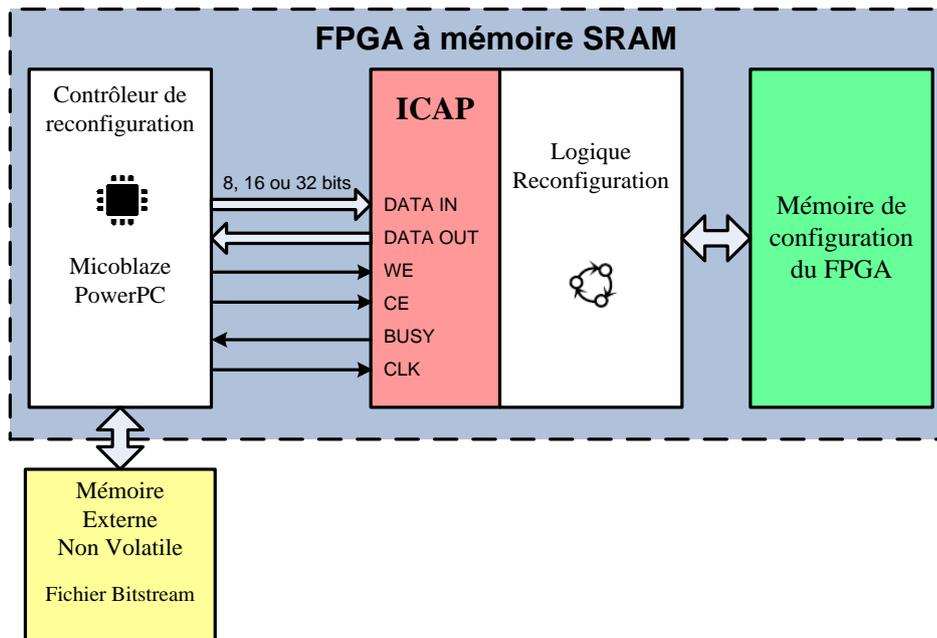


Figure 7-14 : Architecture du contrôleur de reconfiguration Xilinx avec interface ICAP

4. CONCEPTION MATERIELLE DE L'OdB SECONDAIRE

4.1. Architecture adoptée

Conformément aux exigences de l'OdB secondaire, aux enseignements déduits de l'état de l'art des systèmes reconfigurables et aux définitions des solutions de conception de MASAT1 (Chapitre 2 § 3.1 et § 3.2), nous avons dressé l'architecture fonctionnelle de *figure 7-15* :

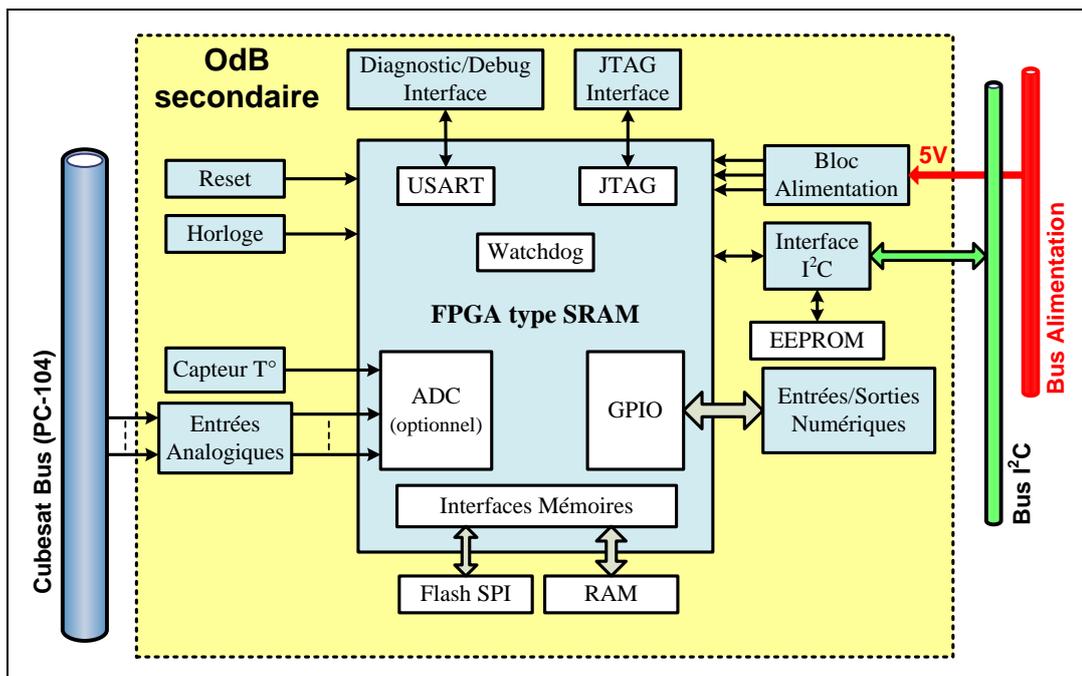


Figure 7-15 : Schéma bloc fonctionnel de l'OdB secondaire de MASAT1

Le schéma bloc fonctionnel prévoit l'utilisation d'un chien de garde logiciel pour la remise à zéro du système en cas de dysfonctionnement logiciel. On distingue aussi l'utilisation de trois types de mémoires :

- **Flash SPI** : Mémoire résistante aux radiations qui joue le rôle de mémoire de masse (disque dur) pour le stockage du système d'exploitation et des logiciels embarqués. Elle contiendra ainsi le "bootloader" qui sera lancé à chaque remise à zéro ou redémarrage de l'OdB et la dernière version valide du fichier bitstream.
- **EEPROM I2C** : Mémoire pour la sauvegarde des données applicatives permanentes et de l'historique concernant les données de télémétrie.
- **SDRAM** : Mémoire de travail permettant le stockage des données temporaires applicatives ainsi que les valeurs temporaires de télémétrie. Comme les mémoires RAM sont sensibles aux effets singuliers dues aux radiations, on doit prévoir une détection/correction d'erreur EDAC (Error Detection And Correction).

L'architecture est constituée aussi de trois types d'interfaces :

- Interface JTAG pour la programmation et la reconfiguration au sol de l'OdB secondaire, via un PC et un adaptateur USB-JTAG.
- Interface série UART pour le débogage et le test au sol.
- Interface I2C pour les besoins de l'architecture distribuée de la plateforme et donc l'interfaçage avec les autres sous-systèmes du segment spatial MASAT1.

Le bloc d'alimentation aura pour rôle de fournir toutes les tensions nécessaires (3.3V, 2.5V, 1.5V, etc.) nécessaires au bon fonctionnement du FPGA et des mémoires utilisées dans l'architecture.

4.2. Choix du FPGA SPARTAN 6 de XILINX

4.2.1. Les arguments

Compte tenu des exigences de reconfiguration pour notre OdB secondaire, les FPGA type SRAM sont la solution la plus appropriée en raison de leur prix, de leur degré élevé d'intégration et de leur flexibilité dans la reconfiguration. Parmi les trois grands fabricants de FPGA Xilinx, Altera et Actel, Xilinx est le plus dominant dans le marché des applications spatiales avec sa mise au point de plusieurs techniques et fonctionnalités permettant l'atténuation et la récupération des effets SEU induites par l'environnement spatial [102].

Le **tableau 7-4** résume les caractéristiques des produits FPGA Xilinx dédiés ou susceptibles d'être utilisés dans les applications spatiales :

Caractéristique		Virtex-4 QV	Virtex-5 QV	Spartan-6 Q
Plage de température		-55°C à +125°C	-55°C à +125°C	-40°C à +125°C
Prix		\$\$\$	\$\$\$	\$
Consommation		■■■■■	■■■■■	■■■
I/O	Banks	13 – 17	24	4 – 6
	Max user	576 - 768	836	328 - 396
Processeur Hard		2 x PowerPC	2 x PowerPC	-
CLB	Logic Cells	56,880 – 14,2128	20,480	11,666 – 23,038
	Register Cells	395 - 987 ⁽¹⁾	1,580 ⁽¹⁾	692 – 1355 ⁽¹⁾
	DSP Slices	128 – 192 ⁽⁴⁾	320 ⁽³⁾	132 – 180 ⁽²⁾
Max Block Mémoire SRAM		10Mb (x 18Kb)	10 Mb (x 36Kb)	4824Kb (x 18Kb)
Horloge	DCM ⁽⁷⁾	12 - 20	12	12
	PLL	-	6	6
TID (Krad)		300	1000	100
SEE (SEL : MeV.cm ² /mg)		> 100	> 100	> 65

⁽¹⁾ : Max Distributed RAM (Kb)

⁽²⁾ : DSP48A1 slice contient 18 x 18 multiplier, an adder, and an accumulator

⁽³⁾ : DSP48E slice contient 25x28 multiplier, an adder and an accumulator

⁽⁴⁾ : XtremDSP with 18x18 multiplier, an accumulator and adder/substracter

⁽⁵⁾ : 18 x18 mulplier

⁽⁶⁾ : with optionnal EDAC protection

⁽⁷⁾ : Digital Clock Manager (DCM)

Tableau 7-4 : Caractéristiques des FPGA type SRAM de Xilinx [107 - 109]

Pour des raisons de coût et de consommation, le choix s'est porté sur l'utilisation du FPGA Spartan-XC6SLX45T-2CSG324C même si ce n'est pas un FPGA Xilinx qualifié espace comme les Virtex-4QV et Virtex-5QV. Cependant, il s'agit d'un produit FPGA de qualité militaire offrant des performances attrayantes pour une faible consommation d'énergie, et les travaux d'évaluation et de test de Spartan-6 pour les applications spatiales (tests ions lourds, tests électriques et mécaniques) ont donné des résultats prometteurs jusqu'à présent [109].

Les principales caractéristiques du FPGA Spartan-XC6SLX45T-2CSG324C sont résumées par le **tableau 7-5** :

Logic Cells	Blocs Logiques Configurables (CLBs)			BRAMs		I/O Banks	Max User I/O
	Slices ⁽¹⁾	Flip-Flops	RAM distribuée	18 Kb ⁽²⁾	Max (Kb)		
43661	6822	54576	401	116	2088	4	296

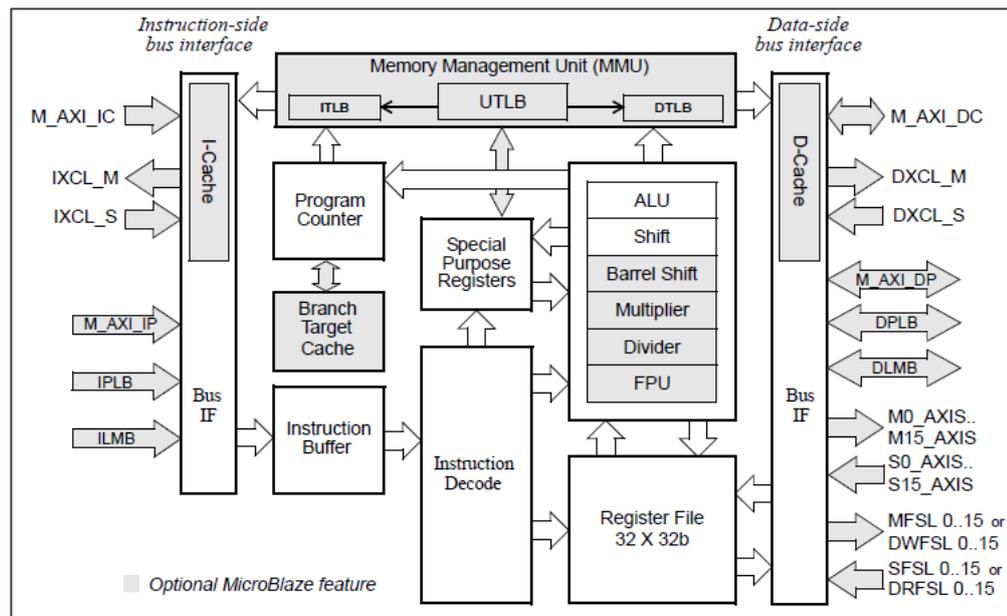
⁽¹⁾ : Chaque slice contient quatre LUT et huit bascules flip-flops.

⁽²⁾ : Les blocs RAM peuvent également être utilisés comme deux blocs indépendants de 9 Kb.

Tableau 7-5 : Résumé des caractéristiques du FPGA Spartan-6 LX-45T [107 - 109]

4.2.2. Le processeur SoftCore MicroBlaze

MicroBlaze est le noyau d'un processeur softcore RISC 32 bits destiné à être utilisé comme microcontrôleur dans les systèmes embarqués à base de FPGA Xilinx (Spartan-6, Virtex ou Artix-7) [121]. Il est caractérisé par une architecture Harvard (bus de mémoire de données et d'instructions séparés) très flexible, prenant en charge plus de 70 options de configuration, comme le montre la *figure 7-16*.



Notes:

- Les parties grise sont optionnelles et montrent la configurabilité du processeur MicroBlaze.
- Les options matérielles Barrel Shifter, Multiplier, Divider et FPU (Floating Point Unit) permettent d'accélérer les traitements en implantant matériellement des opérations mathématiques (division, multiplication, opération sur des nombres réels).

Figure 7-16 : Schéma bloc fonctionnel du softcore MicroBlaze [121]

L'utilisation de Microblaze comme processeur offre ainsi les avantages suivants :

- Selon les exigences du système embarqué, l'utilisateur peut sélectionner une configuration optimisée pour la taille (une architecture de pipeline à trois niveaux) ou une configuration optimisée pour la vitesse (une architecture de pipeline à 5 niveaux avec 1.38 DMIPs/MHz).
- Il supporte nativement le bus AXI4 (Advanced eXtensible Interface) qui fait partie de la famille ARM AMBA (Advanced Microcontroller Bus Architecture). Le standard AXI4 possède trois interfaces :
 - Interface AXI4 pour le mappage du plan mémoire.
 - Interface AXI4-Lite pour la communication avec les périphériques et les registres de contrôle
 - AXI4 / AXI Coherency Extension (ACE) pour l'accès au cache

- Le Mode faible consommation (instruction SLEEP) qui permet au processeur d'entrer en mode veille une fois toutes les transactions en attente sur toutes les interfaces ont été achevées.
- Des fonctionnalités de tolérance aux pannes incluses pour la détection et la correction d'erreur dans les blocs mémoire interne BRAM. Si la tolérance aux pannes est activée, le système ECC (Error Detection and Correction) est automatiquement implémenté dans les contrôleurs d'interface connectés aux BRAM lorsque le système est généré.

Concernant l'utilisation de MicroBlaze dans un système embarqué destiné à une mission spatial, l'université américaine Brigham Young a réalisé une étude comparative entre quatre soft processeurs (MicroBlaze, LEON3, Arm Cortex-M0, OpenRISC 1200) qui a conclu que MicroBlaze apparaît comme le meilleur candidat à utiliser dans un environnement hautement radiatif en raison de sa capacité à exécuter les modèles références (Dhrystone, qsort, Whetstone et basicmath) avec le minimum de ressources matérielles et une sensibilité plus faible [122].

4.2.3. Logique de contrôle de configuration du FPGA Spartan-6

La fonctionnalité de chaque élément configurable dans le FPGA est contrôlée par des trames de données de configuration stockées dans la mémoire de configuration. Pour la modification dynamique du contenu de cette mémoire, le processeur MicroBlaze va utiliser la primitive ICAP_SPARTAN6 comme interface lui permettant de lire et d'écrire dans les registres de configuration, comme ça a été décrit dans la *figure 3-13*.

Il existe trois types de trames de configuration, en fonction du segment spécifique à configurer dans le FPGA :

- Type 0 : CLB, DSP, réseaux d'interconnexion et horloge.
- Type 1 : Blocs mémoires (BRAM).
- Type 2 : Blocs d'entrées/sorties (IOB).

Dans toutes les implémentations mises en œuvre dans l'ODB secondaire MASAT, les données de configurations sont encapsulées dans un paquet Type1 contenant deux sections [123] :

- La section entête organisé toujours sous forme d'un mot de 16 bits comme le montre *le tableau 7-6* :

Type 1	Operation	Adresse registre	Nombre de mots
[15:13]	[12:11]	[10:5]	[4:0]
001	xx	xxxxxx	xxxxx

Tableau 7-6 : Entête d'un paquet de configuration type 1 [123]

- On distingue trois types d'opérations : NOP (00), READ (01) et WRITE (10).
- Une commande de configuration est exécutée lorsqu'elle est lue ou écrite dans le registre de commande approprié. Le *tableau 7-7* résume les principaux registres de configuration utilisés :

Registre	Adresse	Description
CMD	6'h05	Registre de commande
COR2	6'h0B	Définition des options de configuration
GENERAL 1 GENERAL 2	6'h13 6'h14	Registres qui définissent l'adresse de début du fichier bitsream en cas de Multiboot, ainsi que le type de mémoire Flash SPI externe utilisée.
GENERAL 3 GENERAL 4	6'h15 6'h16	Registres qui définissent l'adresse de début du fichier bitsream de récupération en cas de Multiboot.
MODE_REG	6'h18	Registre permettant de paramétrer le mode utilisé pour le redémarrage.

Tableau 7-7 : Description de certains registres de configuration [123]

- La section de données qui suit l'en-tête du paquet de type 1 et qui contient le nombre de mots 16 bits spécifiés par l'en-tête.

4.2.4. Reconfiguration dynamique du FPGA Spartan-6

Pour effectuer une reconfiguration dynamique partielle ou totale du FPGA Spartan-6, on utilise généralement l'IP Core AXI_HWICAP de Xilinx [124]. Comme le montre la *figure 7-17*, ce noyau a la structure d'un périphérique qui utilise la technologie de bus ARM AXI (Advanced eXtensible Interface) [125] comme interface avec le soft processeur intégré MicroBlaze, et intègre le port de reconfiguration ICAP avec une machine à états finis en charge de générer les signaux de contrôle nécessaires.

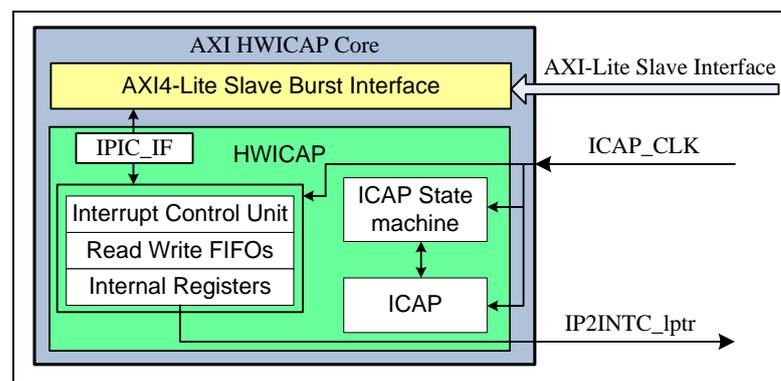


Figure 7-17 : Diagramme bloc fonctionnel de l'IP core AXI_HWICAP de Xilinx

Pour la reconfiguration de FPGA (écriture dans l'ICAP), les données de fichier bitstream sont chargées à partir de la mémoire flash externe et stockées dans une mémoire tampon FIFO (First In First Out), d'où elles peuvent être envoyées vers l'interface ICAP.

L'application logicielle exécutée par le soft processeur MicroBlaze est chargée d'envoyer les commandes de configuration correctes vers la mémoire tampon FIFO. En effet, des pilotes sont disponibles pour contrôler les opérations du noyau AXI_HWICAP et les fonctionnalités incluses dans les fonctions relatives à ces pilotes seront utilisées pour lire et écrire à partir de l'interface ICAP.

4.3. Choix des mémoires

On distingue plusieurs types de mémoires utilisées dans les applications spatiales [126]:

- Mémoires pour le stockage des programmes (bootloader) et des fichiers de configuration (Bitstream) : PROM, EEPROM, MRAM, Flash NOR et Flash NAND
- Mémoires pour les données (Télémétrie, charge utile) : SRAM (Asynchrone, synchrone ou DPRAM) et SDRAM.
- Mémoire de travail du processeur intégré : SRAM (Asynchrone, synchrone), et SDRAM.
- Mémoire de masse pour la sauvegarde des données de la plate-forme : SDRAM, Flash NAND et Flash NOR.

Pour notre ordinateur de bord secondaires nous avons choisi d'utiliser trois types de mémoires :

- Pour simplifier la conception du système et réduire les coûts et la consommation, nous avons utilisé une seule mémoire Flash SPI (Serial Peripheral Interface) qui permettra de stocker le code du logiciel de vol, les données utilisateur et les données de configuration du FPGA.
- Une mémoire de travail pour le processeur type DDR3 SDRAM vu que le FPGA Spartan-6 est équipé d'un contrôleur de mémoire intégré qui ne prend en charge que les mémoires type DDR, DDR2, DDR3, and LPDDR.
- Une mémoire de masse type EEPROM I2C pour qu'elle soit accessible par tous les sous-systèmes de la plateforme MASAT1.

4.3.1. Mémoire Flash

On distingue deux types de mémoires Flash [127] :

- **Mémoire Flash NOR** qui donne un accès direct à n'importe laquelle de ces cellules ce qui implique une vitesse de lecture supérieure par rapport à la Flash NAND. Par contre, la Flash NOR est pénalisée par de faibles vitesses d'écriture, ainsi que par une faible vitesse d'effacement. De ce fait, elle est généralement réservée au stockage des codes des applications qui sont la plupart du temps en lecture seule à haute vitesse.
- **Mémoire Flash NAND** qui impose un accès séquentiel aux cellules mémoires. Elle est plus dense (les cellules de la matrice sont 40 % plus petites que dans la NOR), et moins chère à produire que la Flash NOR. Son architecture lui permet d'accélérer l'écriture et l'effacement par des blocs de programmation des données. Elle est donc idéale pour les applications à faible coût, haute densité, haute-vitesse de programmation / effacement (stockage de données des applications).

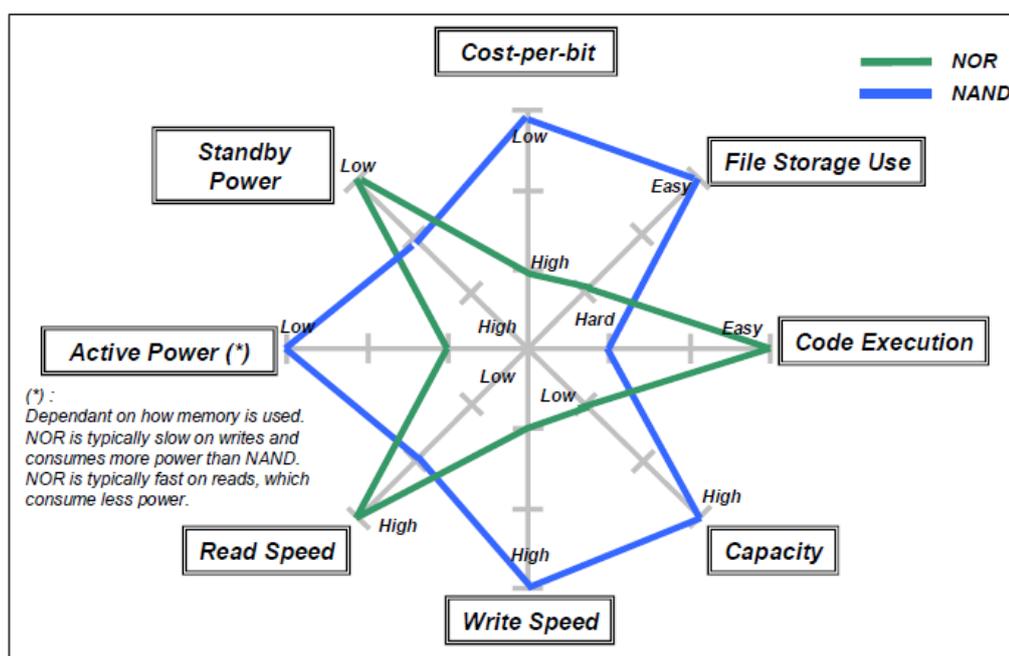


Figure 7-18 : Comparaison des mémoires Flash NOR et Flash NAND [127]

Concernant le comportement des deux types de mémoire Flash dans un environnement spatial, une étude réalisée par Jet Propulsion Laboratory en Californie [128], a permis de conclure que l'architecture des Flash NOR est foncièrement plus robuste vis-à-vis des TID et que les cellules mémoires des deux architectures sont robustes vis-à-vis des événements SEU.

Le choix s'est finalement porté sur la mémoire N25Q064A13ESF40G du fabricant Micron Technology dont la capacité est de 64 Mb [130]. Une interface SPI a été implémentée au niveau du FPGA qui pourra, via le Bank 2, accéder directement à la mémoire Flash SPI. La *figure 7-19* montre le circuit électrique adoptée pour le bloc mémoire Flash SPI :

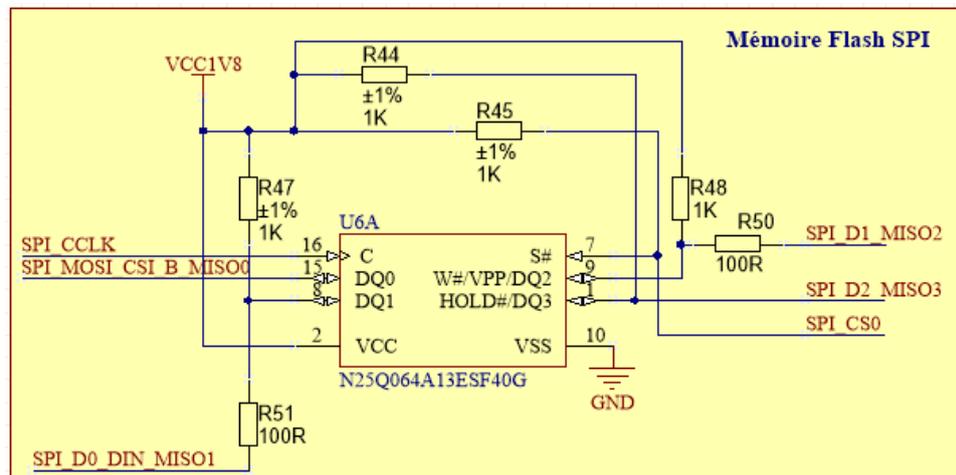


Figure 7-19 : Circuit électrique du bloc mémoire Flash SPI

4.3.2. Mémoire SDRAM

La série de FPGA Spartan-6 de Xilinx contient un contrôleur de mémoire multi-ports embraqué MCB (Memory Controller Block), qui simplifie grandement la tâche d'interfaçage avec les mémoires type SRAM. L'implémentation de ce bloc intégré permet de préserver les ressources du FPGA tout en réduisant la consommation d'énergie et le temps de développement [129].

Chaque FPGA Spartan 6 peut intégrer quatre noyaux MCB pouvant interfacier chacun une mémoire DDR SDRAM de taille maximale 4 Gb et de format 4, 8 ou 16 bits. Le *tableau 7-8* résume les principales fonctionnalités et avantages d'un bloc MCB du FPGA Spartan-6 LX-45T :

Type de mémoire	Débit des données Mb/s (Horloge MHz)		Bande passante maximale par interface MCB		
	Min	Max	4 bits	8 bits	16 bits
DDR	167 Mb/s (83.3 MHz)	400 Mb/s (200 MHz)	1.6 Gb/s	3.2 Gb/s	6.4 Gb/s
DDR2	250 Mb/s (125 MHz)	800 Mb/s (400 MHz)	3.2 Gb/s	6.4 Gb/s	12.8 Gb/s
DDR3	606 Mb/s (303 MHz)	800 Mb/s (400 MHz)	3.2 Gb/s	6.4 Gb/s	12.8 Gb/s
LPDDR	60 Mb/s (30 MHz)	400 Mb/s (200 MHz)	1.6 Gb/s	3.2 Gb/s	6.4 Gb/s

Tableau 7-8 : Fonctionnalités et avantages d'un bloc MCB du Spartan 6 [129]

Le choix s'est finalement porté sur la mémoire DDR3 SDRAM MT41J64M16JT du fabricant Micron Technology avec une capacité d'1Gb et un format des données de 16 bits [131]. Un noyau MCB a été implémenté au niveau du FPGA qui pourra, via le Bank 3, accéder directement à la mémoire SDRAM.

La **figure 7-20** montre le circuit électrique adoptée pour le bloc mémoire DDR3 SDRAM :

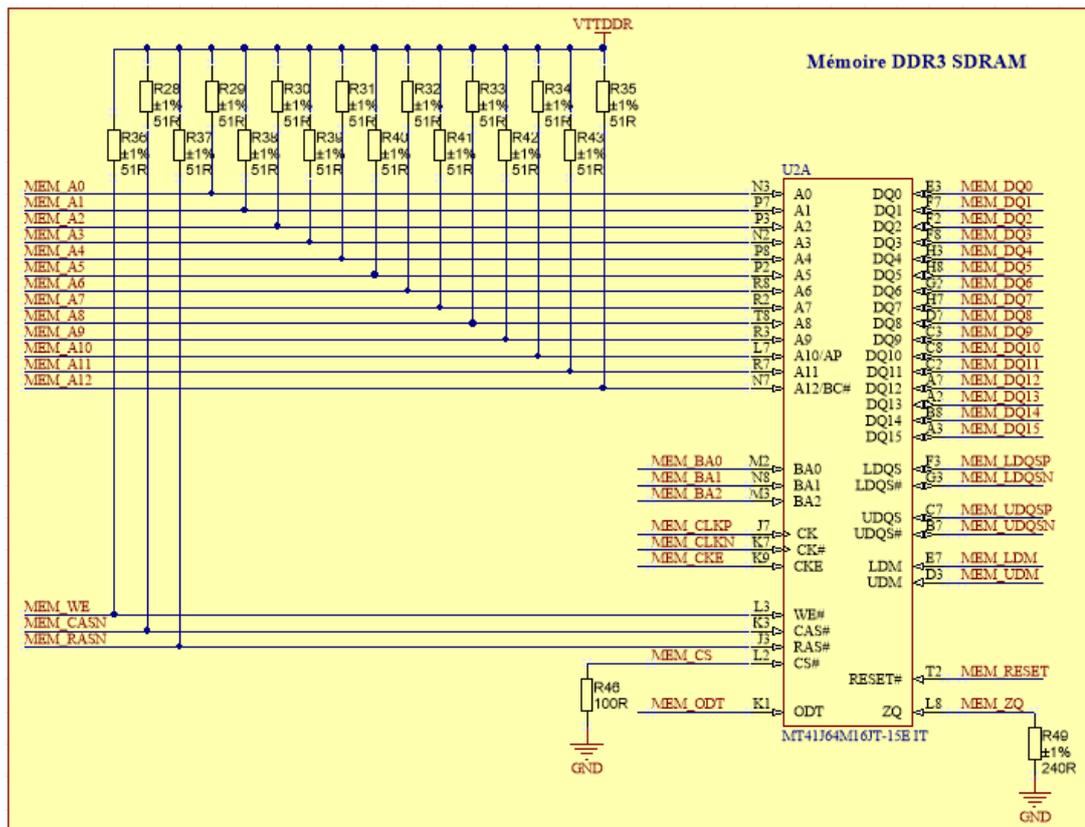


Figure 7-20 : Circuit électrique du bloc mémoire DDR3 SDRAM

4.3.3. Mémoire EEPROM I²C

L'utilisation d'une mémoire série branchée sur le port I²C de la plateforme permettra d'avoir une mémoire de masse partagée par tous les sous-systèmes. Le choix s'est porté sur la mémoire EEPROM série I²C M24C08-W du fabricant STMicroelectronics avec une capacité de 8K bits [132]. Le LogiCORE IP AXI IIC Bus Interface a été implémenté au niveau du FPGA qui pourra, via le Bank 1, accéder directement à la mémoire EEPROM et au bus I²C de la plateforme MASAT1

La **figure 7-21** montre le circuit électrique adoptée pour le bloc mémoire EEPROM IIC :

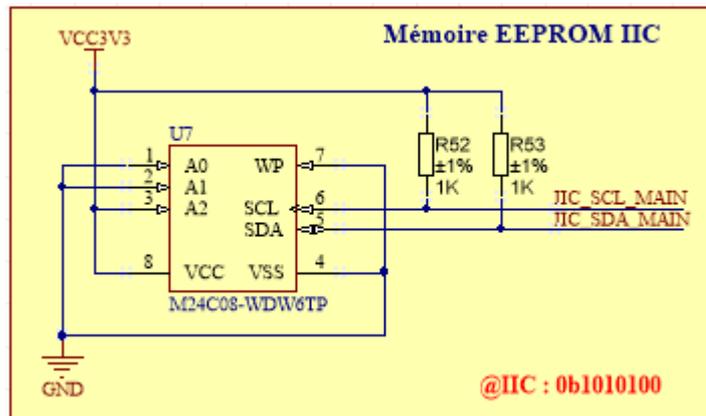


Figure 7-21 : Circuit électrique du bloc mémoire EEPROM IIC

4.4. Conception du bloc d'alimentation

L'OdB secondaire de MASAT1 est alimenté sous 5V-2A à partir de la sortie protégée et configurable OUT0 du module EdB NanoPower 31U, ceci via la broche H1-47 du bus PC104 de la plateforme MASAT1.

Le bloc d'alimentation intégré dans le sous-système OdB secondaire a pour rôle de fournir à partir de cette tension 5V, toutes les tensions nécessaires au bon fonctionnement du module et qui sont résumées par le *tableau 7-9* :

Valeur de la tension	Utilisation
VCC3V3	Bank 0 (VCCO_0) et Bank 1 (VCCO_1) du FPGA Mémoire EEPROM IIC
VCC2V5	Bank 2 (VCCO_2) et VCCAUX du FPGA Port de configuration JTAG
VCC1V8	Mémoire Flash SPI
VCC1V5	Bank 3 (VCCO_3) et VCCINT du FPGA Mémoire DDR3 SDRAM
VTTDDR (0.75V) VTTVREF (0.75V)	Mémoire DDR3 SDRAM

Tableau 7-9 : Tension générée par le bloc d'alimentation de l'OdB secondaire

Le choix s'est porté sur trois régulateurs de tension :

- Le régulateur à découpage abaisseur synchrone 4MHz LTC3633 du fabricant LINEAR Technology [133]. C'est un circuit double sortie qui fonctionne avec une tension d'entrée comprise entre 2.6V et 15V, pour fournir sur chacun des deux canaux de sortie une tension de valeur configurable à partir de 0.6V en utilisant un pont de résistance à la sortie :

$$V_{OUT} = 0.6 \times \left(1 + \frac{R_2}{R_1}\right) \quad (7.1)$$

Il sera utilisé pour générer les tensions 3.3V et 1.5V comme le montre la *figure 7.22*.

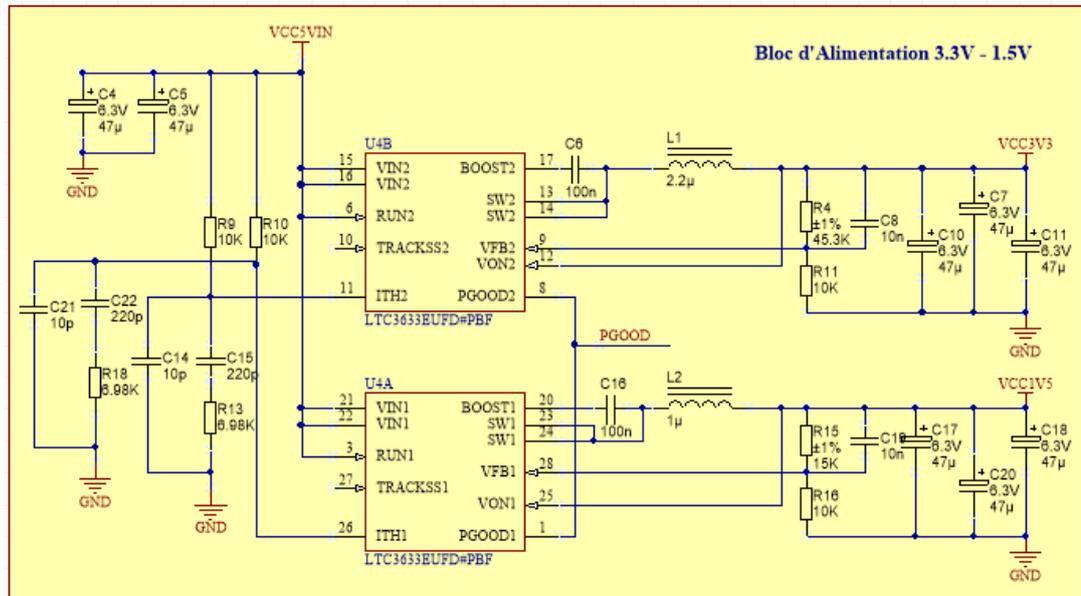


Figure 7-22 : Circuit électrique du bloc alimentation 3.3V et 1.5V

- Le régulateur à découpage abaisseur synchrone 2.25MHz LTC3619 du fabricant LINEAR Technology. C'est un circuit double sortie qui fonctionne avec une tension d'entrée comprise entre 2.5V et 5.5V, pour fournir sur chacun des deux canaux de sortie une tension de valeur configurable entre 0.6V et 5V en utilisant un pont de résistance à la sortie et la même formule (3.1). Il sera utilisé pour générer les tensions 2.5V et 1.8V comme le montre le circuit électrique de la *figure 7-23*.

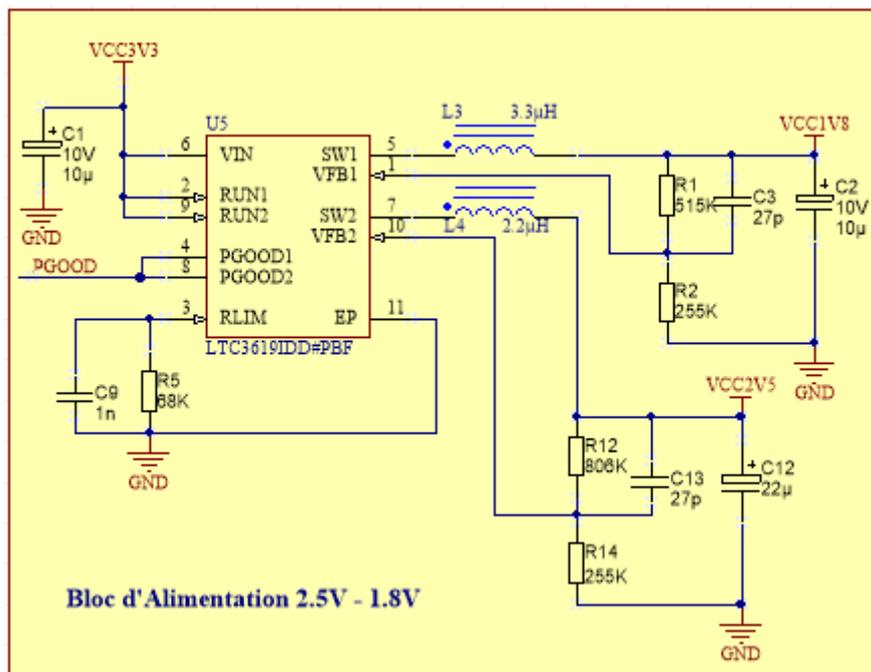


Figure 7-23 : Circuit électrique du bloc alimentation 2.5V et 1.8V

- Le régulateur TPS51200, du fabricant Texas Instrument, spécialement conçu pour fonctionner sous une faible tension d'entrée pour fournir les tension références nécessaires au fonctionnement de la mémoire DDR3 SDRAM comme la tension VTTVREF du contrôleur mémoire intégré IMC (Integrated Memory Controller) [134]. Il permet aussi de monitorer, via le signal drain ouvert PGOOD, la régulation des tensions de sortie. La **figure 7-24** montre le circuit électrique adoptée pour le bloc alimentation DDR VTT.

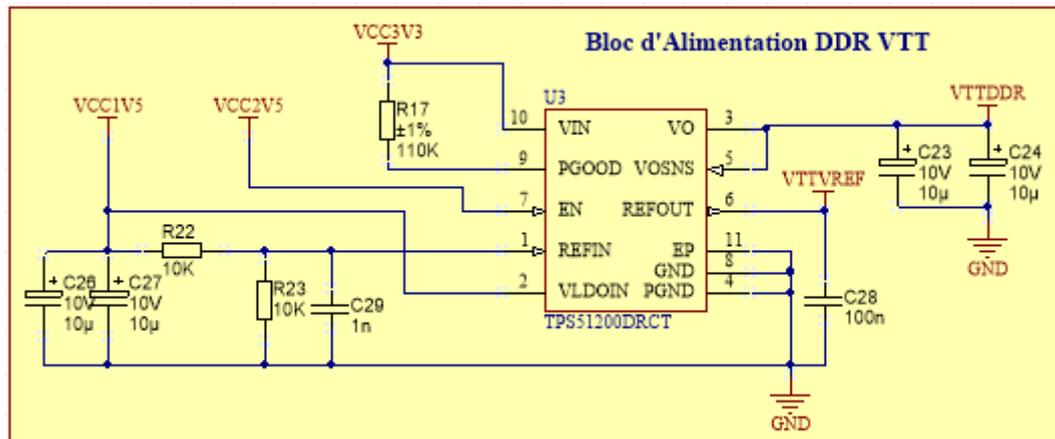


Figure 7-24 : Circuit électrique du bloc alimentation DDR VTT

4.5. Réalisation de la carte OdB secondaire de la mission MASAT1

La réalisation de la carte est prévue sur un circuit imprimé de classe 5 qui va apporter beaucoup de souplesse dans le routage complexe, surtout avec l'utilisation de composants CMS (Composant Monté sur Surface). Cette classe est caractérisée par un isolement minimal entre deux conducteurs de 0.15 mm, un diamètre des trous de 0.3 mm et un diamètre des pastilles de 0.55mm.

Le circuit imprimé comporte six couches :

- Deux couches internes réservées pour les alimentations du FPGA et des composant.
- Une couche interne réservée pour le plan de masse.
- Une couche interne et les deux faces qui sont réservées aux signaux.

Les **figures 7-25** et **7-26** montrent respectivement la représentation des deux faces de la carte OdB secondaire de la mission MASAT1 sachant que la conception a été réalisée en utilisant le logiciel Altium Designer.

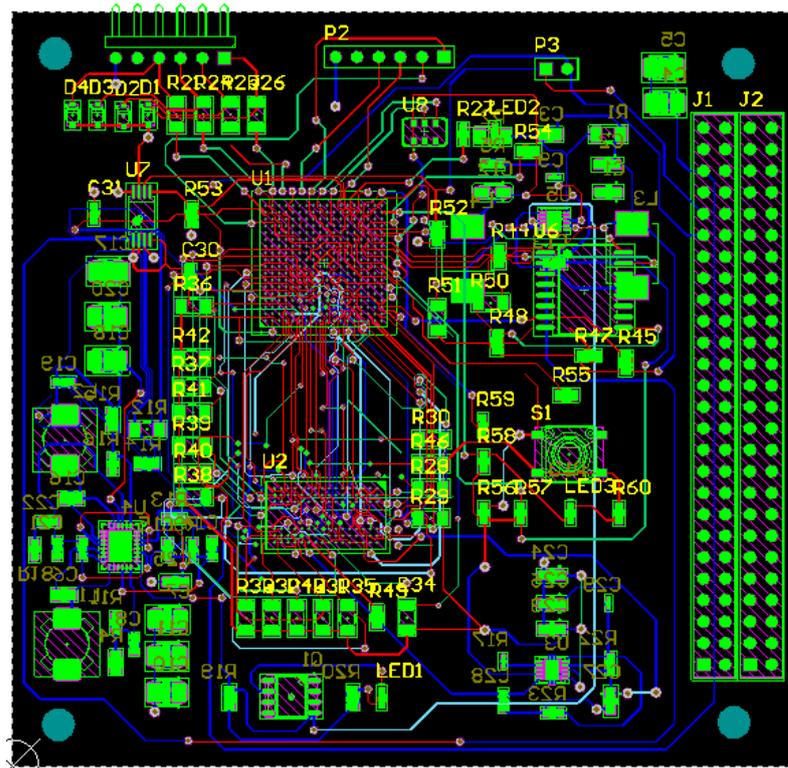


Figure 7-25 : Circuit imprimé de l'OdB secondaire [Altium Designer]

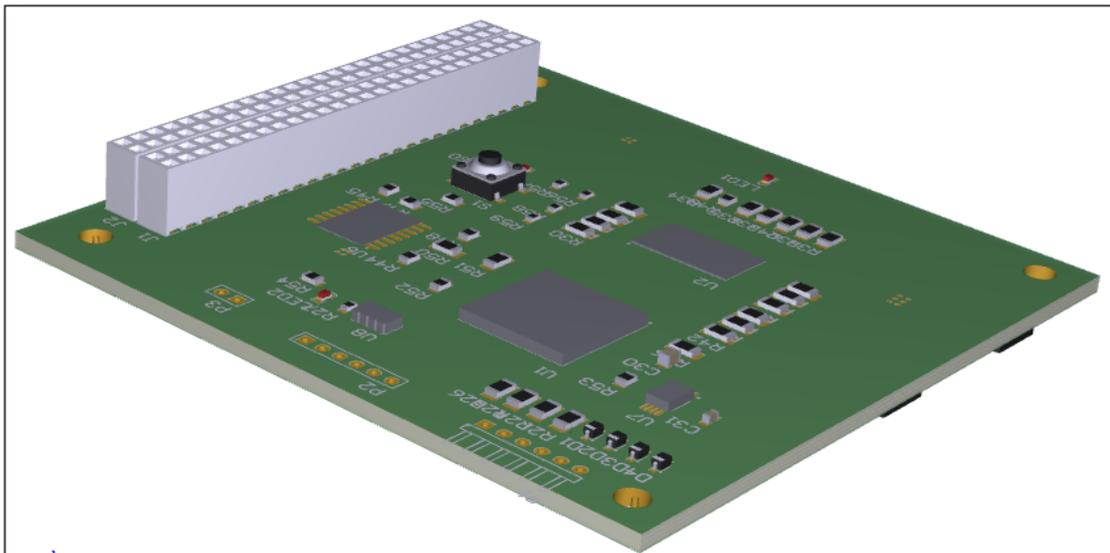


Figure 7-26 : Vue 3D du circuit imprimé de l'OdB secondaire face soudure [Altium Designer]

5. CONCEPTION LOGICIELLE DE L'OdB SECONDAIRE

5.1. Outils de développement

Le choix matériel adopté a aussi été dicté par la disponibilité d'outils de débogage et de développement matériel et logiciel faciles à utiliser. Xilinx fournit en effet une suite logicielle ISE Design Suite 14.3 (Integrated Synthesis Environment) comprenant :

- L'environnement de développement embarqué EDK (Embedded Development Kit) constitué de :
 - **XPS (Xilinx Platform Studio)** pour la configuration matérielle du FPGA à base de blocs IP (Intellectual Property) Plug-and-Play AXI4. Cette plateforme permet d'accélérer le temps de conception d'une plate-forme embarquée basée sur le soft processeur MicroBlaze.

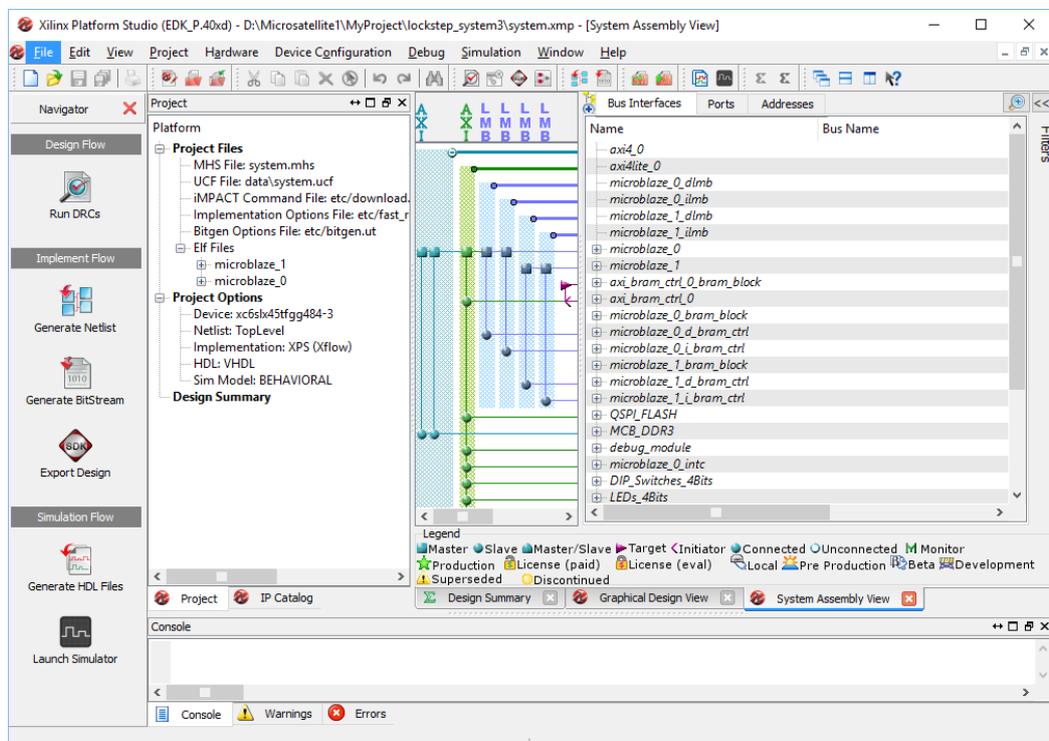


Figure 7-27 : Interface de l'environnement de développement XPS de Xilinx

- **SDK (Software Development Kit)** pour le développement de l'application logicielle qui sera implémenté avec le processeur Microblaze. La plateforme logicielle est basée sur Eclipse 4.5.0 et permet la programmation en C principalement, avec des outils de débogage, de programmation et d'analyse des performances.
- L'outil d'analyse et de conception **PlanAhead** permettant d'implémenter le projet (placement et routage) et la mise en place d'une reconfiguration partielle.

- L'outil **CORE Generator** permettant de créer, via une interface graphique, des modules IP de complexité supérieure comme les contrôleurs mémoires, les fonctions mathématiques ou les interfaces de communications.

La conception d'un système embarqué dans cet environnement de développement doit suivre un tracé linéaire particulier et inévitable qui est résumé par la **figure 7-28** :

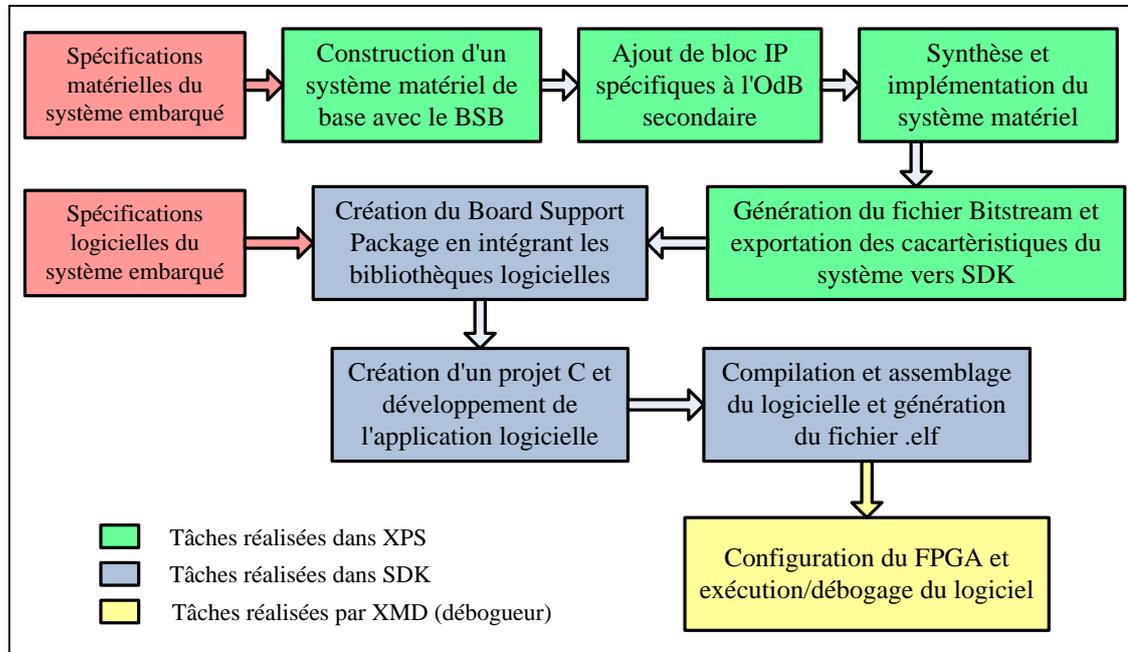


Figure 7-28 : Etapes de la conception d'un système embarqué dans un environnement Xilinx

- L'architecture matérielle du système est créée dans XPS à l'aide du Générateur de système de base BSB (Base System Builder). On pourra configurer les spécifications du soft processeur, ajouter les modules matériels IP core nécessaires. L'ensemble étant interconnectés par les bus AXI.
- Une fois le design matériel à implémenter dans le FPGA est terminé, l'exportation de ses caractéristiques (**fichier bitstream *.bit**) permet de passer à l'environnement SDK pour la génération des bibliothèques logicielles et le développement du logiciel qui sera exécuté par la plate-forme.
- La compilation du projet SDK permettra de générer le fichier le fichier ***.elf** (**Executable and Linkable Format**) qui sera envoyé vers la plate-forme de développement ou le module OdB secondaire à l'aide du débogueur XMD de Xilinx.

5.2. Les bibliothèques logicielles

5.2.1. Le système d'exploitation temps réel XilKernel

Vu que le développement de l'ODB secondaire de MASAT1 a démarré en parallèle avec les autres phases de conception du projet, la plupart des choix technologiques concernant les sous-systèmes de la plateforme n'était pas connu et surtout celui concernant l'architecture et le système d'exploitation temps réel.

Le choix s'est porté alors sur le noyau temps réel Xilkernel qui fonctionne sur les processeurs MicroBlaze, PowerPC 405 et PowerPC 440. Les éléments suivants sont les facteurs déterminants qui ont dicté ce choix :

- C'est une bibliothèque logicielle libre, petite et robuste intégrée à EDK. Elle permet un haut degré de personnalisation permettant d'adapter le noyau en termes de taille et de fonctionnalité.
- La possibilité de porter et d'utiliser des bibliothèques libres comme lwIP (Light Weight Internet Protocol) pour les communications réseau ou LibXil Isf (In-System Flash) qui fournit les fonctions nécessaires pour écrire, lire et effacer la mémoire Flash SPI.

a. Organisation

La conception du noyau XilKernel est hautement modulaire permettant de sélectionner et personnaliser uniquement les modules qui sont nécessaires pour l'application courante. Le schéma bloc de la *figure 7-29* montre les différents modules constituant le noyau et une description des modèles d'ordonnancement et des fonctionnalités de XilKernel est donnée en [80].

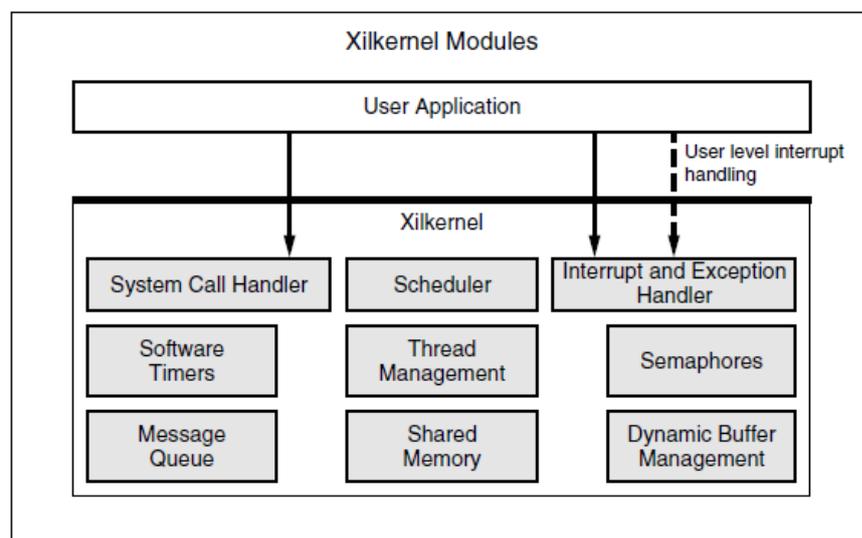


Figure 7-29 : Modules constituant XilKernel [80]

b. Création d'applications XilKernel

Le noyau Xilkernel est organisé sous forme d'une bibliothèque de fonctions et pour le monter sur un processeur Microblaze, il faut :

- Inclure Xilkernel dans la plate-forme logicielle, le configurer de manière appropriée, et exécuter **Libgen** pour générer la bibliothèque Xilkernel et qui se nomme *libxilkernel.a*.
- Editer et développer séparément le code source de l'application C.
- Lier l'application avec la bibliothèque Xilkernel pour construire l'image finale du noyau qui sera implémentée dans le processeur.

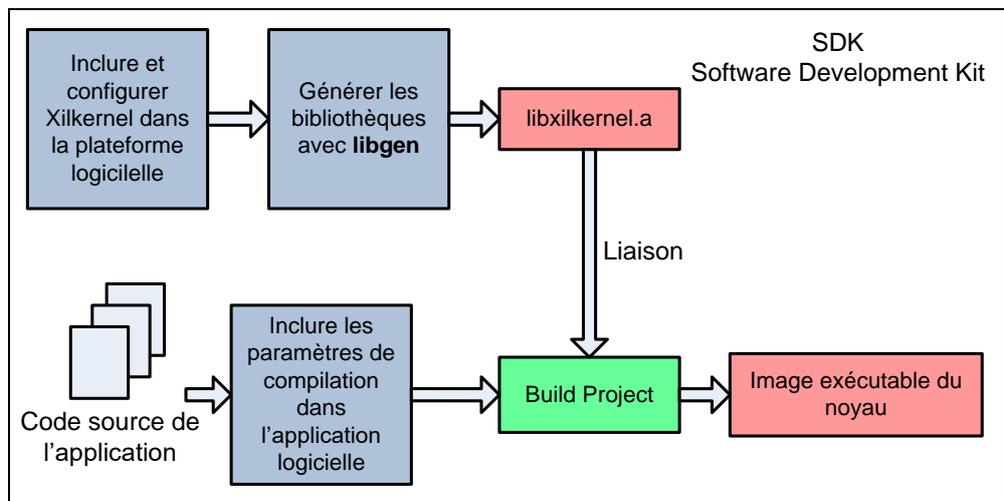


Figure 7-30 : Etapes pour créer une application XilKernel

5.2.2. La bibliothèque LibXil ISF

C'est une bibliothèque qui permet à l'application logicielle, via un pilote interface IP Core SPI implémenté dans XPS, de [80] :

- Ecrire, lire et effacer la mémoire Flash SPI du l'ODB secondaire.
- Protéger les données stockées dans la mémoire Flash SPI contre toute modification indésirable en activant la protection de secteurs mémoires.
- Supporte plusieurs instances de mémoire Flash à la fois à condition qu'elles soient du même fabricant (Atmel, Intel, STM, Winbond, Micron ou Spansion).
- Utilise les pilotes d'interface Xilinx SPI en mode interruption ou non. En mode d'interruption, l'application utilisateur doit reconnaître toute interruption associée au contrôleur d'interruption implémenté dans XPS.

5.3. Implémentation de l'auto reconfiguration dynamique de l'OdB secondaire

L'architecture auto reconfigurable de l'OdB secondaire de MASAT1 est basé sur le soft processeur MicroBlaze, des IP cores intégrés et un logiciel embarqué, qui permettent d'implémenter les fonctionnalités suivantes :

- Une interface de configuration SPI externe avec la mémoire Flash SPI, comme mode de téléchargement du fichier Bitstream (§ 2.6.4).
- Une redondance matérielle par duplication avec comparaison, comme moyen de prévention et d'atténuation des effets des radiations (§ 2.5.2).
- Une auto reconfiguration dynamique totale via l'interface ICAP, comme moyen de récupération après une défaillance (§ 3.3).

5.3.1. Mode et interface externe de configuration

Le FPGA Spartan-6 LX-45T utilise deux broche M[1:0] du Bank 2, pour sélectionner le mode et l'interface externe de configuration, comme le résume le *tableau 7-10* :

Mode de configuration	M[1:0]	Taille du bus
Master SelectMAP/BPI	00	8, 16
Master Serial/SPI	01	1, 2, 4
JTAG	xx	1
Slave SelectMAP	10	8, 16
Slave Serial	11	1

Tableau 7-10 : Mode de configuration du FPGA Spartan-6 [123]

Dans le mode de configuration Master Serial/SPI choisi, nous avons utilisé l'interface de configuration série (voir *figure 7-31*) pour permettre au soft processeur Microblaze d'auto configurer le FPGA directement à partir de la mémoire Flash SPI externe :

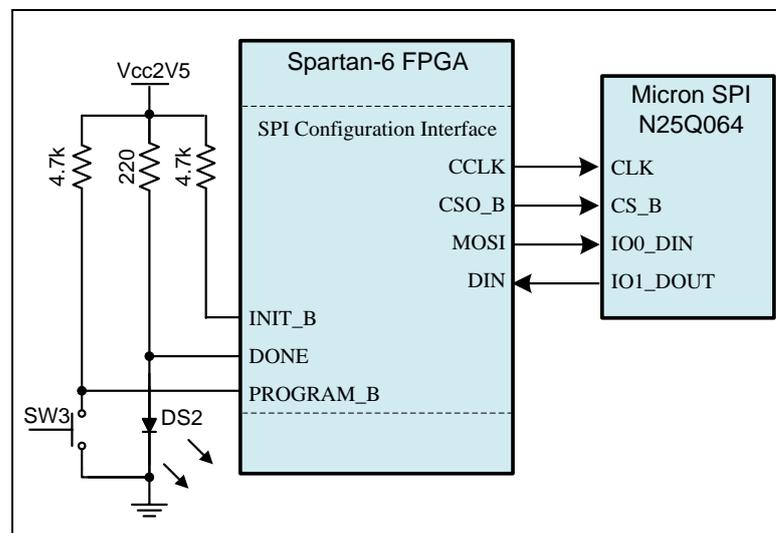


Figure 7-31 : Interface de configuration série de l'OdB secondaire MASAT1

La reconfiguration de FPGA peut alors être déclenchée après :

- Chaque mise sous tension du circuit.
- L'application d'une impulsion sur la broche PROGRAM_B.
- Une IPROG logicielle destinée vers l'interface ICAP.
- Une séquence de récupération due à une configuration défailante.

5.3.2. Implémentation des fonctionnalités Multiboot/fallback

Le FPGA Spartan-6 possède une logique dédiée lui permettant d'utiliser des fonctionnalités de reconfiguration multiboot avec récupération après un défaut de configuration (fallback) [123]. Ces fonctionnalités permettront à l'OdB secondaire de MASAT1 d'avoir deux versions du fichier de configuration (bootloader) avec les deux versions du logiciel de vol correspondants. La mission utile peut basculer via une commande provenant de la station au sol sur n'importe laquelle des deux versions.

Pour l'implémentation de ces fonctionnalités, on va utiliser plusieurs fichiers images qui seront stockés dans la même mémoire Flash SPI externe, comme cela est illustré par la *figure 7-32* :

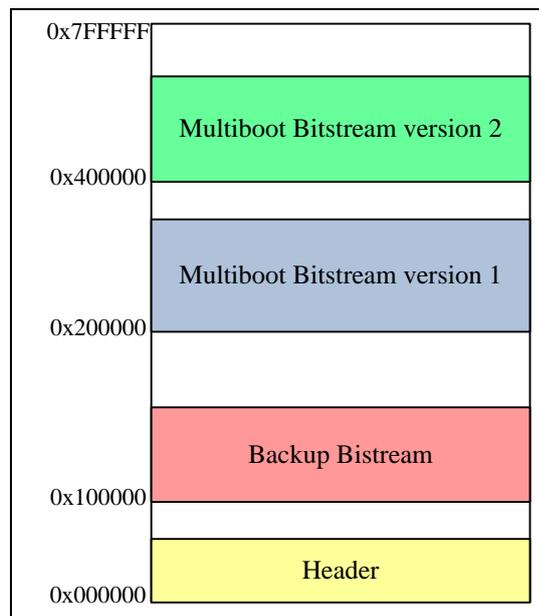


Figure 7-32 : Plan mémoire de la mémoire Flash SPI

- La première fichier image est celui de l'en-tête qui est constitué d'un petit train de bits qui doit commencer à l'adresse 0x0 pour déclencher la reconfiguration. Il contient un ensemble de commandes envoyées à la mémoire de configuration du FPGA à l'aide de la primitive ICAP_SPARTAN6. Le *tableau 7-11* montre la séquence des commandes utilisées dans notre fichier en-tête :

Données de configuration		Description
FFFF	FFFF	Mot de départ
AA99	5566	Mot de synchronisation
31E1	FFFF	Activer Reset après erreur (registre COR2)
3261 3281	xxxx 03xx	Initialiser les registres GENERAL1,2 avec l'adresse initiale du bootloader de l'application à configurer (0x200000 ou 0x400000) et le code de la mémoire Flash SPI utilisée (0x03)
32A1 32C1	0000 0301	Initialiser les registres GENERAL3,4 avec l'adresse initiale du bootloader backup (0x100000) et le code de la mémoire Flash SPI utilisée (0x03)
3301	2100	Configurer le mode Reboot (registre MODE_REG)
3201	001F	Prendre en compte l'initialisation
30A1	000E	Envoyer la commande IPROG (registre CMD)
2000	2000	Pas d'opération

Tableau 7-11 : Fichier entête pour une commande IPROG via ICAP

- La deuxième fichier image est le bitstream de la configuration backup de récupération. L'adresse initiale de ce fichier doit être spécifiée dans les registres de commande GENERAL3.4.
- Les deux autres images sont les deux versions des fichiers bitstream Multiboot. Les valeurs des registres GENERAL1,2 doivent être initialisée avec l'adresse de la version du bitstream que l'utilisateur envisage de configurer.

Chaque fichier bootloader a trois tentatives pour configurer le FPGA et télécharger le logiciel embarqué vers la mémoire SDRAM externe. Si une erreur est détectée (erreur de CRC ou erreur du chien de garde), un compteur s'incrémente (registre BOOTSTS) et la configuration recommence. La séquence de reconfiguration complète est la suivante :

- Au démarrage, la mémoire de configuration est effacée et l'adresse de début 0x0 est utilisée pour la reconfiguration FPGA. Le fichier en-tête initialise les REGISTRES1,2,3,4 et émet une commande IPROG à l'aide de l'interface ICAP.
- La particularité de la commande IPROG est de ne pas réinitialiser la logique de reconfiguration dédiée. Ainsi, l'adresse de départ définie dans REGISTER1,2 (image multiboot) est utilisée lors de la prochaine reconfiguration au lieu de l'adresse par défaut (zéro).
- Si la reconfiguration échoue après trois tentatives, le bootloader backup définie par les registres 3,4 est chargé pour reconfigurer la FPGA.

5.3.3. Implémentation de la duplication matérielle avec comparaison

L'approche "Lockstep" adoptée consiste à implémenter deux processeurs redondants MicroBlaze avec chacun une mémoire locale dédiée. Les deux reçoivent les mêmes signaux d'entrée, fonctionnent en synchronisme mais complètement indépendants l'un de l'autre. Cette synchronisation est réalisée par l'interface "Lockstep" de MicroBlaze qui est conçue pour connecter les deux instances maître et esclave du soft processeur en utilisant les signaux répertoriés dans le *tableau 7-12* [121] :

Signal Name	Description	Direction
Lockstep_Master_Out	Signaux allant du processeur maître au processeur esclave (côté maître).	Sorties
Lockstep_Slave_In	Signaux allant du processeur maître au processeur esclave (côté esclave).	Entrées
Lockstep_Out	Tous les signaux de comparaison provenant des processeurs maître et esclave.	Sorties

Tableau 7-12 : Signaux de l'interface "Lockstep" de MicroBlaze [121]

Pour configurer correctement le système, le paramètre *C_LOCKSTEP_SLAVE* est activé sur l'instance esclave de MicroBlaze et le port *Lockstep_Master_Out* du processeur maître est connecté au port *Lockstep_Slave_In* du processeur esclave. Chaque processeur reçoit tous les signaux d'entrée mais seule le processeur maître peut commander les périphériques du système embarqué. Le port *Lockstep_Out* de chaque processeur alimente un IP core comparateur comme le montre la *figure 7-33* :

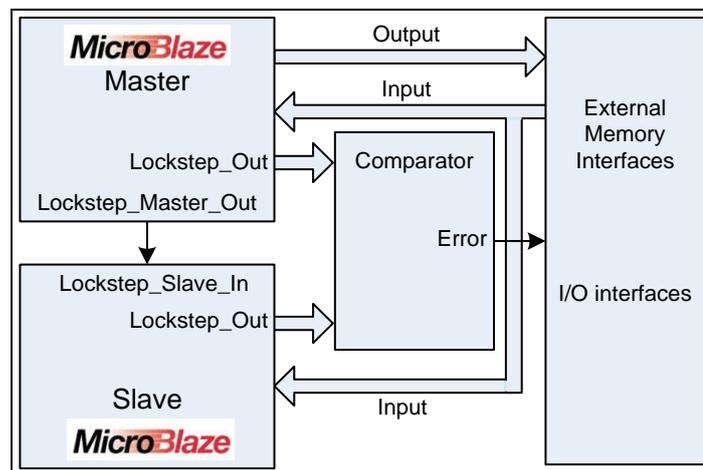


Figure 7-33 : Diagramme bloc fonctionnel du système de détection d'erreur Lockstep

Xilinx fournit un noyau IP core formellement connu sous le nom de *MicroBlaze Comparator* qui peut être adressable via le bus d'interconnexion AXI adopté pour l'architecture implémentée dans le FPGA Spartan-6. La *figure 7-34* résume les ports du noyau MicroBlaze Comparator :

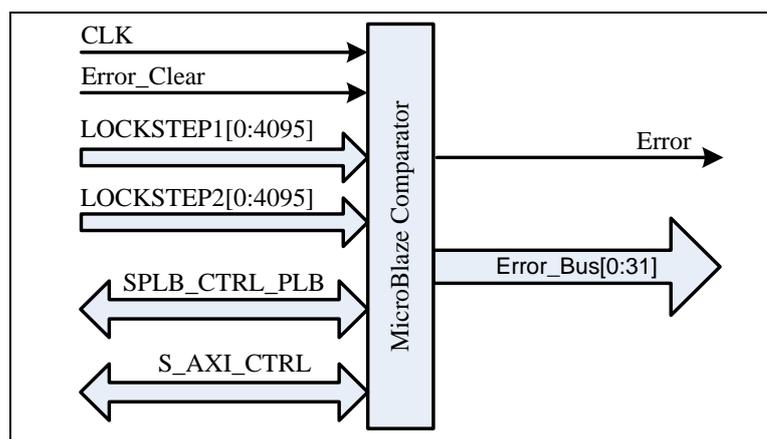


Figure 7-34 : Les ports de l'IP core MicroBlaze Comparator

- Le comparateur est entièrement synchrone via le signal d'horloge CLK.
- Les ports LOCKSTEP1 et LOCKSTEP2 permettent la connexion avec les ports Lockstep_Out des processeurs MicroBlaze maitre et esclave.
- Le comparateur fournit un signal indicateur d'erreur sur un bit, en plus d'un bus pour l'identification de l'erreur.

Le test par injection d'erreurs peut être activé et contrôlé à travers des registres de contrôle du comparateur, qui permettant ainsi de cibler un processeur MicroBlaze spécifique, une interface du processeur ou le bit d'une interface, pour une injection d'erreur. Ce processus sera adopté pour tester notre implémentation et le *tableau 7-13* résume les principaux registres utilisés à cette fin :

Register	Description
Control Register	Active une injection d'erreurs et efface les erreurs détectées.
Status Register	Indique quelle interface a subi une défaillance depuis le dernier effacement d'erreurs détectées.
Interface Fault Register 1	Permet une injection d'erreur sur l'interface spécifiée pour MicroBlaze 1 (LOCKSTEP1)
Interface Fault Register 2	Permet une injection d'erreur sur l'interface spécifiée pour MicroBlaze 2 (LOCKSTEP2)
Fault Inject Bit Register	Spécifie le bit d'une interface pour injecter l'erreur.

Tableau 7-13 : Registres de l'Ip Core MicoBlaze Comparator [117]

5.3.4. Architecture matérielle implémentée dans le FPGA Spartan-6

L'architecture matérielle illustrée par la *figure 7-35* repose sur les approches décrites précédemment et sa mise en œuvre a été effectuée sur le kit d'évaluation Xilinx Spartan-6 FPGA SP605 en utilisant la suite Xilinx Platform Studio (XPS).

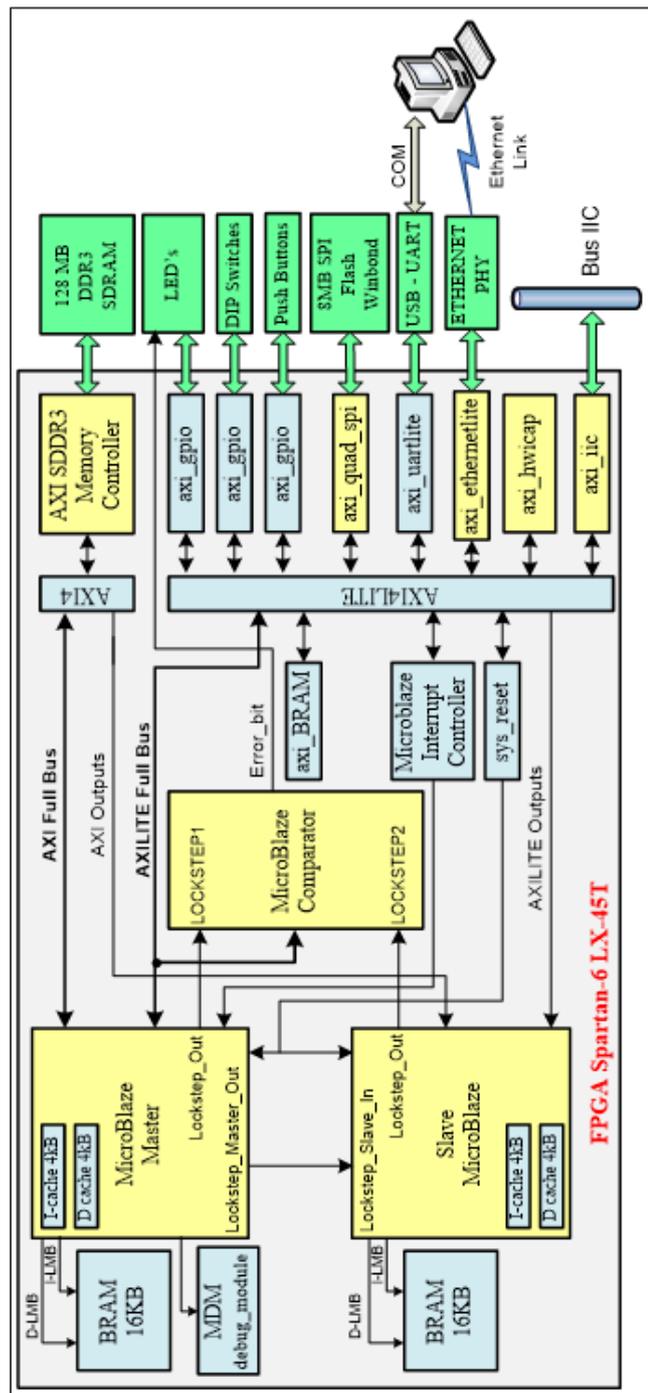


Figure 7-35 : Système référence de la plateforme matérielle implémentée sur le FPGA

La conception est basée sur le bus d'interconnexion AXI qui fonctionne à 50 MHz et comprend les principaux noyaux IP suivants :

- Contrôleur de mémoire AXI SDDR3 fournissant une interface avec le SDRAM embarqué qui contiendra le logiciel embarqué copié à partir de la mémoire flash SPI externe par le fichier bootloader.
- Interface Flash SPI (axi_quad_spi) fournissant une interface avec la mémoire Flash SPI Winbond W25Q64BV SPI.

- Sorties/entrées à usage général (axi_gpio) utilisées pour communiquer avec les LEDs, dont est connectée au comparateur pour indiquer la détection d'une erreur.
- Interface UART (Universal Asynchronous Receiver Transmitter) fournissant une communication série avec un PC (port COM), via un convertisseur UART / USB. Cette interface permettra la simulation des erreurs et le débogage du système.
- L'IP Core AXI_HWICAP utilisé pour la reconfiguration dynamique et complète du système en cas de détection d'erreurs.
- Interface pour le canal de communication Ethernet (axi_ethernetlite) qui va servir pour la mise à jour à distance de l'ODb secondaire reconfigurable (§ 5.4).

5.3.5. Test de l'injection et la récupération d'erreur

L'application logicielle de test implémente et exécute cinq sous-programmes différents pour tester et démontrer les approches décrites précédemment :

- Le sous-programme d'injection et de récupération d'erreur effectue une injection d'erreur sur un bit dans l'interface AXI_DP et déclenche la reconfiguration complète du système lorsque l'erreur est détectée.
- Le sous-programme de lecture et d'affichage (Terminal Windows) de registre d'état du noyau MicroBlaze Comparator. Une valeur lue autre que zéro indique qu'une erreur a été détectée par le comparateur.
- Le sous-programme permettant d'effacer les erreurs détectées par le noyau MicroBlaze Comparator, en activant le bit Clear Fault du registre de contrôle et en vérifiant que le registre d'état a été effacé.
- Le sous-programme permettant d'exécuter un test des périphériques après une reconfiguration. Cet autotest comprend le contrôleur d'interruption, les LEDs, les commutateurs, la base de temps du chien de garde et l'interface ICAP.
- Le sous-programme permettant de vérifier les fonctionnalités globales du noyau MicroBlaze Comparator, y compris l'injection d'erreur. Le test vérifie que l'injection d'erreur sur tous les bits de tous les bus connectés fonctionne et que le registre d'état s'efface lorsqu'il est commandé.

Pour l'exécution des opérations sur le comparateur, nous avons utilisé trois fonctions liées à la bibliothèque du driver fournie par Xilinx :

- ***MbComparatorClearCompError*** () : cette fonction efface le registre d'état.
- ***MbComparatorErrorInject*** () : cette fonction commande l'injection d'une erreur.
- ***MbComparatorReadStatus*** () : cette fonction lit le registre d'état.

L'organigramme de la **figure 7-36** illustre les étapes d'exécution du sous-programme d'injection et de récupération d'erreurs :

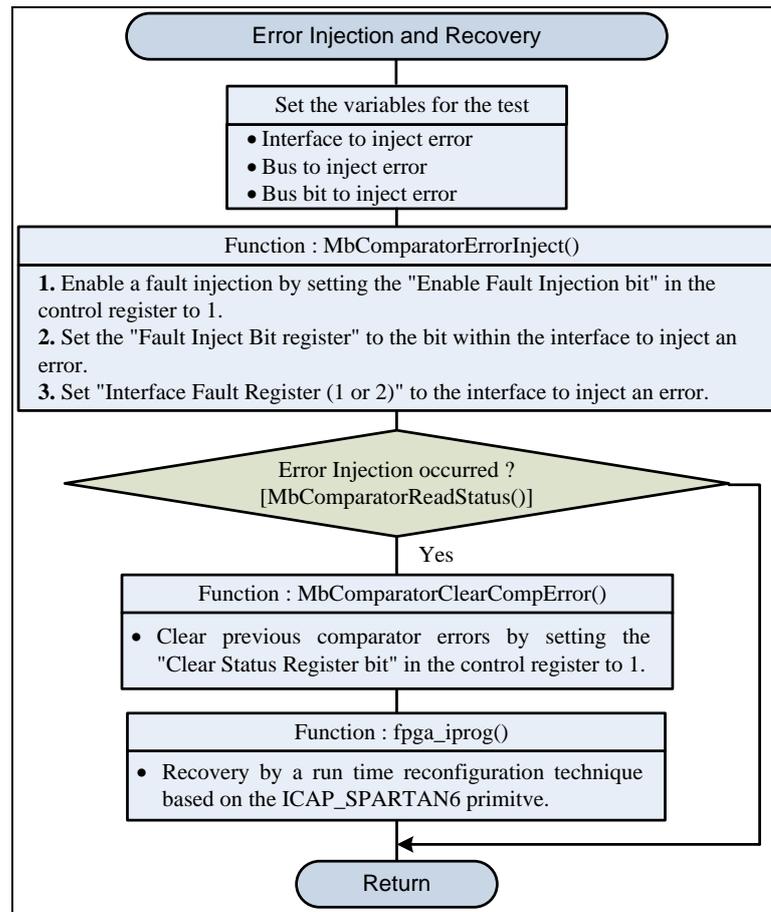


Figure 7-36 : Sous-programme d'injection et de récupération d'erreurs

Pour la récupération d'erreur, nous avons utilisé une technique de reconfiguration en exécutant certaines commandes de configuration, en particulier la commande IPROG qui concerne le registre CMD. Notre application va inclure la fonction de bas niveau **HwIcapLowLevel()** du pilote axi_hwicap en la personnalisant avec la machine d'état concernant la séquence de redémarrage désirée comme indiqué dans la **figure 7-37** :

```

xhwicap_low_level.c
static u32 ReadId[HWICAP_EXAMPLE_BITSTREAM_LENGTH] =
{
    0xFFFF, /* Dummy Word */
    0xAA99, /* Sync Word*/
    0x5566, /* Sync Word*/
    0x30A1, /* Write Word to CMD register */
    0x0000, /* Null command */
    0x30A1, /* Write Word to CMD register */
    0x000E, /* IPROG command */
    0x2000, /* Type 1 NO OP */
    0x2000,
    0x2000,
    0x2000,
};
  
```

Figure 7-37 : Bitstream pour une commande IPROG via ICAP

Pour déboguer l'exécution du test d'injection et de récupération d'erreurs, un programme terminal Windows tel que Tera Term doit être configuré, et le kit d'évaluation SP605 doit être configuré dans le mode de configuration Master serial SPI (M [1: 0] = 01) comme indiqué sur la *figure 7-38* :

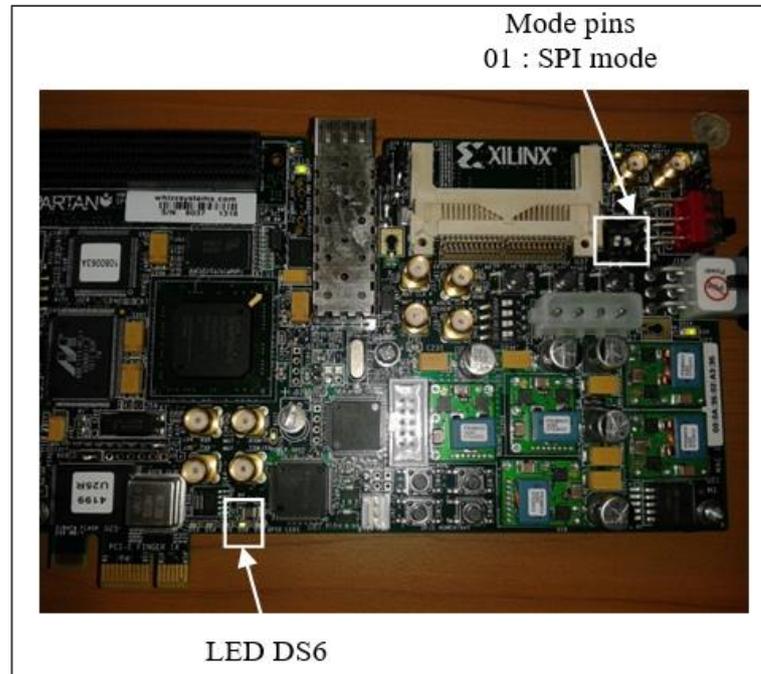


Figure 7-38 : Paramétrage du mode de configuration et LED indicatrice d'erreur

Après la mise sous tension du kit de développement ou l'appui sur le bouton PROG, l'application Terminal Windows affiche l'évolution du système comme le montre la *figure 7-39* :

```

COM13 - Tera Term VT
File Edit Setup Control Window Help
SREC SPI Bootloader
Loading SREC image from SPI flash @ address: 00180000
Bootloader: Processed (0x)00000768 S-records
Executing program starting at address: 00000000

*****
** Dual Lockstep Microblaze Solution **
*****

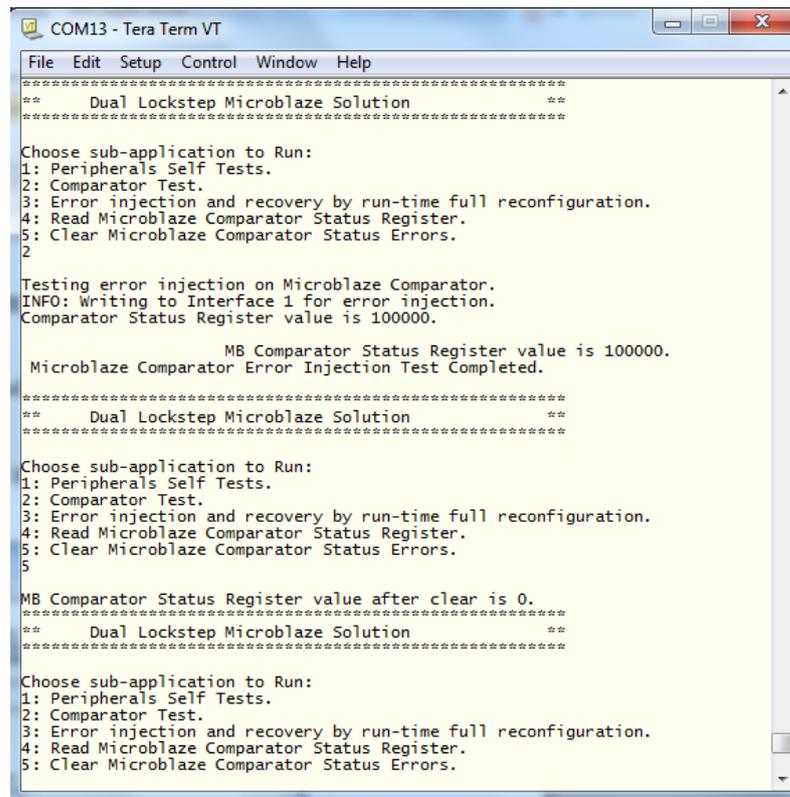
Choose sub-application to Run:
1: Peripherals Self Tests.
2: Comparator Test.
3: Error injection and recovery by run-time full reconfiguration.
4: Read Microblaze Comparator Status Register.
5: Clear Microblaze Comparator Status Errors.

```

Figure 7-39 : Démarrage de l'application logicielle pour le test d'injection d'erreur

- Le bootloader indique son état pendant le chargement de l'application logicielle à partir de la mémoire Flash SPI.
- Après cela, l'application démarre à l'adresse SDRAM 0x00 et affiche l'invite permettant de sélectionner le sous-programme à exécuter.

Les *figures 7-40* et *7-41* montrent respectivement une capture d'écran de l'exécution des sous-programmes : test comparateur et injection et récupération d'erreur :



```

COM13 - Tera Term VT
File Edit Setup Control Window Help
*****
** Dual Lockstep Microblaze Solution **
*****

Choose sub-application to Run:
1: Peripherals Self Tests.
2: Comparator Test.
3: Error injection and recovery by run-time full reconfiguration.
4: Read Microblaze Comparator Status Register.
5: Clear Microblaze Comparator Status Errors.
2

Testing error injection on Microblaze Comparator.
INFO: Writing to Interface 1 for error injection.
Comparator Status Register value is 100000.

MB Comparator Status Register value is 100000.
Microblaze Comparator Error Injection Test Completed.

*****
** Dual Lockstep Microblaze Solution **
*****

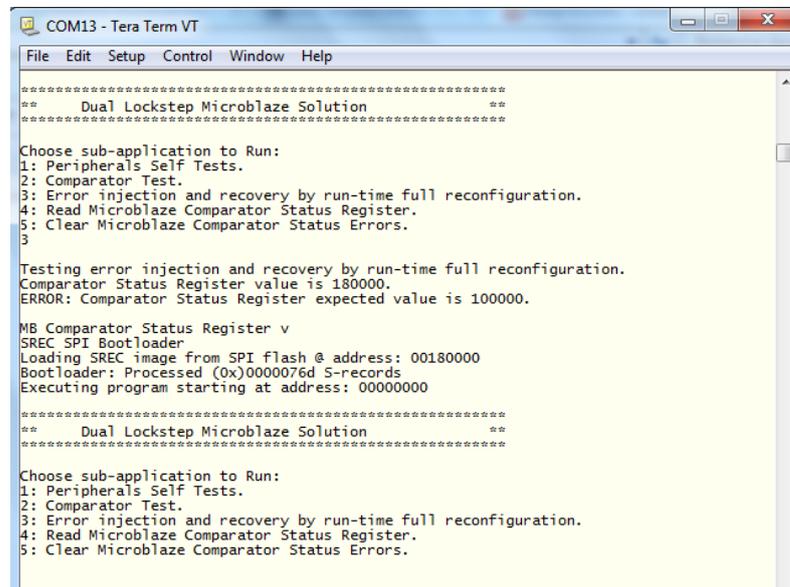
Choose sub-application to Run:
1: Peripherals Self Tests.
2: Comparator Test.
3: Error injection and recovery by run-time full reconfiguration.
4: Read Microblaze Comparator Status Register.
5: Clear Microblaze Comparator Status Errors.
5

MB Comparator Status Register value after clear is 0.
*****
** Dual Lockstep Microblaze Solution **
*****

Choose sub-application to Run:
1: Peripherals Self Tests.
2: Comparator Test.
3: Error injection and recovery by run-time full reconfiguration.
4: Read Microblaze Comparator Status Register.
5: Clear Microblaze Comparator Status Errors.

```

Figure 7-40 : Test du comparateur et effacement du registre indicateur d'erreurs



```

COM13 - Tera Term VT
File Edit Setup Control Window Help
*****
** Dual Lockstep Microblaze Solution **
*****

Choose sub-application to Run:
1: Peripherals Self Tests.
2: Comparator Test.
3: Error injection and recovery by run-time full reconfiguration.
4: Read Microblaze Comparator Status Register.
5: Clear Microblaze Comparator Status Errors.
3

Testing error injection and recovery by run-time full reconfiguration.
Comparator Status Register value is 180000.
ERROR: Comparator Status Register expected value is 100000.

MB Comparator Status Register v
SREC SPI Bootloader
Loading SREC image from SPI flash @ address: 00180000
Bootloader: Processed (0x)0000076d S-records
Executing program starting at address: 00000000

*****
** Dual Lockstep Microblaze Solution **
*****

Choose sub-application to Run:
1: Peripherals Self Tests.
2: Comparator Test.
3: Error injection and recovery by run-time full reconfiguration.
4: Read Microblaze Comparator Status Register.
5: Clear Microblaze Comparator Status Errors.

```

Figure 7-41 : Injection et récupération d'erreur par reconfiguration dynamique complète

5.3.6. Test des fonctionnalités Multiboot/fallback

La *figure 7-42* reprend l'ensemble du processus permettant la mise en œuvre et le test des fonctionnalités Multiboot/Fallback.

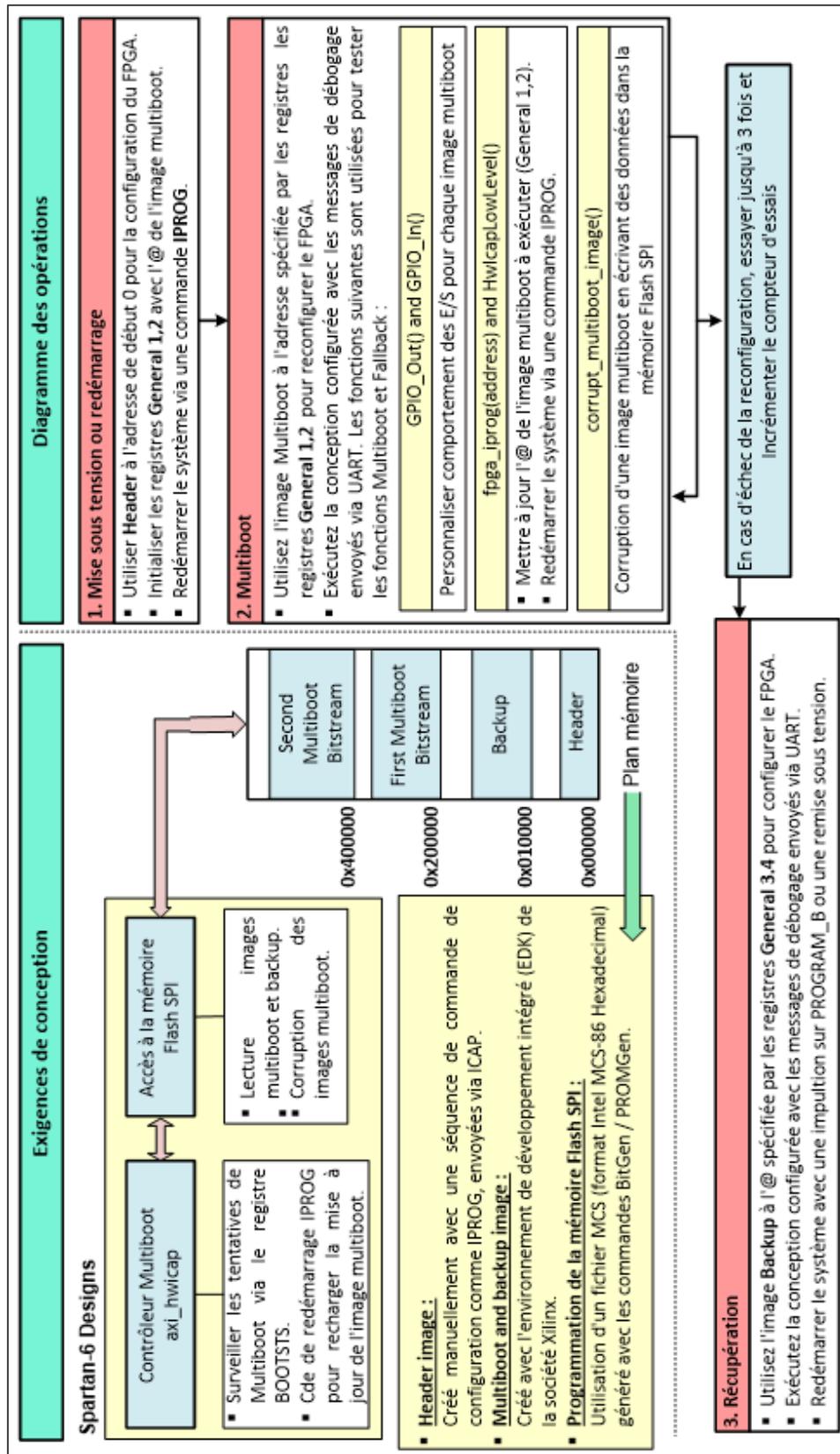


Figure 7-42 : Modèle logique de processus d'implémentation du multiboot/fallback

Pour programmer la mémoire flash SPI, nous avons généré un fichier image au format MCS (Intel SMC-86 Hexadecimal Format) qui est contient des chaînes ASCII définissant l'adresse de stockage des différents fichiers de données. Il est créé à partir des fichiers suivants, en utilisant les commandes BitGen / PROMGen [69] :

- Le fichier Header au format hexadécimal (HEX) que nous avons créé en utilisant un éditeur hexadécimal (Shareware HexEdit) pour le remplir avec les séquences des commandes de configuration décrites dans le *tableau 7-11*.
- Les fichiers bistream de l'image backup et des deux images Multiboot, que nous avons généré par l'environnement de développement SDK.

La *figure 7-43* résume le flux logiciel permettant de générer le fichier MCS, tandis que la *figure 7-44* montre le fichier de commande *build_script.bat* utilisé pour générer le fichier *header_golden_multiboot.mcs* final permettant de programmer la mémoire Flash SPI suivant le plan mémoire de la *figure 7-32*.

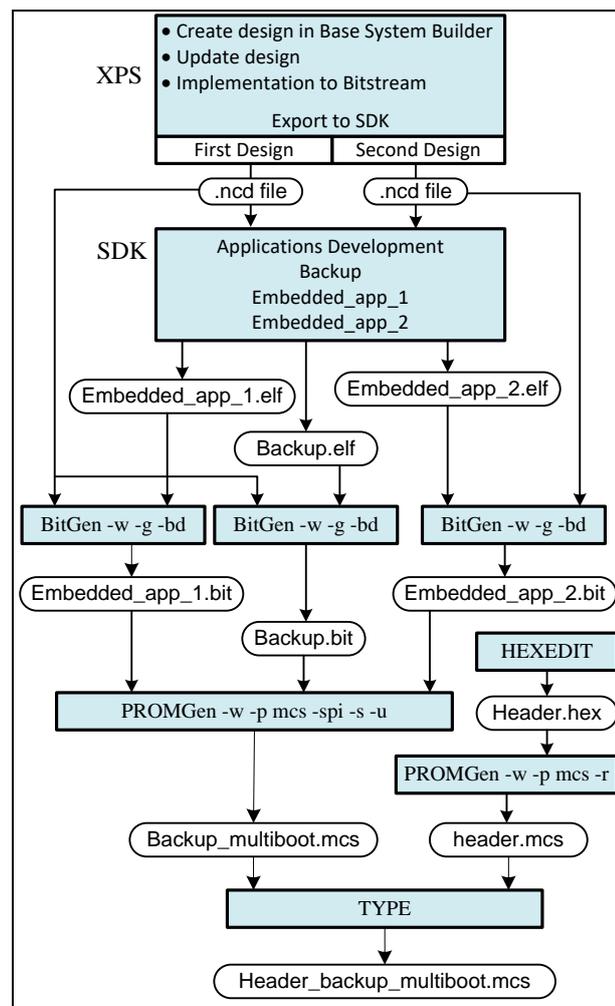


Figure 7-43 : Flux des opérations logicielles pour créer le fichier .mcs

```
cd <project_path>;implementation
bitgen -w -g next_config_register_write:Disable -g reset_on_err:Yes -g spi_buswidth:1 -bd <Project_path>\Debug\Embedded_app_1.elf system.ncd Embedded_app_1.bit
bitgen -w -g next_config_register_write:Disable -g reset_on_err:Yes -g spi_buswidth:1 -bd <Project_path>\Debug\Embedded_app_2.elf system.ncd Embedded_app_2.bit
bitgen -w -g next_config_register_write:Enable -g reset_on_err:Yes -g spi_buswidth:1 -bd <Project_path>\Debug\Backup.elf system.ncd Backup.bit
promgen -w -p mcs -r header.hex -o header.mcs
promgen -w -p mcs -s 16384 -u 010000 Backup.bit -u 200000 Embedded_app_1.bit -u 400000 Embedded_app_2.bit -o backup_multiboot.mcs
type header.mcs backup_multiboot.mcs > header_backup_multiboot.mcs
```

Figure 7-44 : Script exécutable pour générer le fichier MCS final

Pour le test de l'implémentation, nous avons utilisés tester le logiciel iMPACT de Xilinx pour lire le registre de configuration BOOTSTS qui sauvegarde les erreurs potentielles d'une opération Multiboot :

- Le bit VALID_0 est mis à jour avec l'état actuel.
- Le bit FALLBACK_1 est mis à jour si une récupération se produit.
- Le bit OMC_ERROR_1 est mis à jour si une erreur du chien de garde survient.
- Le bit CRC_ERROR_1 est mis à jour si une erreur CRC se produit.

La **figure 7-45** montre le résultat des messages de débogage de l'application de test, la **figure 7-46** montre le contenu du registre BOOTSTS après une reconfiguration réussie avec une image Multiboot et la **figure 7-47** montre le contenu du même registre en cas de récupération avec un échec de configuration avec une image multiboot.

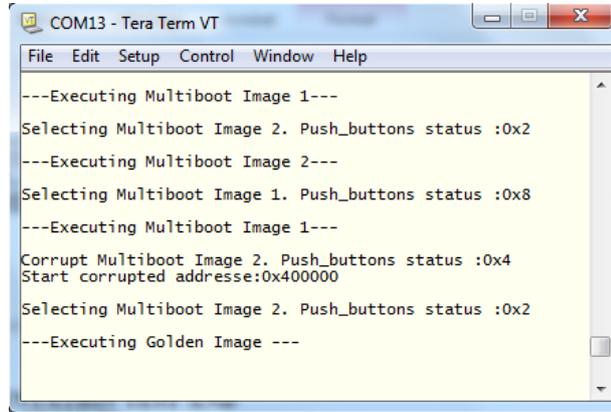


Figure 7-45 : Script exécutable pour générer le fichier MCS final

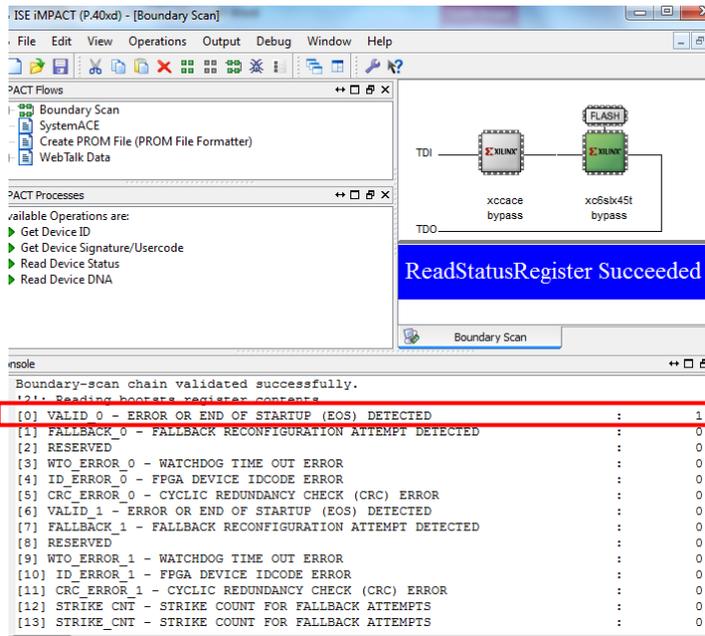


Figure 7-46 : L'état du registre BOOTSTS après une reconfiguration réussie

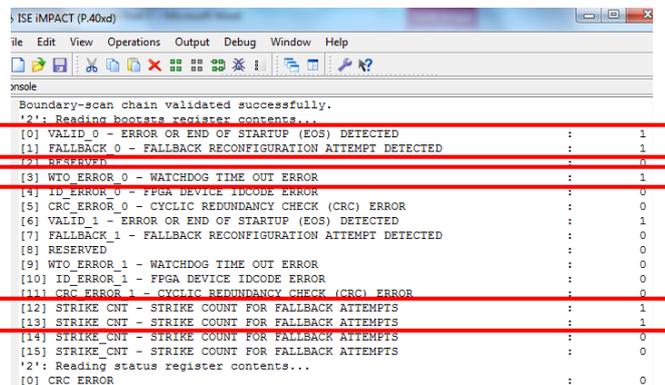


Figure 7-47 : L'état du registre BOOTSTS après une opération fallback

5.4. Implémentation de la mise à jour à distance de l'OdB secondaire

Pour augmenter la flexibilité de notre OdB secondaire reconfigurable, nous avons mis en œuvre une méthode relativement sûre de mise à jour à distance de l'architecture matérielle et logicielle du système afin de pouvoir prendre en charge les exigences d'une mission scientifique dynamique.

La solution du point de vue système, utilise une méthode basée sur le soft processeur MicroBlaze qui contrôle le canal de mise à jour à distance (Ethernet) pour télécharger les fichiers bitstream des configurations matérielles et des applications logicielles. Les composants fonctionnels de l'approche adoptée sont illustrés par la *figure 7-48* :

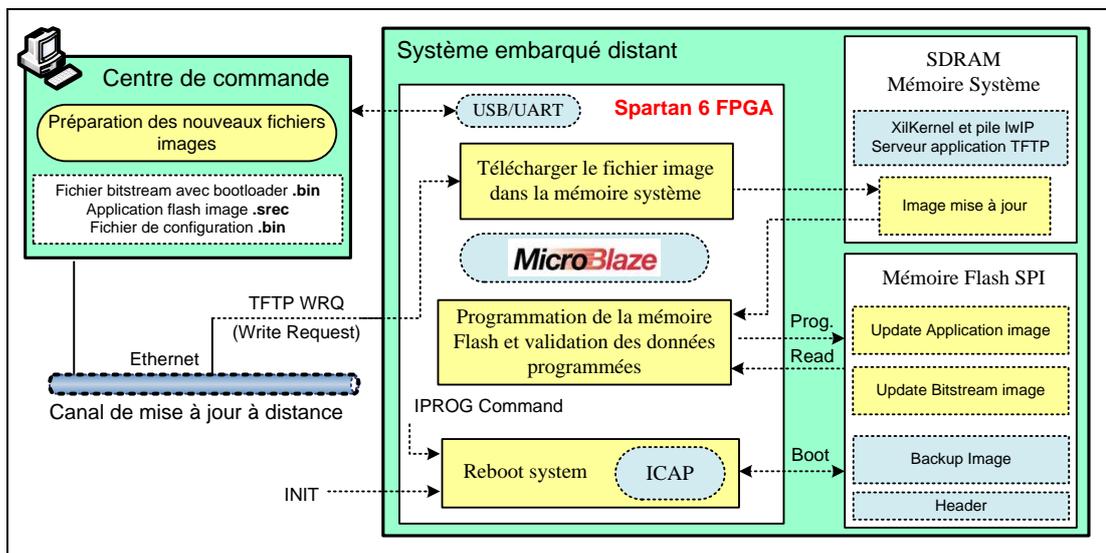


Figure 7-48 : Description fonctionnelle de la méthode de mise à jour distante

Le système distant (OdB secondaire) utilise le protocole TFTP pour recevoir des fichiers d'image matériels et logiciels mis à jour via une interface Ethernet. Le soft processeur Microblaze stocke, dans un premier temps, les données du fichier reçu dans la mémoire système SDRAM, en utilisant le contrôleur SDRAM. Une fois que le système a validé les données reçues par le canal de mise à jour, le processeur utilise ensuite un algorithme spécifique pour écrire ces données à l'adresse appropriée de la mémoire flash SPI. Une fois la programmation de mémoire Flash terminée, le processeur peut déclencher à distance la reconfiguration du FPGA en redémarrant le système. La communication via l'interface UART permet le débogage et la supervision du processus.

L'implémentation de cette mise à jour distante se fera sur le même système embarqué référence présenté à la *figure 3-34* du § 5.3.4.

5.4.1. Préparation des fichiers d'images mis à jour

Étant donné que notre conception est basée sur un soft processeur, nous devons prendre en charge aussi bien la mise à jour de la configuration matériel du FPGA, que celle du code logiciel que le processeur est censé exécuter. Puisque ce code logiciel est exécuté à partir de la mémoire système SDRAM externe, nous devons préparer deux fichiers :

- Un fichier bitstream concernant la configuration matérielle (format .bin) avec un bootloader.
- Un fichier image flash de l'application logicielle au format S-record (.srec).

La **figure 7-49** résume le flux logiciel pour la préparation de ces fichiers. Il est à noter que pour la programmation de la mémoire flash avec une image bitstream (format .bit), nous devons le convertir au format .bin, qui contient des données de configuration binaires sans informations d'en-tête.

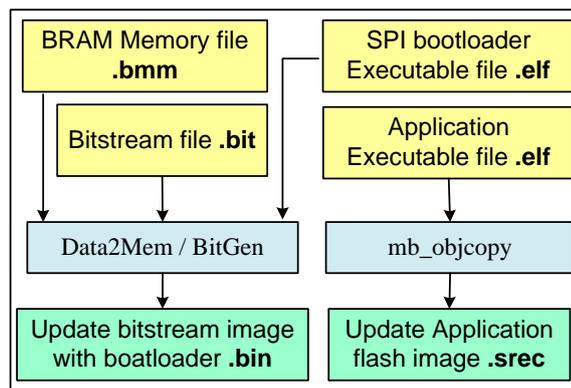


Figure 7-49 : Flux logiciel pour la préparation des fichiers images mis à jour

5.4.2. Téléchargement du fichier image dans la mémoire système SDRAM

Nous avons utilisé la pile de protocole Lightweight IP (LwIP) pour implémenter un serveur TFTP dans l'OdB secondaire distant. Ce choix réduit l'utilisation des ressources de FPGA Spartan-6 avec l'utilisation de quelques dizaines de kilo Octets de mémoire SDRAM et environ 40 kO de code dans la mémoire Flash SPI.

L'environnement de développement EDK offre plusieurs versions personnalisées de LwIP pour fonctionner sur des systèmes embarqués basés sur MicroBlaze. La dernière version 1.4 prend en charge le protocole UDP nécessaire, ainsi que le mode d'utilisation SOCKET API (Application Programming Interface) qui fournit une interface socket BSD (Berkeley Software Distribution) très portable et nécessitant l'utilisation d'un RTOS.

En effet, l'utilisation d'un RTOS pour la mise en œuvre de tâches concernant la gestion du protocole TFTP et le contrôle du segment spatial MASAT1 est beaucoup plus intuitive, facilite la gestion du temps partagé entre les tâches et augmente la réactivité. Dans notre cas, nous avons choisi le système d'exploitation XiKernel avec la politique d'ordonnancement Round-robin, ce qui ne nécessite aucune exigence particulière pour le portage de la pile de protocole LwIP.

En plus, la bibliothèque logicielle correspondante à LwIP fournit des fonctions qui simplifient grandement l'utilisation des API lwIP [80]. Les principales fonctions utilisées par notre application sont résumées par le *tableau 7-14* :

Fonction	Description
void lwip_init()	Fournit la seule fonction d'initialisation pour les structure de données utilisées par LwIP.
sys_thread_t sys_thread_new()	Doit être utilisée pour créer les tâches qui utilisent LwIP
struct netif*xemac_add()	Fournit une interface unifiée pour ajouter une adresse Xilinx EMAC IP
void xemacif_input_thread()	Envoie les paquets reçus par gestionnaires d'interruption vers la tâche tcpip_thread de lwIP

Tableau 7-14 : Principales fonctions utilisées pour implémenter le serveur TFTP [80]

Pour communiquer et télécharger le fichier image via l'interface Ethernet, le serveur TFTP utilise les paquets TFTP WRQ (demande d'écriture) selon l'organigramme illustré par la *figure 7-50* et inspiré du travail réalisé par MacMahon et al. [104].

Il est à noter que pendant la procédure de téléchargement, le fichier image mis à jour est sauvegardé dans la mémoire système SDRAM par des blocs de 512 octets. C'est le rôle de la fonction *wr_packet_to_sys_mem()* qui, entre autres, va concaténer les octets reçus pour construire des mots de 32 bits avant de les écrire dans la mémoire SDRAM.

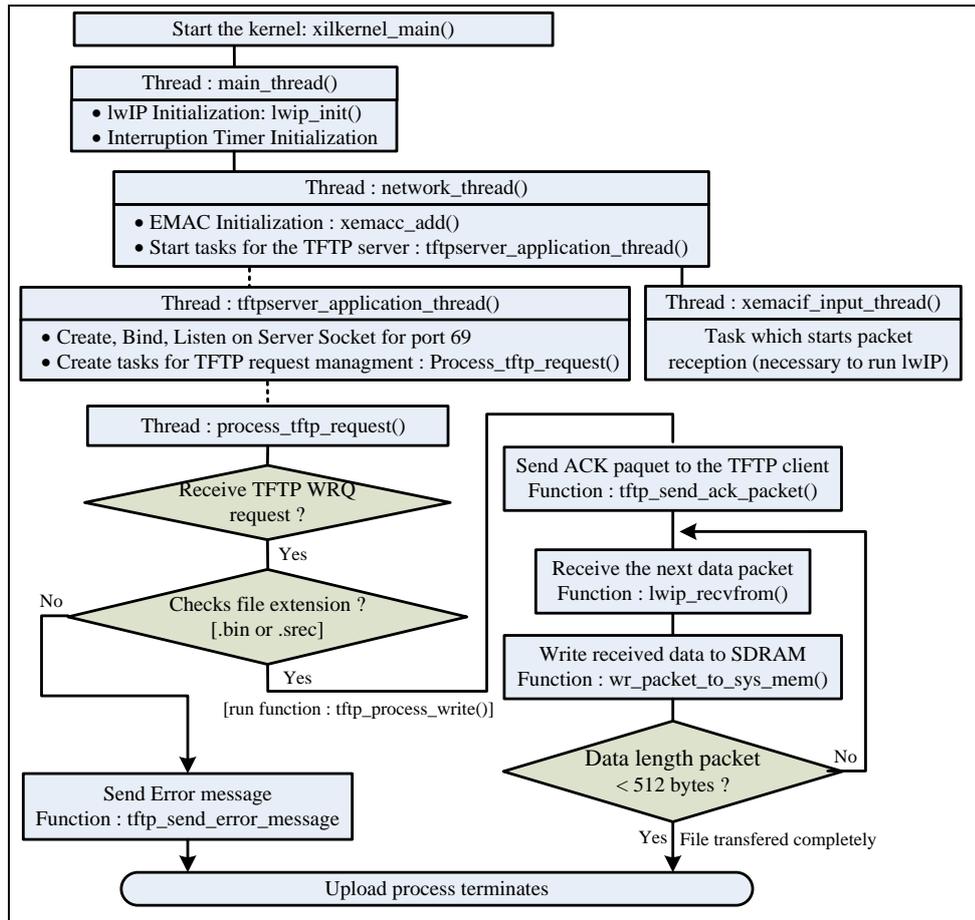


Figure 7-50 : Flux logiciel du téléchargement du fichier image via le serveur TFTP

5.4.3. Programmation de la mémoire Flash SPI

La mémoire Flash SPI externe permettra toujours d'implémenter les fonctionnalités Multiboot/Fallback décrite en § 5.3.2, mais avec une organisation mémoire une peu différente. En effet, la nouvelle organisation présentée par la *figure 7-51* prend en compte la présence d'un fichier image représentant le code logiciel destiné à être téléchargé par le bootloader dans la mémoire système SDRAM, afin d'être exécuté par le soft processeur MicroBlaze. La mémoire Flash est ainsi divisée en cinq zones :

- Fichier *Header* pour à l'implémentation des fonctionnalités Multiboot/Fallback.
- Fichier *Backup Bootloader* utilisé pour reconfigure le FPGA en cas d'échec de la reconfiguration en mode normal avec le fichier Bootloader Application.
- Fichier *Application Bootloader* utilisé pour reconfigurer la FPGA en mode normal pour l'exécution de l'application.
- Fichier *Backup Software* qui est l'image du code logiciel Backup.
- Fichier *Application Software* qui est l'image du code logiciel de l'application à exécuter en mode normal.

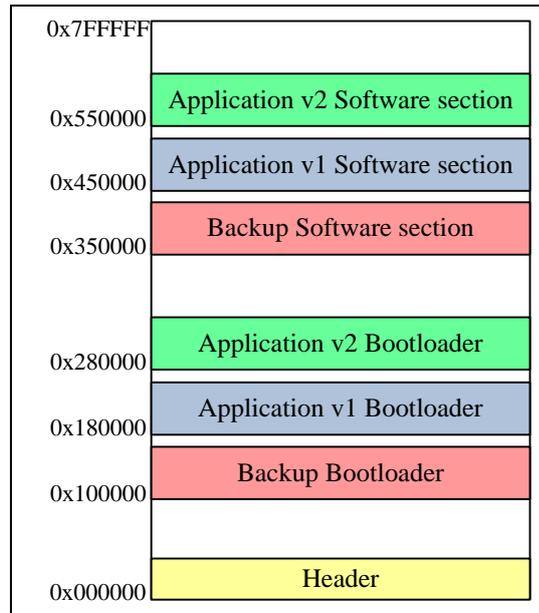


Figure 7-51 : Plan mémoire de la mémoire Flash SPI pour la mise à jour à distance

a. Programmation initiale

Pour programmer initialement la mémoire Flash SPI externe, nous avons généré un fichier MCS (Intel SMC-86 Hexadecimal Format) à partir des cinq fichiers qu'elle doit contenir (en prenant en compte uniquement la version 1 de l'application à exécuter). La *figure 7-52* montre le fichier *build_script.bat* utilisé pour générer le fichier final *header_remote_update_app.mcs*. Notez l'utilisation de :

- L'option "binary" de la commande BitGen pour créer un fichier binaire (.bin).
- L'option "-data_file" de la commande PROMGen pour ajouter un fichier de données (.srec) au fichier PROM final.

```
cd <project_path>\implementation
bitgen -w -g Binary:Yes -g spi_buswidth:1 -bd <project_path>\Appli_boot.elf system.ncd appli_boot.bit
bitgen -w -g Binary:Yes -g spi_buswidth:1 -bd <project_path>\Backup_boot.elf system.ncd backup_boot.bit
promgen -w -p mcs -r header hex -o header mcs
promgen -w -p mcs -spi -c FF -s 8192 -u 010000 <project_path>\backup_boot.bit -u 0180000 appli_boot.bit
      -data_file up 350000 backup_apps.srec -data_file up 450000 nominal_apps.srec -o remote_update_app.mcs
type header mcs remote_update_app.mcs > header_remote_update_app.mcs
```

Figure 7-52 : Script exécutable pour générer le fichier MCS final

b. Programmation du fichier d'image mis à jour

La conception adoptée utilise le pilote d'interface SPI de Xilinx en mode interrogé (pooled mode) pour communiquer avec la mémoire Flash SPI. La bibliothèque LibXil Isf (In-System Flash), incluse dans l'environnement de développement EDK, fournit les fonctions permettant d'écrire, lire et effacer le mémoire Flash.

En résumé, la programmation de la mémoire flash avec le fichier image téléchargé en mémoire SDRAM sera gérée en trois étapes :

1. Effacez les blocs mémoire flash en fonction de la taille du fichier image. En effet, l'écriture sur la mémoire flash ne peut que changer les bits de l'état 1 à 0, et l'effacement d'un bloc mémoire ou d'un secteur force tous les bits à 1, après quoi toute valeur peut être programmée.
2. Programmer la mémoire flash en lisant les données à partir de la mémoire SDRAM et en les écrivant à l'adresse correcte de la mémoire flash.
3. Validez l'opération de programmation en effectuant une opération de lecture et en comparant les données lues par rapport aux données écrites à partir de la mémoire SDRAM.

5.4.4. Déclenchement à distance de la reconfiguration dynamique du FPGA

Pour activer la nouvelle image de configuration mise à jour à distance, le système implémente et déclenche à distance, via une interface UART, le processus de reconfiguration dynamique complet décrit au § 5.3.2.

On va ainsi déclencher à distance l'envoi d'une IPROG via l'interface ICAP, en utilisant la même fonction bas niveau *HwIcapLowLevel()* du pilote *axi_hwicap*. Cette dernière sera personnalisée par la machine d'état concernant la séquence de redémarrage désirée comme indiqué précédemment dans la *figure 7-37*.

5.4.5. Mise en œuvre, test et résultats

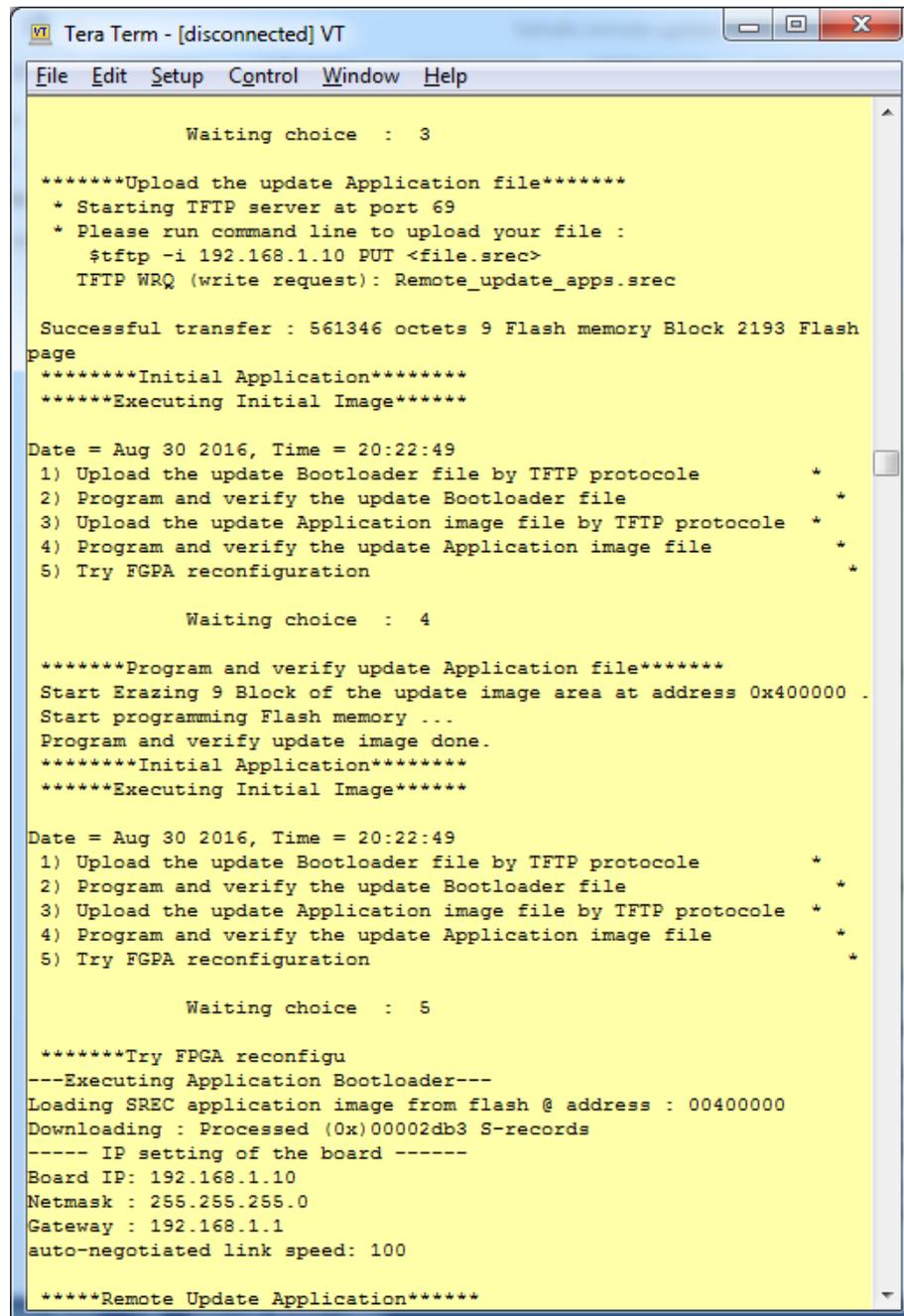
Pour la mise en œuvre et le test, nous avons créé trois applications logicielles : Initial, update et backup_application. Pour toutes les applications, nous avons utilisé le même BSP après avoir ajouté et configuré les bibliothèques Xilkernel, Xilisf et Lwip140.

L'exécution de chaque application de test comprend des messages personnalisés envoyés via UART et les adresses mémoires des fichiers image de mis à jour.

Dans le centre de conception matérialisé par le PC distant, un programme terminal Windows est configuré pour déboguer, surveiller et redémarrer à distance l'ODB secondaire. Chaque application de test doit établir au départ une connexion Ethernet avec l'ODB secondaire distant, puis fournir un menu avec les trois options suivantes :

- Mettre à jour le fichier bootloader de configuration
- Mettre à jour le fichier image de l'application logiciel
- Tenter une reconfiguration FPGA à distance.

La *figure 7-53* montre les résultats de l'exécution de l'application initiale, avec un transfert et une programmation réussis de la mémoire Flash SPI avec le fichier image de l'application mise à jour, et enfin une reconfiguration à distance réussie avec l'application mise à jour. Si une interruption se produit pendant le processus du transfert du fichier mise à jour ou la programmation de la mémoire flash, le fichier bootloader backup est chargé lors de la prochaine reconfiguration.



```
VT Tera Term - [disconnected] VT
File Edit Setup Control Window Help

Waiting choice : 3

*****Upload the update Application file*****
* Starting TFTP server at port 69
* Please run command line to upload your file :
  $tftp -i 192.168.1.10 PUT <file.srec>
  TFTP WRQ (write request): Remote_update_apps.srec

Successful transfer : 561346 octets 9 Flash memory Block 2193 Flash
page
*****Initial Application*****
*****Executing Initial Image*****

Date = Aug 30 2016, Time = 20:22:49
1) Upload the update Bootloader file by TFTP protocole *
2) Program and verify the update Bootloader file *
3) Upload the update Application image file by TFTP protocole *
4) Program and verify the update Application image file *
5) Try FGPA reconfiguration *

Waiting choice : 4

*****Program and verify update Application file*****
Start Erasing 9 Block of the update image area at address 0x400000 .
Start programming Flash memory ...
Program and verify update image done.
*****Initial Application*****
*****Executing Initial Image*****

Date = Aug 30 2016, Time = 20:22:49
1) Upload the update Bootloader file by TFTP protocole *
2) Program and verify the update Bootloader file *
3) Upload the update Application image file by TFTP protocole *
4) Program and verify the update Application image file *
5) Try FGPA reconfiguration *

Waiting choice : 5

*****Try FPGA reconfigu
---Executing Application Bootloader---
Loading SREC application image from flash @ address : 00400000
Downloading : Processed (0x)00002db3 S-records
----- IP setting of the board -----
Board IP: 192.168.1.10
Netmask : 255.255.255.0
Gateway : 192.168.1.1
auto-negotiated link speed: 100

*****Remote Update Application*****
```

Figure 7-53 : Fenêtre terminal Windows affichant l'exécution de l'application initiale

6. CONCLUSIONS ET PERSPECTIVES

Dans le contexte de la partie de ce travail de recherche visant à concevoir un OdB secondaire reconfigurable à base de FPGA type SRAM, nous avons pu :

- Développer une architecture de base pour un OdB auto-reconfigurable dynamiquement en se basant sur l'environnement de développement EDK et sur les solutions technologiques de reconfiguration de Xilinx comme l'interface ICAP ou l'approche "Lockstep" pour la duplication matérielle. Mais, avant la validation expérimentale de cette solution par injection d'erreurs, nous pouvons comme perspective future :
 - Utiliser un FPGA Spartan-6 de la série XC6SLX75 ou XC6SLX150 pour pouvoir implémenter la technique IDF (Xilinx Isolation Design Flow) permettant de diviser le système implémenter dans le FPGA en plusieurs zones et fonctions isolées entre elles.
 - Utiliser le contrôleur SEM (Soft Error Mitigation) qui est une solution configurée automatiquement pour détecter et corriger les erreurs logicielles dans la mémoire de configuration des FPGA Xilinx.
- Maitriser le processus de mise à jour à distance de l'architecture matérielle et logicielle de notre OdB secondaire reconfigurable, via un lien Ethernet. Mais dans la perspective d'implémenter le processus dans la mission MASAT1, il reste à améliorer :
 - La conception du système du point de vue sécurité en améliorant la confidentialité et l'intégrité des fichiers images mis à jour.
 - Adaptez la méthode pour utiliser le canal de communication radiofréquence à la place du lien Ethernet.

CONCLUSION GENERALE

Le travail de recherche présenté dans ce mémoire traite de l'étude et de la conception de MASAT1, le premier nanosatellite universitaire marocain basé sur le standard Cubesat. Ce dernier est constitué de :

- Une plateforme composée de plusieurs sous-systèmes organisés sous forme d'une architecture distribuée avec I²C comme bus de communication série. Les sous-systèmes abordés par cette thèse sont l'Energie de Bord, Communication, Ordinateur de Bord, Structure et contrôle d'attitude.
- Une charge utile principale qui consiste à concevoir un ordinateur de bord secondaire hautement intégré et reconfigurable à base de FPGA à mémoire SRAM. Le but étant de proposer une alternative à faible coût, fiable et reconfigurable pour remplacer les architectures matérielles traditionnelles basées sur les microcontrôleurs.

Vu les spécificités et les contraintes des nanosatellites, l'approche traditionnelle adoptée pour la conception et le développement des satellites a été remplacée par une approche d'ingénierie système partiellement basée aussi bien sur les standards ECSS, que sur les directives du manuel d'ingénierie système de la NASA.

Concernant le choix des solutions technologiques adoptées pour la plateforme MASAT1 et sa charge utile principale, nous avons opté pour l'utilisation de composants COTS, ce qui a permis de diminuer les coûts tout en conservant la fiabilité nécessaire pour une mission en orbite LEO.

En plus de décrire les étapes fondamentales dans la conception du nanosatellite universitaire MASAT1, ce mémoire de thèse permettra de partager avec les futurs chercheurs marocains, les connaissances acquises au niveau de :

- L'application des standards ECSS au niveau du management d'une mission spatiale aux standards Cubesat.
- L'état de l'art de la dynamique orbitale permettant de réaliser une étude orbitale dont la finalité est de déterminer les orbites optimales du segment spatial ainsi que les paramètres déterminants dans la conception des différents sous-systèmes qui le constituent.
- La définition des solutions de conception en se basant sur une approche descendante permettant de traduire les objectifs d'une mission en sous-systèmes, composants matériels et tâches logicielles.

Cette thèse a permis aussi l'application des techniques de reconfiguration dynamique propres aux FPGA type SRAM, pour concevoir un ordinateur de bord secondaire reconfigurable à base d'un FPGA faible consommation comme le Spartan-6 de Xilinx. Les travaux de test et de simulation ont relativement montré la faisabilité et l'utilité d'une telle approche pour les applications nanosatellites ; ceci en termes de gain en puissance de calcul, en densité d'intégration, en temps de développement et en flexibilité.

Concernant les perspectives à venir, ce projet permettra de rendre accessible l'exploration scientifique spatiale aux universités Marocaines en mettant à disposition une littérature complète et détaillée concernant l'état de l'art et les étapes de conception des nanosatellites.

Nous pouvons ainsi mettre en place à l'Université Sidi Mohamed De Fès, un centre de compétences dans le domaine des nanosatellites dont la finalité sera :

- La promotion de la R&D dans le domaine de l'ingénierie spatiale.
- Le lancement d'autres missions cubesat 3U qui se baseront sur le succès futur de la mission MASAT1.
- La mise en place de collaboration scientifique avec des structures universitaires internationales et des opérateurs privés.

ANNEXES

ANNEXE A

**Tableau des données CIRA72 d'intervalle d'altitude pour la densité
atmosphérique [23]**

Altitude Interval, km $[h_i - h_{(i+1)}]$	Scale Height, km H_i	Base Density, kg/m ³ ρ
0 - 25	7.249	1.225E+00
25 - 30	6.349	3.899E-02
30 - 40	6.682	1.774E-02
40 - 50	7.554	3.972E-03
50 - 60	8.382	1.057E-03
60 - 70	7.714	3.206E-03
70 - 80	6.549	8.770E-05
80 - 90	5.799	1.905E-05
90 - 100	5.382	3.396E-06
100 - 110	5.877	5.297E-07
110 - 120	7.263	9.661E-08
120 - 130	9.473	2.438E-08
130 - 140	12.636	8.484E-09
140 - 150	16.149	3.845E-09
150 - 180	22.523	2.070E-09
180 - 200	29.74	5.464E-10
200 - 250	37.105	2.789E-10
250 - 300	45.546	7.248E-11
300 - 350	53.628	2.418E-11
350 - 400	53.298	9.518E-12
400 - 450	58.515	3.725E-12
450 - 500	60.828	1.585E-12
500 - 600	63.822	6.967E-13
600 - 700	71.835	1.454E-13
700 - 800	88.667	3.614E-14
800 - 900	124.64	1.170E-14
900 - 1000	181.05	5.245E-15

ANNEXE B

Code source de certains scripts et fonctions Matlab permettant l'évaluation des perturbations orbitales

1. Fonction de conversion du vecteur d'état \vec{r} et \vec{v} vers les éléments orbitaux

```
function OE=PV2OE(R,V)

%r2d - transformation from radian to degree
% mu - gravitational parameter (km^3/s^2)
% R - position vector in the geocentric equatorial frame(km)
% V - velocity vector in the geocentric equatorial frame(km)
% r, v - the magnitudes of R and V
% vr - radial velocity component (km/s)
% H - the angular momentum vector (km^2/s)
% h - the magnitude of H (km^2/s)
% incl - inclination of the orbit (rad)
% N - the node line vector (km^2/s)
% n - the magnitude of N
% cp - cross product of N and R
% RA - right ascension of the ascending node (rad)
% E - eccentricity vector
% e - eccentricity (magnitude of E)
% eps - a small number below which the eccentricity is
% considered to be zero
% w - argument of perigee (rad)
% TA - true anomaly (rad)
% a - semimajor axis (km)

mu=398600;
eps = 1.e-3;
r = norm(R);
v = norm(V);
vr = dot(R,V)/r;
H = cross(R,V);
h = norm(H);

% inclination determination
incl = acos(H(3)/h);
% vector of node line
N = cross([0 0 1],H);
n = norm(N);

% RAAN determination
if n~=0
    RA = acos(N(1)/n);
    if N(2) < 0
        RA = 2*pi - RA;
    end
else
    RA=0;
End

% Eccentricity determination
E = 1/mu*((v^2 - mu/r)*R - r*vr*V);
e = norm(E);
```

```

% argument og perige determination
if n~=0
    if e > eps
        w = acos(dot(N,E)/n/e);
        if E(3) < 0
            w=2*pi-w;
        end
    else
        w=0;
    end
else
    w=0;
end

% true anomaly determination
if e > eps
    TA = acos(dot(E,R)/e/r);
    if vr < 0
        TA = 2*pi - TA;
    end
else
    cp = cross(N,R);
    if cp(3) >= 0
        TA = acos(dot(N,R)/n/r);
    else
        TA = 2*pi - acos(dot(N,R)/n/r);
    end
end

% semi major axis determination
a = h^2/mu/(1 - e^2);
% radian to degree
r2d=180/pi;

OE=[ a ; e ; incl*r2d ; w*r2d ; RA*r2d ; TA*r2d ];

```

2. Evaluation de l'effet de l'aplatissement de la terre (Propagateur J2)

```

function k1=module1(t,y1)
mue=398600.4415;
k1=mue*norm(y1)^-3;

function k2=module2(t,y1)
mue=398600.4415;
j=0.00108263;
Rt=6378;
r=norm(y1);
fx=((5*y1(3)^2-r^2)*1.5*j*mue*Rt^2)/r^7;
fy=fx;
fz=((5*y1(3)^2-3*r^2)*1.5*j*mue*Rt^2)/r^7;
k2=[fx 0 0;0 fy 0;0 0 fz];

function dy=j2(t,y)
y1=y([1:3],1);
y2=y([4:6],1);
dy1=y2;
dy2=-module1(t,y1)*y1+module2(t,y1)*y1;
dy=[dy1;dy2];

```

```

sim1.m
%r0 is initial state position and velocity in ECI
r0=[4670.2 ; 4084.2 ; 3302.2 ; -4.4436 ; 0.0498 ; 6.2229];
%tf end simulation time
tf=14*6150; % pas simulation step
pas=1;
options = odeset('RelTol',1e-6);
[t,y] = ode23(@j2,[1:pas:tf],r0,options);
OE=zeros(tf/pas,6);
for i=1:tf/pas
    OE(i,:)=PV2OE(y(i,[1:3]),y(i,[4:6]));
end

acc=zeros(tf/pas,1);
for i=1:tf/pas
    vacc=module2(t,y(i,[1:3]))*y(i,[1:3]);
    acc(i)=norm(vacc);
end

%plotting
figure('Color',[1 1 1]);
subplot(2,3,1),plot(OE(:,1))
set(gca,'XLim',[1 86200/3600],'Ylim',[OE(1,1)-100 OE(1,1)+100]);
grid on
title('Semi-major axis')
xlabel('Time (s)')
ylabel('Km')

subplot(2,3,2),plot(OE(:,2))
set(gca,'XLim',[1 86200],'Ylim',[OE(1,2)-0.1 OE(1,2)+0.1]);
grid on
title('Excentricity')
xlabel('Time (hr)')
ylabel('')

subplot(2,3,3),plot(OE(:,3))
set(gca,'XLim',[1 86200],'Ylim',[OE(1,3)-10 OE(1,3)+10]);
grid on
title('Inclination')
xlabel('Time (s)')
ylabel('(Degree)')

subplot(2,3,4),plot(OE(:,5))
set(gca,'XLim',[1 86200],'Ylim',[OE(1,5)-10 OE(1,5)+10]);
grid on
title('RAAN')
xlabel('Time (s)')
ylabel('(Degree)')

subplot(2,3,5),plot(OE(:,4))
set(gca,'XLim',[1 86200],'Ylim',[OE(1,4)-10 OE(1,4)+10]);
grid on
title('Perige Argument')
xlabel('Time (s)')
ylabel('(Degree)')

subplot(2,3,6),plot(OE(:,6))
set(gca,'XLim',[1 86200]);
grid on
title('True Anomalie')
xlabel('Time (s)')
ylabel('Degree')

```

3. Effet de la traînée atmosphérique

```
function d=density(t,y1)
% Rt == earth radius and r == orbit radius
% h == orbit altitude and d == density
% d0 == reference density and h0 == reference altitude
% H == height scale is the change in altitude, in km, for the atmospheric
% density to decrease by a factor of e

Rt=6378.1363;
r=sqrt(y1(1)^2+y1(2)^2+y1(3)^2);
h=r-Rt;
if h>=1000
    d0=3.019E-15;
    H=268.000;
    h0=1000;
elseif h<1000 && h>=900
    d0=5.245E-15;
    H=181.050;
    h0=900;
elseif h<900 && h>=800
    d0=1.170E-14;
    H=124.640;
    h0=800;
elseif h<800 && h>=700
    d0=3.614E-14;
    H=88.667;
    h0=700;
elseif h<700 && h>=600
    d0=1.454E-13;
    H=71.835;
    h0=600;
elseif h<600 && h>=500
    d0=6.967E-13;
    H=63.822;
    h0=500;
elseif h<500 && h>=450
    d0=1.585E-12;
    H=60.828;
    h0=450;
elseif h<450 && h>=400
    d0=3.725E-12;
    H=58.515;
    h0=400;
elseif h<400 && h>=350
    d0=9.518E-12;
    H=53.298;
    h0=350;
elseif h<350 && h>=300
    d0=2.418E-11;
    H=53.628;
    h0=300;
elseif h<300 && h>=250
    d0=7.248E-11;
    H=45.546;
    h0=250;
elseif h<250 && h>=200
    d0=2.789E-10;
    H=37.105;
    h0=200;
elseif h<200 && h>=180
    d0=5.464E-10;
    H=29.740;
```

```

    h0=180;
elseif h<180 && h>=150
    d0=2.070E-09;
    H=22.523;
    h0=150;
elseif h<150 && h>=140
    d0=3.845E-09;
    H=16.149;
    h0=140;
elseif h<140 && h>=130
    d0=8.484E-09;
    H=12.636;
    h0=130;
elseif h<130 && h>=120
    d0=2.438E-08;
    H=9.473;
    h0=120;
elseif h<120 && h>=110
    d0=9.661E-08;
    H=7.263;
    h0=110;
elseif h<110 && h>=100
    d0=5.297E-07;
    H=5.877;
    h0=100;
elseif h<100 && h>=90
    d0=3.396E-06;
    H=5.382;
    h0=90;
elseif h<90 && h>=80
    d0=1.905E-05;
    H=5.799;
    h0=80;
elseif h<80 && h>=70
    d0=8.770E-05;
    H=6.549;
    h0=70;
elseif h<70 && h>=60
    d0=3.206E-04;
    H=7.714;
    h0=60;
elseif h<60 && h>=50
    d0=1.057E-03;
    H=8.382;
    h0=50;
elseif h<50 && h>=40
    d0=3.972E-03;
    H=7.554;
    h0=40;
elseif h<40 && h>=30
    d0=1.774E-02;
    H=6.682;
    h0=30;
elseif h<30 && h>=25
    d0=3.899E-02;
    H=6.349;
    h0=25;
elseif h<25 && h>=0
    d0=1.225E+00;
    H=7.249;
    h0=0;
end
d=d0*exp((h0-h)/H);

```

```

function k=module(t,y1)
mue=398600.4415;
k=mue*norm(y1)^-3;

function V=Vrel(t,y1,y2)
w=7.27e-5;
Vvec=[y2(1)+w*y1(2);y2(2)-w*y1(1);y2(3)];
V=norm(Vvec);

function k=matrice()
w=7.27e-5;
k=[0 -w 0;w 0 0;0 0 0];

function dy=drag(t,y)
y1=y([1:3],1);
y2=y([4:6],1);
dy1=y2;
dy2=-module(t,y1)*y1-
1e6*0.5*2.2*(0.014/1.33)*density(t,y1)*Vrel(t,y1,y2)*(y2-matrice*y1);
dy=[dy1;dy2];

sim2.m % Script de simulation
r0=[4670.2 ; 4084.2 ; 3302.2 ; -4.4436 ; 0.0498 ; 6.2229];
tf=14*6150;
pas=1;
options = odeset('RelTol',1e-6);
[t,y] = ode23(@drag,[0:pas:tf],r0,options);

% from PV to OE
OE=zeros(tf/pas,6);
for i=1:tf/pas
    OE(i,:)=PV2OE(y(i,[1:3]),y(i,[4:6]));
end

% acc perturbatrice
acc=zeros(tf/pas,1);
for i=1:tf/pas

vacc=1e6*0.5*2.2*(0.015/1.33)*density(t,y(i,[1:3]))*Vrel(t,y(i,[1:3]),y(i,
[4:6]))*(y(i,[4:6])'-matrice*y(i,[1:3]'));
    acc(i)=norm(vacc);
end

% plotting
figure('Color',[1 1 1]);
subplot(2,3,1),plot(OE(:,1),'r')
set(gca,'XLim',[1 86200]);
grid on
title('Semi-major axis')
xlabel('Time (s)')
ylabel('Km')

subplot(2,3,2),plot(OE(:,2))
set(gca,'XLim',[1 86200]);
grid on
title('Excentricity')
xlabel('Time (s)')
ylabel('')

```

```

subplot(2,3,3),plot(OE(:,3))
set(gca,'XLim',[1      86200]);
grid on
title('Inclination')
xlabel('Time (s)')
ylabel('Degree')

subplot(2,3,4),plot(OE(:,5))
set(gca,'XLim',[1      86200]);
grid on
title('RAAN')
xlabel('Time (s)')
ylabel('Degree')

subplot(2,3,5),plot(OE(:,4))
set(gca,'XLim',[1      86200]);
grid on
title('Perige Argument')
xlabel('Time (s)')
ylabel('Degree')

subplot(2,3,6),plot(OE(:,6))
set(gca,'XLim',[1      86200]);
grid on
title('True Anomalie')
xlabel('Time (s)')
ylabel('Degree')

```

ANNEXE C

Benchmark fonctionnel des OdB de différentes missions Cubesat

Mission	Date de lancement et orbite	Microcontrôleur de l'OdB	Mémoires	OS	Bus	Status
AAUSAT-1	2003 820km – 98° Sun Synch	Siemens C161-RI (8051) 16-Bit CISC & RISC CPU 12,5 MHz	RAM : 4Mb PROM : 256Kb Flash : 256Kb	RTX166	I2C	Echoué (Battery problem)
QuakeSat	2003 820km – 98° Sun Synch	Prometheus PC-104 (ZFx86) 100MHz to 66MHz	SRAM : 32MB Flash : 128MB	Red Hat 9 Linux	-	Succès
CanX-1	2003 820km – 98° Sun Synch	ATMEL ARM7 32-Bit RISC CPU 40MHz	SRAM : 2MB Flash : 32MB EPROM : 128kB	eCos	-	Echoué (Pas de signal reçu)
Cubesat XI-IV	2003 820km – 98° Sun Synch	PIC16F877 4MHz	RAM: 368 Byte ROM: 32KB	Non	I2C	Succès
DTUSat-1	2003 820km – 98° Sun Synch	AT91M40800 32-Bit RISC CPU 16MHz	RAM : 1MB ROM : 16MB Flash : 2MB	eCos	SPI	Echoué (Pas de signal reçu)
UWE-1	2005 700km Sun Synch	H8S-2674R 32-bit Super-H RISC CPU	RAM : 8MB ROM : 4.5MB	µCLinux	-	Succès (Partiel)
GeneSat-1	2006 460km – 40,5°	PIC18	Flash	Non	I2C	Succès
Cute-1.7 + APDII	2006 299 x 712km 98°	Hitachi PDA NPD- 20JWL 32-Bit ARM CPU ARMV4I 400MHz	RAM : 32MB Flash : 128MB	Windows CE.NET	USB	Succès
Polysat CP1	2006 523km Sun Synch	PIC18LF6720 8-Bit PIC RISC CPU 4 MHz	RAM : 4Kb ROM : 1Kb Flash : 128Kb	Custom	I2C	Echoué (Lancement)
Polysat CP2	2006 523km Sun Synch	PIC18LF6720 8-Bit PIC RISC CPU 4 MHz	RAM : 4Kb ROM : 1Kb Flash : 128Kb	Custom	I2C	Echoué (Lancement)
MEROPE	2006 523km Sun Synch	MC68HC812A4 16-Bit ARM RISC CPU 8 MHz	RAM : 1KB ROM : 154KB	Custom	-	Echoué (Lancement)
KUTESat	2006 523km Sun Synch	PIC18F4220 8-Bit PIC RISC CPU 8 MHz	RAM : 512MB ROM : 4kB Flash : 2MB	-	I2C	Echoué (Lancement)
Ncube-1	2006 523km Sun Synch	Atmel AVR ATmega 32L 8-Bit AVR RISC CPU 4 MHz	RAM : 2kB ROM : 32kB Flash : 1kB	-	I2C	Echoué (Lancement)
RinconSat	2006 523km Sun Synch	PIC16C77 8-Bit PIC RISC CPU 4 MHz	RAM : 368Byte FRAM : 64kB	-	I2C	Echoué (Lancement)
Polysat CP3	2007 650*800km Sun Synch	AT91SAM9G20 32-bit ARM RISC CPU 400 MHz	RAM : 64MB ROM : 528MB	Linux	I2C	Succès
Polysat CP4	2007 650*800km Sun Synch	AT91SAM9G20 32-bit ARM RISC CPU 400 MHz	RAM : 64MB ROM : 528MB	Linux	I2C	Succès

Mission	Date de lancement et orbite	Microcontrôleur de l'OdB	Mémoires	OS	Bus	Status
AAUSAT-II	2008 630km Helio synch	AT91SAM7A1 32-Bit ARM7 RISC CPU 8/40 MHz	RAM : 2MB Flash : 2*4MB	eCos RedBoot	CAN	Succès
Delphi-C3	2008 630km Helio synch	TI MSP430 16-Bit ARM RISC CPU 8 MHz	RAM : 10KB Flash : 55KB	Salvo	I2C	Succès
COMPASS-1	2008 Helio synch	C8051F123 8-Bit ARM RISC CPU 24 MHz	RAM : 8KB Flash : 16MB	-	I2C	Succès
CanX-2	2008 635 km 97,4°	ARM7 32-bit ARM RISC CPU 15MHz	SRAM : 6MB Flash : 16 MB	CANOE	-	Succès
SwissCube	2009 726*752km Sun Synch	AT91M558800A 32-bit ARM v5 RISC CPU	SRAM : 1MB Flash soft 2MB Flash data 8MB	eCos	I2C	Succès
BeeSat-1	2009 726*752km	ARM7 LCP2292 32-bit ARM RISC CPU 60MHz	RAM : 2MB ROM : 16MB	TinyBOSS	CAN	Succès
ITUpSAT	2009 726*752km	TI MSP430 16-Bit ARM RISC CPU 8 MHz	RAM : 10KB Flash : 55KB	Salvo 4	I2C	Succès
StudSat-1	2010 700km Sun Synch	AT91SAM9260 32-bit ARM RISC CPU 180 MHz	RAM : 64KB Flash : 512KB	VxWorks	I2C	Succès
MCubed-2	2011 458*816km 101°	Stamp9G20 ARM9 core 32-bit ARM RISC CPU 400 MHz	SDRAM : 54M NAND Flash 128MB	RT Linux	I2C	Succès
AubieSat-1	2011 452*755km	ATMEGA128 8-bit AVR RISC CPU	-	RTOS	-	Succès
KySat-1	2011	TI MSP430F1612 16-Bit ARM RISC CPU 8 MHz	RAM : 10KB Flash : 55KB	Salvo	I2C	Echoué (Lancement)
PW-Sat-1	2012 300*1023km 97,8°	AT91SAM7X 32-bit ARM RISC CPU 55 MIPS	RAM : 256KB Flash : 2MB	FreeRTOS	-	Succès
ROBUSTA	2012 302*1089km 69,47°	PIC 18F4680 8-Bit PIC RISC CPU 20 MHz	SRAM : 4KB Flash : 64 KB EEPROM	µC OS II	CAN	Echoué Charge Batterie
AAUSAT-3	2013 781km 98,6°	ARM7 AT90CAN128 32-bit ARM RISC CPU	RAM : 36KB Flash : 128 KB EEPROM : 4KB	FreeRTOS	CAN	Succès
ESTCube-1	2013 670km	ARM Cortex-M3 STM32F103VET 6 72MHz	Flash : 16MB FRAM : 256kB	FreeRTOS	SPI	Succès
Delfi-n3Xt	2013 599*780km 97,76°	TI MSP430F1611 16-Bit ARM RISC CPU 8 MHz	-	-	I2C	Succès
iCube-1	2013 594*641km 97,77°	TI MSP430F1612 16-Bit ARM RISC CPU 8 MHz	RAM : 5KB Flash : 55KB	-	-	Succès
STRaND-1	2013 785 km	ARM7 32-bit ARM RISC CPU 40MHz	RAM : 2 MB Flash : 8 MB	FreeRTOS	I2C	Succès
DTUSat-2	2014 630 km – 98° Sun Synch	ARM7 Philips LPC2292 32-bit ARM RISC CPU 14.7456 MHz	-	eCos	CAN	Succès
Oufi	2015 ?	TI MSP430F1612 16-Bit ARM RISC CPU 8 MHz	-	FreeRTOS	-	

ANNEXE D

Code Matlab pour déterminer les caractéristiques I-V et P-V des panneaux photovoltaïques

```
function I = Cell_GaAs(V,G,TaC)
%Model of 30% efficiency solar cell from Azurspace : I=f(V,T)
%Use of function : I = Cell_GaAs(V,G,TaC)
%V = Voltage on cell terminals [V]
%G = relative insolation [-] (G=1 => 1367 W/m^2)
%TaC = temperature of the cell in operation [Celsius]

%Boltzman constant
k = 1.38e-23;
%Electric charge
q = 1.60e-19;
%Quality factor of the diode (1<n<2)
n = 1.5;
%Band gap voltage (1.12eV < Vg < 1.757eV)
Vg = 1.75;
%Reference values
Tref = 273 + 28; %temperature
Voc_Tref = 2.69; %open circuit voltage (G=1 et T=Tref)
Isc_Tref = 0.5196; %short circuit current (G=1 et T=Tref)
%Temperature of the cell in operation
TaK = 273 + TaC;
%Photo-current thermal coefficient
K0 = 2.72e-4/Isc_Tref;
%Photo-current (G=1 et T=Tref)
Iph = Isc_Tref * G * (1 + K0*(TaK - Tref));
%Diode saturation current (T=Tref)
Id_Tref = Isc_Tref / (exp(q*Voc_Tref/(n*k*Tref))-1);
%Diode saturation current (T=TaK)
Id = Id_Tref * (TaK/Tref)^(3/n) * exp(-(Vg * q/(n*k))*(1/TaK - 1/Tref));
%Calculation of serie resistance Rs
Vpmax_Tref = 2.409; %voltage at maximum power for T=Tref
Ipmax_Tref = 0.5029; %current at maximum power for T=Tref
Rs = (Voc_Tref-Vpmax_Tref)/Ipmax_Tref;
%Iterative calculation of Isc
I = zeros(size(V));
for i=1:10 %number of iteration = 10
    for j=1:length(V);
        I(j) = I(j) - (Iph - I(j) - Id*(exp(q*(V(j)+I(j)*Rs)/(n*k*Tref)) - 1))/...
            (-1 - (Id*(exp(q*(V(j)+I(j)*Rs)/(n*k*Tref)) - 1))*q*Rs/(n*k*Tref));
    end
end

function vv = v_diode(vi)
% V = Voltage on diode terminals [V]
% V is computed by linear interpolation, on the basis of real measurements
% of the I-V curve of the diode.
% I = Current in the diode [A]
% mesures (except the first)
d_v = [0 0.141 0.213 0.227 0.24 0.25 0.26 0.27 0.28 0.3 0.32 0.34 0.36
0.38 0.39 0.4 0.41 0.42 0.44 0.46 0.48 0.5 0.52 0.54 0.56 0.58] ;
d_i = [0 0.00001 0.00002 0.00003 0.0005 0.00007 0.0001 0.00017 0.00023
0.00047 0.00095 0.00206 0.0036 0.0095 0.015 0.021 0.031 0.04 0.072 0.122
0.184 0.27 0.358 0.46 0.605 0.9 ];
```

```

vv = zeros(size(vi));
for k = 1:(length(vi)-1)
    i = vi(k);
    dii = (d_i - i);
    co = 0;
    for j = 1:(length(dii)-1)
        if ((dii(j) <= 0) && (dii(j+1) > 0))
            co = j;
        end
    end
end

% linear interpolation
if (co == 0)
    else
        a = (i - d_i(co+1)) / (d_i(co) - d_i(co+1));
        v = d_v(co)*a + d_v(co+1)*(1-a);
    end
vv(k) = v;
end

cellsim.m % Script de simulation

vmax=3.5;
v = [0:0.05:vmax];

ia1 = Cell_GaAs(v, 1, 5);
da1 = v_diode(ia1);
va1 = 2*v-da1;

figure('Color',[1,1,1]);
plot(va1,ia1,'g')
axis([0 2*vmax 0 0.6])
grid ON
xlabel('Tension [V]')
ylabel('Courant [A]')

ia2 = Cell_GaAs(v, 0.75, 5);
da1 = v_diode(ia2);
va1 = 2*v-da1;
hold on
plot(va1,ia2,'b')

ia3 = Cell_GaAs(v, 0.5, 5);
da1 = v_diode(ia3);
va1 = 2*v-da1;
hold on
plot(va1,ia3,'r')

gtext('G = 1','Color','green','FontSize',10)
gtext('G = 0.75','Color','blue','FontSize',10)
gtext('G = 0.5','Color','red','FontSize',10)

```

REFERENCES

Bibliographic

- [1] Woellert, K., Ehrenfreund, P., Ricco, A., and Hertzfeld, H., "Cubesats: Cost-Effective Science and Technology Platforms for Emerging and Developing Nations," *Advances in Space Research*, 2010.
- [2] Wood, D., Weigel, A. "The Evolution of Satellite Programs in Developing Countries," IAC-09-B4.1.3. Daejeon, Korea, 2009.
- [3] NASA/TP-2015-216648/REV1, "Small Spacecraft Technology State of the Art," Mission Design Division Ames Research Center, Moffett Field, California, December 2015
- [4] Slane, F., D. R. and Cook, D., "Use of Model-Driven Architectures to Increase Responsiveness and Reduce Cost," Reinventing Space Conference, Los Angeles, CA, March 2-5, 2011.
- [5] M. Gruntman, *Blazing the Trail. The Early History of Spacecraft and Rocketry*, Fig. 15.30, p. 375, AIAA, Reston, 2004.
- [6] Siegfried W. Janson, "Small Satellites: Past, Present, and Future," p.47, Editeurs: Henry Helvajain et Siegfried W. Janson, The Aerospace Press, El Segundo, California, 2008.
- [7] Gallton, Daniel A., "The challenge of small satellite systems to the space security environment," NPS Outstanding Thesis Collection, 2012.
- [8] W. J. Larson et J. R. Wertz, "Space Mission Analysis and Design," Space Technology Library, Microcosm Press El Segundo, California, 2005.
- [9] Puig-suari, J., Turner C. and R.J. Twiggs, "Cubesat: The Development and Launch Support Infrastructure for Eighteen Different Satellite Customers on One Launch." 15th Annual/USU Conference on Small Satellites, 2001.
- [10] CubeSat Design Specification (CDS) Rev.13 CubeSat Program, Cal Poly
- [11] Heidt, H., Puig-Suari, J., Moore, A., Nakasuka, S., and Twiggs, R., "CubeSat : A new generation of picosatellite for education and industry low-cost space experimentation," *Proceedings of the Utah State University Small Satellite Conference*, Logan, UT, Citeseer, 2001, pp. 1–2.
- [12] Poly Picosatellite Orbital Deployer Mk. III Rev. E User Guide CubeSat Program, Cal Poly
- [13] Buchen, E. and DePasquale D. "2014 Nano/Microsatellite Market Assessment, SpaceWorks Enterprises", Inc. (SEI) 2014
- [14] David T. Gerhardt and Scott E. Palo. "Passive Magnetic Attitude Control for CubeSat Spacecraft". In 24th Annual AIAA/USU Conference on Small Satellites. University of Colorado, Boulder, 2010.
- [15] ECSS-E-M30A, "Space Project Management- Project Phasing and Planning", ESA-ESTEC, 19 April 1996.
- [16] NASA/SP-2014-3705 "NASA Space Flight Program and Project Management Handbook", National Aeronautics and Space Administration Headquarters Office of the Chief Engineer Washington, D.C. 20546, Septembre 2014.
- [17] M. Jones, E. Gomez, A. Mantineo and U.K. Mortensen, "Introducing ECSS Software Engineering Standards within ESA", *ESA Bulletin* 111, Août 2002.
- [18] ECSS-S-ST-00C, "Description, implementation and general requirements", ESA-ESTEC, 31 Juillet 2008.
- [19] ECSS-M-ST-10C, "Project Planning and Implementation", ESA-ESTEC, 06 Mars 2009.
- [20] ECSS-E-ST-10C, "System engineering - General requirements", ESA-ESTEC, 06 Mars 2009.
- [21] ECSS-E-ST-10-06C, "System engineering - Technical requirements specification", ESA-ESTEC, 06 Mars 2009.
- [22] ECSS-E-ST-20C, "System engineering - Electrical and electronic", ESA-ESTEC, 31 Juillet 2008.
- [23] Vallado, D. A., & McClain, W. D. "Fundamentals of astrodynamics and applications", New York: McGraw-Hill Companies, Inc, 1997.
- [24] Howard D. Curtis. "Orbital Mechanics for Engineering Students", Elsevier Butterworth-Heinemann, 2005.

-
- [25] Kyle T. Alfriend et. al, "Spacecraft Formation Flying", 1st Edition, Butterworth-Heinemann Publications, 2010
- [26] Capderou, M. "Handbook of satellite orbits: From kepler to GPS", NASA Jet Propulsion. Laboratory, California Institute of Technology, California, USA, Springer, 2014
- [27] Chobotov, V.A. "Orbital mechanics" Third Edition. Air Force Institute of Technology, Ohio, USA, American Institute of Aeronautics and Astronautics, Inc, 2002.
- [28] ECSS-E-ST-10-04C, "Space engineering: Space environment", ESA-ESTEC, 15 Novembre 2008.
- [29] Franz T. Geyling and H. Robert Westerman "Introduction to orbital mechanics" Bell Telephone Laboratories, Whippany, New Jersey, Addison-wesley, 1971.
- [30] Kedare, S. S., Ulrich, S. "Design and evaluation of a semi-empirical piece-wise exponential atmospheric density model for CubeSat applications". AIAA SciTech. 2015
- [31] Tewari, A. 2007. "Atmospheric and space flight dynamics: Modeling and simulation with MATLAB and Simulink" Birkhäuser Boston, USA
- [32] Montenbruck, O & Gill, E. 2001. "Satellite orbits: Models, methods, and applications". Verlag Berlin Heidelberg, New York. : Springer.
- [33] Oltrogge D.L., Leveque K., "An Evaluation of CubeSat Orbital Decay", 25th Annual AIAA/USU Conference on Small Satellites, 2011
- [34] Halis C. Polat, Josep Virgili-Llop and Marcello Romano, "Survey, Statistical Analysis and Classification of Launched CubeSat Missions with Emphasis on the Attitude Control Method", JoSS, Vol. 5, No. 3, 2016.
- [35] NASA Procedural Requirements NPR 8715.6A "NASA Procedural Requirements for Limiting Orbital Debris" 2009
- [36] Tapparel Pierre-André, "Control and Data-Management System SwissCube", HEVs Sion Switzerland. 24/02/2007
- [37] Keys A.S. et al, "Radiation Hardened Electronics for Space Environments (RHESE)", Military and Aerospace FPGA and Applications (MAFA), November 2007
- [38] F.B. Mc Lean, "A Framework of Understanding Radiation Induced Interface in SiO₂ MOS Structures", IEEE Transaction on Nuclear Science, vol 31, pp. 1651-1657, 1984.
- [39] D. Bessot, "Radiation hardening technologies facing total dose, S.E.U. and S.E.L. in spatial environment", Apr 1991, 148 p, IMAG, Grenoble (France)
- [40] Sammy Kayali, "Space Radiation Effects on Microelectronics" NASA Radiation Effects Group
- [41] "VEGA user's manual", Issue 4, Rev. 0, April 2014
- [42] "DNEPR user's guide", Issue 2, Novembre, 2001
- [43] "PSLV user's manual", Decembre 2001
- [44] "SOYUZ user's manual", Issue 2, Rev. 0, Mars 2012
- [45] "ARIANE 5 user's manual", Issue 5, Rev. 2, Octobre 2016
- [46] "ROCKOT user's guide", Issue 5, Rev. 0, Août 2011
- [47] "H-IIA user's manual", Ver 4.0, Fevrier 2015
- [48] ECSS-Q-ST-70-02C "Space product assurance - Thermal vacuum outgassing test for the screening of space materials", ESA-ESTEC, 15 November 2008
- [49] National Aeronautics and Space Administration NASA Headquarters, "Systems Engineering Handbook", NASA/SP-2007-6105 Rev1, 2007.
- [50] P. Kuligowski, D. Roszkowski, "PW-SAT2 Preliminary Requirements Review: On-Board Computer", Students' Space Association, The Faculty Of Power And Aeronautical Engineering, Warsaw University Of Technology, 2014.
- [51] A. Denis, J. Pisane, "OUFTI-1 Phase A: Mission definition Space and ground systems description", Université de Liège, September 2009
-

-
- [52] R. Krpoun, M. Noca, N. Scheidegger, "Phase B: Project, Mission, Space and Ground System Overview", SwissCube Mission, UNINE / HES-SO / EPFL Lausanne, Switzerland, 2007
- [53] G. Aalbers, G. N. Gaydadijev, R. Amini, "CDHS Design for a university nano-satellite", 57th International Astronautical Congress (IAF), American Institute of Aeronautics and Astronautics, Valencia, Spain, 2006.
- [54] "I2C-bus specification and user manual", Rev. 6, NXP Semiconductors, 4 April 2014
- [55] ISO 11898-1:2015 "Road vehicles - Controller area network (CAN) - Part 1:Data link layer and physical signaling", Second edition, 2015-12-15
- [56] J. M. Irazabal and S. Blozis, "I2C Manual", Application Note, AN10216-01, Philips Semiconductors March 24, 2003
- [57] "Phase A study of a pico satellite project, Compass-1", University FH Aachen, Allemagne, 2003
- [58] "PC/104 Specification", Ver. 2.6, PC/104 embedded Consortium, October 2008
- [59] "IEEE Standard Letter Designations for Radar-Frequency Bands". IEEE Std. 521-2002. (Approved 12 September 2002, Reaffirmed 17 June 2009)
- [60] Klofas, B. and K. Leveque, "A Survey of CubeSat Communication Systems: 2009-2012", 10th Annual Cubesat Developers Workshop. April 2013 Cal Poly.
- [61] Traussnig, W. "Design of a Communication and Navigation Subsystem for a CubeSat Mission", Post-graduate University Course Space Science, Master of Sciences, Karl Franzens University of Graz, Austria, 2007.
- [62] J. A. King, "Technical, Security, and Regulatory Considerations when Planning the Use of Simplex Telecommand Systems in Small Satellites", Radio Amateur Satellite Corp. (AMSAT), Juin 2004
- [63] "ITU Radio Regulations Articles", Edition 2012
- [64] NASA PRACTICE NO. PD-ED-1258, "Space Radiation Effects on Electronic Components in Low-Earth Orbit", JOHNSON SPACE CENTER, April 1996.
- [65] Solar Reference Cells, Abet Technologies Model 15150: <http://abet-technologies.com/wp-content/uploads/Reference-cells-2015.pdf>
- [66] R. Leyva, C. Alonso, I. Queinnec, A. Cid-Pastor, D. Lagrange and L. Martinez-Salamero, "MPPT of photovoltaic systems using extremum - seeking control," in IEEE Transactions on Aerospace and Electronic Systems, vol. 42, no. 1, pp. 249-258, Jan. 2006.
- [67] T. Eswam and P. L. Chapman, "Comparison of photovoltaic array maximum power point tracking techniques", IEEE Trans. Energy Convers., vol. 22, no. 2, pp. 439-449, Jun. 2007
- [68] Subudhi B ; Pradhan R, "A Comparative Study on Maximum Power Point Tracking Techniques for Photovoltaic Power Systems", IEEETrans on sustainable energy., vol. 4, No 1, January 2013.
- [69] Xilinx Inc, "Command Line Tools User Guide", UG628 Xilinx Guide 2012.
- [70] C. Clark and A. Lopez, "Power system challenges for small satellite missions," in Proceedings of the 2006 Small Satellites, Systems and Services Symposium, D. Danesy, Ed. The Netherlands, ESA, 2006.
- [71] DES : Document des exigences système MASAT1
- [72] N. Fatemi, H. Pollard, H. Hou, and P. Sharps, "Solar array trades between very high-efficiency multi-junction and Si space solar cells," in Conference Record of the 28th IEEE Photovoltaic Specialists Conference (PVSC 2000). New York, NY: IEEE, 2000, pp. 1083-1086.
- [73] M. R. Reddy, "Space solar cells - tradeof analysis," Solar Energy Materials and Solar Cells, vol. 77, no. 2, pp. 175-208, May 2003.
- [74] LUQUE A., HEGEDUS S., "Handbook of photovoltaic science and engineering", Wiley, second edition 2011
- [75] Pierre THIRION, "Design and Implementation of On-board Electrical Power Supply of Student Nanosatellite OUFIT-1 of University of Liège", Master in Electrical Engineering, Applied Science Faculty of the University of Liège, 2009

-
- [76] Atakan Sirin, "Power System Analysis of J3 CubeSat and RATEX-J High Voltage Power Supply Calibration", Luleå University of Technology Department of Computer Science, Electrical and Space Engineering, 2015
- [77] ClydeSpace "User Manual: CubeSat 1U Electronic Power System and Batteries CS-1UEPS2", 2010
- [78] GOMSSpace "NanoPower P31U / P31US: Datasheet", November 2016
- [79] Crystalspace "Crystalspace P1U Vasik Power Supply", January 2017
- [80] Xilinx Inc, "OS and Libraries Document Collection", UG643 (v2014.1) April 1, 2014.
- [81] K. Z. Brown, T. Clements and J. Evans "CUBESAT Orientation Control and Matching to Communication System Requirements", Advances in the Astronautical Sciences Second IAA DyCoss'2014, Volume 153
- [82] NanoPower Battery Datasheet: <https://gomspace.com/UserFiles/Subsystems/datasheet/gs-ds-nanopower-batteries-10.pdf>. January 2017
- [83] Klofas B., "Cubesat communication system 2003-2016", Version 14, February 28, 2017.
- [84] I. Latachi, M. Karim, A. Hanafi, T. Rachidi, et al. " Link Budget Analysis for a LEO CubeSat Communication Subsystem", International Conference on Advanced Technologies for Signal & Image Processing, ATSIP'2017, Mai 2017.
- [85] Jinjun Shan, "Communications Satellite Payload - Link Budget", Department of Earth and Space Science and Engineering, YORK University,
- [86] Alexandre Boyer, "ANTENNES", Institut National des Sciences Appliquées de Toulouse, 2011
- [87] Louis J. Ippolito, Jr. "Satellite Communications Systems Engineering - Atmospheric Effects, Satellite Link Design and System Performance" ITT Advanced Engineering & Sciences, USA, and The George Washington University, Washington, DC, USA, JohnWiley & Sons Ltd, 2008
- [88] Brenda Lidia Escobar Mendez, "Link Budget for NTNU Test Satellite", Master thesis of Science in Communication Technology, Norwegian University of Science and Technology, July 2013.
- [89] GOMSpace "NanoCOM AX100 Datasheet: Long-range software configurable UHF transceiver", August 2016
- [90] Pumpkin P/N 710-00484 "CubeSat Kit Motherboard (MB): Single Board Computer Motherboard for Harsh Environments", 2009 (<http://www.cubesatkit.com/docs/datasheet/>)
- [91] GOMSpace "NanoMind A712D Datasheet: On-board Computer System for mission critical space application with limited resources", March 2016
- [92] GOMSpace "NanoMind A3200 Datasheet: On-board Computer System for mission critical space applications", December 2016
- [93] GOMSpace "NanoDock Motherboard DMC-3 Datasheet: Daughter module carrier for 4 daughterboards", December 2016
- [94] Duncan Miller, "MCubed-2 Magnetic Research", MXL Tech Memo, Michigan Exploration Laboratory, May 2, 2013
- [95] Olivier Maloberti. "Contribution à la modélisation de la dynamique d'aimantation dans les matériaux magnétiques doux : Caractérisation et simulation.", Energie électrique, Institut National Polytechnique de Grenoble - INPG, 2006
- [96] Samir Ahmed Rawashdeh, "Passive attitude stabilization for small satellites", University of Kentucky, 2010
- [97] "CubeSat Space Protocol (CSP): Network-Layer delivery protocol for CubeSats and embedded systems". <http://gomspace.com/documents/ds/gs-csp-1.1.pdf>
- [98] W. A. Beech, D. E. Nielsen, and J. Taylor, "AX. 25 Link Access Protocol for Amateur Packet Radio," Tucson Amateur Packet Radio Corporation, Tucson, no. July, 1998.
- [99] ECSS-E-70-41A, " Space engineering: Ground systems and operations - Telemetry and telecommand packet utilization", ESA-ESTEC, 30 January 2003

-
- [100] Sebastien De Dijcker, "Implémentation de la gestion des télécommandes et des télémetries au sein de l'ordinateur de bord du nanosatellite OUFTI-1", Master Informatique Industrielle, Haute Ecole de la Province de Liège, Catégorie Technique, 2010–2011
- [101] ATMEL Application note AT04056, " Getting Started with FreeRTOS on Atmel SAM Flash MCUs" 2014 Atmel Corporation.
- [102] S. Habinc, "Suitability of reprogrammable FPGAs in space applications" (compilation), FPGA-002-01, Report ESA contract No. 15102/01/NL/FM(SC) CCN-3, September 2002.
- [103] ECSS-E-ST-70-11C, Space segment operability
- [104] MacMahon, S., Zang, N. and Sarangi, A. LightWeight IP (lwIP) Application Examples, Xilinx Application note XAPP1026 (v5.1). 2014.
- [105] Arnaud Tisserand, "Introduction aux circuits FPGA", INRIA LIP Arénaire, OCAM (ACI-SI), Lyon, 23-24 septembre 2003.
- [106] S. Brown, J. Rose, "Architecture of FPGA and CPLDs: A tutorial", IEEE Design and Test of Computers, volume 13, pages 42-57, 1996.
- [107] Fabula, J. J., DeJong, J. L., Lesea, A., and Hsieh, W.-L. "The Total Ionizing Dose Performance of Deep Submicron CMOS Processes", Military and Aerospace Programmable Logic Devices Conference, Maryland, USA, Sept. 2008.
- [108] Ramin Roosta, "A Comparison of Radiation-Hard and Radiation-Tolerant FPGAs for Space Applications", NASA Electronic Parts and Packaging Program, December 30, 2004.
- [109] Wazard, N., Salvaterra, G. and PIKE, P. "SPARTAN 6 Evaluation for Space Application", Airbus Defence & Space. 2016.
- [110] Sandor, M., Davarpanah, M., Soliman, K., Suszko, S., and Mackey, S., "Field Programmable Gate Arrays: Evaluation Report for Space-Flight Application," Tech. Rep., NASA JPL Report, NASA CR-192796, Sep. 1992.
- [111] D. Ratter, "FPGAs on Mars," Xcell Journal, vol. 50, pp. 8-11, 2004.
- [112] Ebert, Dean A. "Design And Development Of A Configurable Fault-Tolerant Processor (CFTP) For Space Applications", Naval Postgraduate School Monterey, California, June 2003
- [113] Felix Huber, Peter Behr, Hans-Peter Röser, Samuel Pletner, " FPGA based On-Board Computer System for the "Flying Laptop" Micro-Satellite", Proceedings of DASIA 2007 - Data Systems In Aerospace: 29 May - 1 June 2007, Naples.
- [114] Thong Thai, "Applications for FPGAs on Nanosatellites", Graduate Program in Earth and Space Science York University Toronto, Ontario April 2014.
- [115] Gilles Foucard, "Taux d'erreurs dues aux radiations pour des applications implémentées dans des FPGAs à base de mémoire SRAM : prédiction versus mesures", Thèse doctorat, Laboratoire TIMA, Institut polytechnique de Grenoble Spécialité : Micro et Nano Electronique, juin 2010.
- [116] Shirvani, P., "The Effects of Radiation on Electronic Systems" October 2000 Lecture for Stanford University Center for Reliable Computing.
- [117] T.Hardcastle "Spartan-6 FPGA Dual-Lockstep MicroBlaze Processor with Isolation Design Flow", Xilinx Application Note Author, XAPP584 (v1.0) 2012.
- [118] J. Johnson, W. Howes, M. Wirthlin, D. L McMurtrey, M. Caffrey, P. Graham, and K. Morgan. "Using duplication with compare for on-line error detection in FPGA-based designs". In Proceedings of IEEE Aerospace Conference, 2008, page 1–11, 2008.
- [119] Carl Carmichael, "Triple Module Redundancy Design Techniques for Virtex FPGAs Author", Application note, Virtex series, XAPP197 (v1.0.1) July 6, 2006.
- [120] Francisco Fons Lluís, "Embedded Electronic Systems Driven By Run-Time Reconfigurable Hardware", Doctoral Thesis, Universitat Rovira I Virgili, 2012.
- [121] Xilinx, Inc. "MicroBlaze Processor Reference Guide - Embedded Development Kit EDK 14.7", UG081 (v14.4), 2013.

-
- [122] Nathan Arthur Harward, "Measuring Soft Error Sensitivity of FPGA Soft Processor Designs Using Fault Injection", Department of Electrical and Computer Engineering, Brigham Young University, March 2016.
- [123] Xilinx, Inc. "Spartan-6 FPGA Configuration User Guide", UG380, January 23, 2013.
- [124] Xilinx, Inc, "LogiCORE IP AXI HWICAP", DS817 Xilinx IP Documentation, July 25, 2012.
- [125] Xilinx, Inc, "AXI Reference Guide", UG761 Xilinx Guide, November 15, 2012.
- [126] Véronique Ferlet-Cavrois, "Overview of Radiation Test Activities on Memories at ESA", ESA ESTEC, TEC-QEC, 30/04/2015.
- [127] "NAND vs. NOR Flash Memory Technology Overview", Toshiba America, Electronic Component Inc., 2006.
- [128] Tetsuo Miyahira and Gary Swift, "Evaluation of Radiation Effects in Flash Memories", Jet Propulsion Laboratory, California Institute of Technology Pasadena, California.
- [129] Xilinx, Inc, "Spartan-6 FPGA Memory Controller User Guide" UG388 (v2.3) August 9, 2010
- [130] Micron Datasheet, " N25Q064A, Micron Serial NOR Flash Memory, 3V, Multiple I/O, 4KB Sector Erase", 2011 Micron Technology, Inc.
- [131] Micron Datasheet, "DDR3 SDRAM, MT41J64M16 – 8 Meg x 16 x 8 banks", 2006 Micron Technology, Inc.
- [132] ST Datasheet, "M24C08-W M24C08-R M24C08-F 8-Kbit serial I²C bus EEPROM", May 2017 DocID023924 Rev 5.
- [133] Linear Technologie, "Dual Channel 3A, 15V Monolithic Synchronous Step-Down Regulator", Linear Technology Corporation 2010.
- [134] Texas Instrument, "TPS51200 Sink and Source DDR Termination Regulator", SLUS812C, February 2008, Revised November 2016.

Webographie

- [135] NASA Launch Services Program, Site du projet ELaNa: https://www.nasa.gov/directorates/heo/home/CubeSats_initiative, 2016.
- [136] Programme ESA "Fly your satellite!", site de l'ESA : http://www.esa.int/Education/CubeSats_Fly_Your_Satellite/
- [137] Programme ESA SSETI, site de l'ESA : http://www.esa.int/Education/SSETI_Express/
- [138] Bureau des affaires spatiales OOSA de l'ONU : <http://www.unoosa.org/oosa/index.html>
- [139] Programme "KiboCUBE" de coopération ONU-Japon sur le déploiement de CubeSat à partir de la Station spatiale internationale (ISS) : <http://www.unoosa.org/oosa/fr/ourwork/psa/hsti/kibocube.html>
- [140] Erik Kulu, Nanosatellite Database : <http://www.nanosats.eu/index.html#database>. Janvier 2017.
- [141] European Cooperation for Space Standardization, site : <http://www.ecss.nl/>.
- [142] <https://sohowww.nascom.nasa.gov/gallery/SolarCorona/>
- [143] <https://sdo.gsfc.nasa.gov/gallery/main>
- [144] Logiciel SPENVIS en ligne : <https://www.spervis.oma.be/intro.php>
- [145] Modèle CAO 3D structure Pumpkin 1U : <http://www.cubesatkit.com/content/design.html>
- [146] <http://www.azurspace.com/index.php/en/products/products-space>
- [147] <http://pveducation.org/pvcdrom/2-properties-sunlight/atmospheric-effects>
- [148] ITU-RP 676, RP 840, RP 618. <https://www.itu.int/rec/R-REC-P>.