



# Modélisation et simulation de microsystemes multi domaines à signaux mixtes : vers le prototypage virtuel d'un microsysteme autonome

A. Zenati

## ► To cite this version:

A. Zenati. Modélisation et simulation de microsystemes multi domaines à signaux mixtes : vers le prototypage virtuel d'un microsysteme autonome. Micro et nanotechnologies/Microélectronique. Université Joseph-Fourier - Grenoble I, 2007. Français. NNT : . tel-00202164

**HAL Id: tel-00202164**

**<https://theses.hal.science/tel-00202164>**

Submitted on 4 Jan 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Joseph Fourier Grenoble I

N° attribué par la bibliothèque

\_\_\_\_\_

**THESE**

pour obtenir le grade de

**DOCTEUR DE L'UNIVERSITE JOSEPH FOURIER**

**Spécialité : MICRO ET NANO ELECTRONIQUE**

Préparée au laboratoire **TIMA**

dans le cadre de l'Ecole Doctorale

Electronique, Electrotechnique, Automatique, Télécommunications, Traitement du Signal

Présentée et soutenue publiquement

par

**Amel ZENATI**

Le 26/10/2007

Titre :

# Modélisation et simulation de microsystèmes multi domaines à signaux mixtes : vers le prototypage virtuel d'un microsystème autonome

Directeur de thèse  
**Skandar BASROUR**

Jury

Pr. Jérôme DELAMARE  
Pr. Tarik BOUROUNA  
Pr. Ian O'CONNOR  
Pr. Skandar BASROUR

Président  
Rapporteur  
Rapporteur  
Directeur de thèse



*A ceux qui m'ont tout donné*  
*A mes parents*



# Remerciements

J'adresse mes sincères remerciements à mon directeur de thèse Monsieur Skandar BASROUR pour avoir proposé et encadré cette thèse. Je lui exprime ma gratitude et ma reconnaissance pour sa disponibilité, ses conseils judicieux son soutien et surtout sa patience. Je remercie également Monsieur Jérôme Delamare pour avoir présidé le jury de ma thèse, ainsi que Messieurs Tarik BOUROUINA et Ian O'CCONOR pour avoir accepté de rapporter mes travaux.

Je manifeste ma reconnaissance aussi à Monsieur Bernard COURTOIS l'ancien directeur du laboratoire TIMA de m'avoir accueillie et permis de travailler sur ce sujet, sans oublier Madame Dominique Borionne, actuellement directrice du laboratoire.

Je remercie les collègues du groupe MNS pour leur collaboration et leur aide précieuse : Agnès, Hella, Yasser, Marcin et Karine.

Un grand merci au staff TIMA/CMP : Anne-Laure, Chantal, Corinne, Françoise, Isabelle, Lucie, Marie-Christine, Hubert et Khouldoun pour leur accueil, leur soutien et leur gentillesse.

Mes remerciements à Alexandre et Robin du service CIME pour leurs aides et interventions qui ont favorisées le bon déroulement des travaux de recherche ainsi que les membres du service informatique ; Ahmed, Frédéric et Nicolas.

Mes remerciements fraternels vont à Abdelaziz, Aimen, Faiza, Lobna, Wafa, et Wassim ; les piliers qui m'ont soutenue dans les moments difficiles.

Je tiens à remercier Anna, Alexandre, Laurent, Karim, Amine, Lilia et tous les amis du laboratoire pour les moments agréables que nous avons partagés.

A tous ceux qui ont contribué de près ou de loin à la réalisation de cette thèse durant toutes ces années, à : Hassane ALLA, Leila, Hichem, Nejma, Nadia, Myriam, Linda, et Carole.

Pour finir, je salue toute ma famille ; mes parents, mes frères et sœurs de m'avoir soutenue et d'avoir cru en moi.



# Table des matières

1. Contexte .....	2
2. Objectifs .....	2
2.1 Modélisation et simulation multi domaines à signaux mixtes .....	3
2.2 Application aux microsystemes autonomes .....	5
3. Contributions .....	8
4. Plan du manuscrit .....	9
Chapitre II- Méthodes de conception et outils de modélisation des microsystemes.....	11
A. Conception des microsystemes .....	12
1. Définition d'un microsysteme sur puce .....	12
2. Intégration des microsystemes sur puces .....	13
3. Description des microsystemes à différents niveaux d'abstraction .....	15
3.1 Modélisation au niveau système .....	16
3.2 Modélisation comportementale .....	16
3.2.1 Modélisation fonctionnelle.....	16
3.2.2 Modélisation comportementale basée sur le circuit électrique équivalent.....	18
3.3 Modélisation au niveau composant .....	20
3.4 Modélisation au niveau physique .....	20
3.5 Dispositif physique versus modèle des composants comportementaux.....	21
4. Flot de conception des microsystemes .....	21
5. Simulation des microsystemes .....	23
6. Prototypage virtuel .....	23
7. Méthodologies de conception.....	24
7.1 Cycle de conception en cascade .....	24
7.2 Cycle de conception en V.....	24
7.3 Méthodologie proposée : le cycle de conception en spirale.....	25
8. Bilan .....	27
B. Langages et environnement de modélisation.....	28
1. Définition des outils de modélisation.....	28
2. Modélisation collaborative .....	29
3. Langages et environnements de modélisation.....	30
3.1 Langage C++ .....	30
3.2 SystemC .....	30
3.3 VHDL-AMS et/ou Verilog-AMS.....	31
3.4 Modelica.....	32
4. Comparaison entre VHDL-AMS et Modelica .....	32
4.1 Simulateurs électriques .....	33
5. L'environnement MATLAB/SIMULINK .....	34
5.1 Processus de simulation sous Simulink.....	34
5.2 Outils de cosimulation avec MATLAB/Simulink.....	35
6. Bilan .....	38
7. Conclusion.....	39
Chapitre III- Modélisation du microsysteme autonome.....	43
A. Modélisation d'un microsysteme autonome .....	44
1. Présentation du microsysteme autonome .....	44
2. Générateurs piézoélectriques.....	46
2.1 Modèle générique de conversion des vibrations mécaniques en électricité.....	48



2.2	Modélisation du générateur de puissance.....	49
2.2.1	Modèle en circuit électrique équivalent .....	50
	• Implémentation du modèle en circuit électrique équivalent .....	51
	• Fonction de transfert du générateur piézoélectrique .....	52
2.2.2	Modèles fonctionnels basés sur une approche analytique.....	53
	• Modèle en diagramme blocs .....	59
	• Modèle équivalent électrique raffiné.....	63
2.2.3	Modèle analytique unidimensionnel d'un générateur MEMS piézoélectrique	65
2.2.4	Modèle physique (EF).....	68
	• Comparaison du modèle FEM avec le modèle analytique .....	68
2.3	Construction d'une bibliothèque de générateurs piézoélectriques .....	71
2.4	Bilan .....	72
3.	Modélisation du circuit de récolte d'énergie.....	74
3.1	Redresseur AC/DC .....	75
3.1.1	Implémentation de l'AC/DC.....	77
3.2	Convertisseur DC/DC .....	80
3.2.1	Principe de fonctionnement du convertisseur convertisseur DC/DC en mode continu .....	81
	• Ondulations du courant $i_L$ et de la tension $v_c$ .....	82
	• Choix de L et de C.....	83
3.2.2	Fonctionnement en mode discontinu.....	83
	• Choix de L et de C.....	84
	• Exemple de calcul du rapport cyclique en mode discontinu .....	84
3.2.3	Modèle du CONVERTISSEUR DC/DC.....	86
	• Evaluation du modèle.....	86
	• Introduction des effets "snubber" .....	89
3.2.4	Modèles raffinés des composants non linéaires .....	90
3.2.5	Evaluation du modèle raffiné .....	91
3.3	Génération de la commande du commutateur.....	93
3.4	Implémentation du bloc numérique.....	95
	• En diagramme blocs .....	96
	• En code VHDL.....	97
3.5	Bilan .....	98
4.	Elément de stockage : la batterie.....	100
4.1	Modèle comportemental dynamique.....	100
4.1.1	Circuit équivalent électrique .....	101
4.1.2	Modèle basé sur les LUT .....	105
4.1.3	Modèle de décharge par interpolation de courbes.....	108
4.1.4	Bilan .....	108
4.2	Modèle raffiné de la batterie .....	109
4.2.1	Implémentation du modèle.....	109
4.3	Bilan .....	115
5.	Conclusion.....	116
B.	Stratégie de la simulation .....	117
1.	Simulation globale.....	117
1.1	Premier prototype virtuel .....	118
1.1.1	Simulation en mode continu.....	119
1.1.2	Discretisation du prototype .....	120
1.2	Exemple du prototype virtuel avec plusieurs niveaux d'abstraction .....	122
2.	Cosimulation .....	124

3. Comparaison des temps de simulation .....	127
4. Validation expérimentale .....	129
5. Conclusion.....	130
Chapitre IV- Conclusion et perspectives.....	132
Références .....	134
Glossaire.....	172



# Table des figures

Figure II-1 : Exemple de microsystème monolithique : accéléromètre ADXL105 .....	3
Figure II-2 : Capteur sans fil MICA2DOT à côté d'une pièce d'un quart U.S (Image source : Crossbow Technology, Inc.). .....	5
Figure II-3 : Structure du réseau WINS : Les nœuds, le service réseau (ou l'utilisateur), le pont pour l'acheminement des données. ....	6
Figure II-4 : Le dispositif Smart dust contient des microcapteurs, un récepteur optique, des transmetteurs passifs et actifs optiques, circuiterie de traitement des signaux et de contrôle, et une source d'énergie. ....	7
Figure II-5 : Architecture du microsystème autonome du projet VIBES. ....	8
Figure II-1 : Diagramme d'un microsystème multi domaines à signaux mixtes .....	13
Figure II-2 : Microsystèmes monolithique (a), microsystème multi chip (b) .....	14
Figure II-3 : Interaction entre les composants dans les domaines hybrides .....	17
Figure II-4 : (a) Circuit électrique, (b) Diagramme bloc, (c) Graphe de liaison .....	17
Figure II-5 : Interaction entre les composants dans le domaine électrique .....	18
Figure II-6 : Variable terminale : flux et effort .....	19
Figure II-7 : Flot de conception des éléments d'un microsystème à différents niveaux d'abstraction.....	22
Figure II-8 : Les phases du cycle en cascade .....	24
Figure II-9 : Le modèle de conception en V .....	25
Figure II-10 : Les phases du cycle de conception en spiral .....	27
Figure II-11 : Architecture de Real Time Workshop .....	37
Figure II-12 : Interface Link for ModelSim/MATLAB/SIMULINK .....	38
Figure III-1 : Schéma du microsystème autonome .....	44
Figure III-2 : Partitionnement du microsystème autonome selon les domaines et la nature des signaux .....	45
Figure III-3 : Principe de la piézoélectricité.....	47
Figure III-4 : Modes d'excitation d'un barreau piézoélectrique .....	47
Figure III-5 : Différentes structures de type bimorphe .....	48
Figure III-6 : Schéma fonctionnel d'un transducteur des vibrations en électricité.....	49
Figure III-7 : Circuit électrique équivalent du transducteur en circuit ouvert .....	50
Figure III-8 : Implémentation du modèle comportemental sous Simulink à l'aide de la boîte à outil SimPowerSystems.....	51
Figure III-9: Tension fournie par le générateur aux bornes de la résistance optimale.....	52
Figure III-10 : Réponse de la fonction de transfert du modèle comportemental .....	53
Figure III-11 : Structure simple à modéliser .....	54
Figure III-12 : Structure du transducteur piézoélectrique .....	55
Figure III-13 : Implémentation du modèle analytique du microgénérateur sous Simulink : (a) modèle en diagramme blocs, (b) fonction de transfert de la tension.....	60
Figure III-14 : Modèle mixte : analytique et électrique (Simulink/SimPowerSystems).....	61
Figure III-15 : Puissance et tension en fonction de la charge $R$ (a) résultats publiées [Rou-03] (b) résultats obtenus .....	62
Figure III-16 : Allure de la tension $v_p(t)$ .....	62
Figure III-17 : Schéma électrique du générateur piézoélectrique implémenté sous Simulink à l'aide de la boîte SimPowerSystems.....	64
Figure III-18: Comparaison entre la tension de sortie du modèle électrique comportemental et raffiné .....	64

Figure III-19 : (a) Structure de la poutre encastrée-libre (b) Mouvement de la poutre décrite	65
Figure III-20 : Modèle unidimensionnel développé par M. Marzencki [Mar-07]	66
Figure III-21 : Représentation du déplacement en fonction de la fréquence et de la résistance de charge	67
Figure III-22 : Evolution de la tension en fonction du temps	67
Figure III-23 : Exemple de modélisation avec les EF	68
Figure III-24 : Exemple d'un fichier de résultats EF implémenté sous Simulink	69
Figure III-25 : Diagramme de Bode de la tension	70
Figure III-26 : Comparaison entre le modèle analytique sous Simulink et le modèle en EF	70
Figure III-27 : Modèle paramétrable du générateur de puissance	71
Figure III-28 : Tableau récapitulatif des avantages et des inconvénients de la modélisations d'un MEMS sous Simulink/SimPowerSystems	73
Figure III-29 : Schéma bloc du circuit de récolte d'énergie	75
Figure III-30 : Schéma électrique du redresseur AC/DC	75
Figure III-31 : (a) Allure du courant du générateur, (b) Allure de la tension	76
Figure III-32 : Allure du courant de l'AC/DC	76
Figure III-33 : Schéma de l'AC/DC sous Simulink avec les composants de SimPowerSystems	77
Figure III-34 : Exemple de la diode 1N4002 sous SimPowerSystems	78
Figure III-35 : Comparaison des allures du courant $I_{rect}$ à la sortie de l'AC/DC sous SimPowerSystems et SPICE	79
Figure III-36 : Allure de la tension du générateur de puissance à l'entrée de l'AC/DC avec SimPowerSystems	79
Figure III-37 : Allure de la tension de sortie de l'AC/DC avec SimPowerSystems	80
Figure III-38 : Schéma du convertisseur abaisseur de tension (Buck)	81
Figure III-39 : Signaux $v_L$ et $i_L$ de l'abaisseur de tension DC/DC	82
Figure III-40 : Forme de l'ondulation du courant dans l'inductance	83
Figure III-41 : La forme d'onde des différents signaux buck_DCM	84
Figure III-42 : Modèle étudié pour calculer le rapport cyclique	85
Figure III-43 : Caractérisation de l'inductance (a) et de la capacité (b) du DC/DC en mode continu	86
Figure III-44 : (a) courbes de $i_L$ , (b) $v_{out}$ sous SPICE	87
Figure III-45 : Schéma du DC/DC sous Simulink	88
Figure III-46 : Courbe de $i_L$ sans l'effet "snubber"	88
Figure III-47 : (a) Paramètres du transistor MOS et de la diode, (b) $i_L$ , (c) $v_{out}$ avec SimPowerSystems	90
Figure III-48 : Schéma électrique équivalent d'une diode	91
Figure III-49 : Schéma électrique équivalent du transistor MOS	91
Figure III-50 : Paramètres du modèle raffiné (a) du MOSFET, (b) de la diode	92
Figure III-51 : Implémentation du modèle raffiné du DC/DC avec MATLAB:Simulink/SimPowerSystems	93
Figure III-52 : Comparaison des courbes de $i_L$ obtenue avec le modèle du DC/DC au niveau circuit	93
Figure III-53 : (a) Génération du signal MLI, (b) Modèle de base, (c) Modèle analytique	95
Figure III-54 : $\alpha=f(I_p)$ & $I_{bat}=f(\alpha)$	96
Figure III-55 : Courbe théorique du courant dans la batterie $I_{bat} = f(\alpha)$	96
Figure III-56 : Diagramme bloc de l'algorithme de contrôle	97
Figure III-57 : Réponse de l'algorithme de contrôle à un signal sinusoïdal	97
Figure III-58 : Exemple d'exécution de l'algorithme à pas fixe en VHDL	98
Figure III-59 : Caractéristique de charge et de décharge en fonction du temps d'une batterie Sony 18650 (4,2 V; 1,5 Ah)	101

Figure III-60 : (a) Modèle de Thevenin (b) Tension en fonction du temps .....	102
Figure III-61 : Circuit électrique équivalent d'une batterie .....	103
Figure III-62 : Caractéristiques de la batterie Sony 18650 (4,2 V; 1,5 Ah).....	104
Figure III-63 : Train d'impulsions appliqué pour la décharge.....	104
Figure III-64 : Implémentation du modèle de la batterie en charge avec un LUT.....	106
Figure III-65 : Tension de charge pour la batterie Sony 18650 (4,2 V; 1,5 Ah).....	106
Figure III-66 : Implémentation du modèle de la batterie en décharge avec un LUT.....	106
Figure III-67 : Caractéristique de $v_{bat}$ en fonction de l'état de décharge .....	107
Figure III-68 : Caractéristiques de la batterie pour différents courants de décharge .....	107
Figure III-69 : Modèle de la micro batterie en utilisant les S-FUNCTION.....	108
Figure III-70 : Sous-systèmes du modèle au niveau circuit de la batterie .....	111
Figure III-71: Implémentation du modèle au niveau circuit de la batterie sous Simulink/SimPowerSystems .....	112
Figure III-72 : Allure du courant appliqué pour la décharge .....	114
Figure III-73 : Comparaison entre le modèle SPICE et Simulink/SimPowerSystems .....	114
Figure III-74 : Tension de charge aux bornes de la batterie.....	115
Figure III-75 : Premier prototype virtuel du microsystème .....	119
Figure III-76 : Résultats de simulation avec/sans l'accélérateur de Simulink.....	120
Figure III-77 : Evolution du rapport cyclique pour différents modes de simulation globale.	121
Figure III-78 : Comparaison des résultats de la simulation globale en temps continu et discret .....	122
Figure III-79 : Evolution du rapport cyclique pour le prototype multi niveaux .....	123
Figure III-80 : Courant dans la batterie pour le prototype multi niveaux d'abstraction .....	124
Figure III-81 : Prototype virtuel avec la cosimulation Simulink/ModelSim .....	125
Figure III-82 : Fenêtre des signaux sous ModelSim .....	126
Figure III-83 : Evolution du rapport cyclique en cosimulation.....	126
Figure III-84 : Comparaison des résultats de cosimulation avec/sans l'accélérateur de Simulink et simulation globale.....	127
Figure III-85 : Expérimentation d'un macro modèle à base d'une structure bimorphe .....	129
Figure III-86 : Comparaison entre le résultat expérimental, simulation globale et cosimulation .....	130
Figure IV-1 : Déploiement d'un réseau sans fil.....	144
Figure IV-2 : Modèle électrique du générateur de puissance sous SPICE.....	146
Figure IV-1 : La représentation du système complet. ....	149
Figure I-2 : Le circuit équivalent du système complet.....	150
Figure I-3 : Les courants de sortie et dans la capacité. ....	150
Figure I-4 : Les résultats de dimensionnement de la capacité $C_{rect}$ . ....	152
Figure I-5 : Les courbes du courant dans l'inductance et la tension sur la capacité du convertisseur.....	153
Figure I-6 : Modélisations sous Simulink de la diode.....	156



# Tableaux

---

Tableau II-1 : Les niveaux d'abstraction pour les domaines mécanique, électronique analogique et électronique numérique.....	15
Tableau II-2 : Exemples de simulateurs pour VHDL-AMS .....	32
Tableau III-1 : Propriétés du transducteur piézoélectrique à base de matériau PZT-5H .....	61
Tableau III-2 : Définition des facteurs des équations différentielles .....	66
Tableau III-3 : Composants utilisés pour modéliser le AC/DC d'après [Ott-02] .....	77
Tableau III-4 : Paramètres de simulation de l'AC/DC .....	78
Tableau III-5 : Composants linéaires utilisés .....	87
Tableau III-6 : Valeurs utilisées pour le calcul du rapport cyclique .....	95
Tableau III-7 : Récapitulatif de la modélisation du circuit de récolte d'énergie .....	99
Tableau III-8 : Paramètres utilisés pour le modèle de Thevenin.....	101
Tableau III-9 : Paramètres de la batterie .....	103
Tableau III-10 : Paramètres du modèle au niveau circuit de la batterie [Gol-97].....	114
Tableau III-11 : Synthèse de la modélisation d'une batterie .....	115
Tableau III-12 : Différentes simulations multi niveaux .....	118
Tableau III-13: Paramètres utilisés dans le premier prototype virtuel .....	119
Tableau III-14 : Comparaison des temps de simulation entre les différents modes de simulation.....	128
Tableau III-15 : Paramètres de la structure piézoélectrique du générateur.....	129
Tableau IV-1 : VHDL-AMS vs. Verilog-AMS .....	142
Tableau IV-2 : Analogies entre les différents domaines .....	143
Tableau IV-3 : Analogies des équations entre le domaine électrique et mécanique.....	143





# Introduction générale

---

1. Contexte .....	2
2. Objectifs .....	2
2.1 Modélisation et simulation multi domaines à signaux mixtes .....	3
2.2 Application aux microsystemes autonomes .....	5
3. Contributions .....	8
4. Plan du manuscrit .....	9

Ce chapitre est destiné à définir le contexte des nos travaux de thèse qui concerne la conception des microsystemes. Ensuite, nous donnons les motivations de la modélisation et de la simulation à signaux mixtes et multi domaines<sup>1</sup> (MSMD) appliquées à une nouvelle catégorie de microsystemes destinés à des applications sans fil. Finalement, nous énumérons les contributions de nos travaux de thèse, afin de répondre aux enjeux du temps de mise sur le marché.

## 1. Contexte

L'introduction de nouvelles technologies d'intégration de dispositifs électriques et non électriques sur la même puce ainsi que la miniaturisation des systèmes, entraîne une croissance des besoins pour la communication entre différents environnements : mécanique, électrique, électrochimique, thermique, fluide ou optique. Ces dernières années, on observe un grand intérêt vis-à-vis d'une nouvelle génération de microsystemes dédiés à des applications portables et intégrant les communications sans fils. Ces dispositifs seront utilisés dans plusieurs domaines sensibles tel que le médical, l'automobile, les constructions antisismiques...etc.

Dans certaines applications spécifiques, les microsystemes requièrent d'être complètement intégrés dans des structures indépendantes de l'environnement extérieur. L'alimentation ordinaire et la communication avec le monde extérieur pour ce type d'applications deviennent alors difficiles. La solution attrayante pour le problème de l'alimentation consiste, en la récupération de l'énergie sous différentes formes (vibrations, lumière, chaleur...), et sa conversion en une énergie électrique. Cette énergie sera alors emmagasinée dans des éléments de stockage tel que des batteries. D'un autre côté, les protocoles de communication sans fil seront utilisés dans ce genre d'applications. Cette nouvelle génération de microsystemes sans fil est appelée autonome (SPMS)<sup>2</sup> et fait l'objet de plusieurs travaux de recherche. Certains de ces travaux ont été commercialisés [Par-05]. L'idée de base est d'intégrer dans un même système compact la technologie des microcapteurs, des modules de calcul et d'adaptation des signaux, en plus d'une interface réseau à très faible consommation (ULP)<sup>3</sup> et au moindre coût.

## 2. Objectifs

La conception des microsystemes requiert des compétences multidisciplinaires nécessitant la collaboration de plusieurs groupes de différentes spécialités. Ces groupes ont leurs propres méthodologies de travail et leurs langages de modélisations qui sont spécifiques à un domaine particulier. L'objectif primordial de ma thèse est de développer de nouvelles méthodes afin d'exploiter ces travaux sans pour autant changer les méthodes de travail des groupes spécialisés. Ainsi de

---

<sup>1</sup> **MSMD**- de l'anglais **M**ixed **S**ignal and **M**ulti **D**omain

<sup>2</sup> **SPMS**- de l'anglais **S**elf **P**owered **M**icro**S**ystems

<sup>3</sup> **ULP**- de l'anglais **U**ltra **L**ow **P**ower

nouvelles méthodes de spécification et de conception apparaissent là où plusieurs méthodes et langages sont nécessaires pour concevoir le même système.

Les méthodes de spécification et de conception utilisent des outils qui convergent de plus en plus vers une modélisation comportementale en permettant l'intégration de plusieurs langages.

Un flot de conception hiérarchique basé sur une approche descendante (top down), commençant du niveau système vers le niveau physique, est nécessaire pour conduire le processus de conception directement avec les caractéristiques désirées. Cependant, la vérification par l'approche ascendante (bottom up) est aussi importante.

L'intérêt de ces méthodes de spécification, de modélisation et de validation est d'autant plus important lorsqu'elles sont appliquées à de nouvelles générations de systèmes complexes et hétérogènes dont les parties sont développées indépendamment.

Les deux principaux aspects qui ont fait l'objet d'étude dans le cadre de cette thèse sont le développement d'une méthode de modélisation et de simulation MSMD, appliquée à la nouvelle génération de microsystèmes autonomes.

## 2.1 Modélisation et simulation multi domaines à signaux mixtes

Les systèmes MSMD intègrent des dispositifs optiques, mécaniques, électroniques et des ASIC<sup>4</sup>. Un microsystème combinera les techniques d'intégration à très grande échelle (VLSI), avec des dispositifs de type capteurs/actionneurs, des interfaces de communication RF, et de contrôle avec des circuits analogiques sur une même (Figure II-1) ou plusieurs puces.

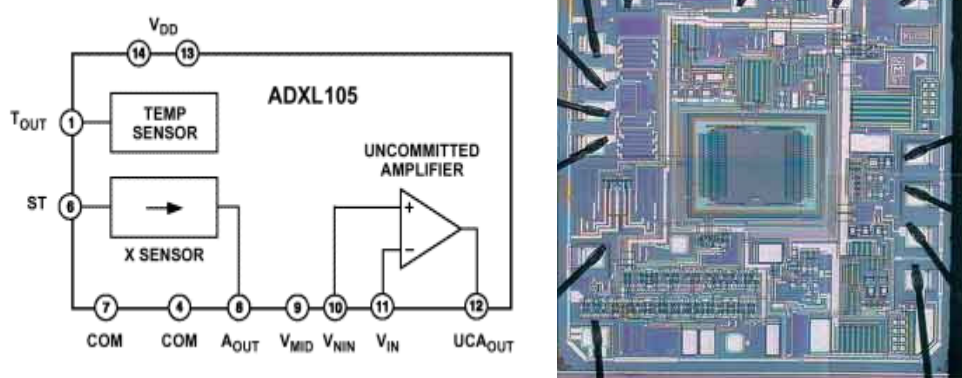


Figure II-1 : Exemple de microsystème monolithique : accéléromètre ADXL105

Les outils de conception utilisés dans un domaine spécifique comme dans la conception numérique en CMOS, sont traditionnellement basés sur l'abstraction de la description pour gérer la complexité ainsi que l'échelle d'intégration. Par conséquent, cette méthodologie de conception basée sur l'abstraction exige des compétences de spécialistes pour les divers domaines. Dans le cas de la

<sup>4</sup> ASIC- de l'anglais Application Specific Integrated Circuit

conception des systèmes MSMD, les concepteurs ayant toutes les compétences nécessaires dans les différents domaines tel que la mécanique, le fluide, l'optique et l'électronique sont rares. Donc, les outils de conception des systèmes MSMD doivent aussi bien permettre d'abstraire les détails et fournir une méthodologie de modélisation compatible avec tous les domaines. Cette méthodologie doit aussi pouvoir être accessible à un concepteur qui a des connaissances limitées dans un domaine spécifique.

D'un autre côté, la création d'outils de simulations supportant les différents domaines, et les ordres de grandeurs des amplitudes dans le temps et l'échelle de longueur<sup>5</sup> s'avère difficile. Ces difficultés sont liées directement au niveau des détails de description des modèles pour déterminer leurs performances estimées et leurs précisions.

Ils existent quelques outils pour la conception à signaux mixtes tel que : l'environnement CoventorWare™ [COV] de Coventor associé avec Cadence Design System, Inc. Cet outil rassemble les bibliothèques de MEMS et les capacités d'analyse en 3D de Coventor et le logiciel de conception des circuits intégrés de Cadence sous un même environnement de conception. Le logiciel CoventorWare permet la simulation par la méthode des éléments aux frontières (BEM)<sup>6</sup> et la méthode des éléments finis (FEM)<sup>7</sup> des composants, pour l'analyse détaillée et l'extraction des macro modèles, grâce à son module ANALYSER. Avec le module ARCHITECT basé sur le simulateur électrique Saber de Synopsys, il est possible d'intégrer des modèles en HDL.

SoftMEMS [SOF] est basé sur le langage HDL-A, et Simplorer [ANS] développé par ANSOFT est basé pour sa partie VHDL-AMS sur le simulateur hAMSter. Il permet la création de modèles en code C/C++, et de les interfacer avec d'autres environnements tel que MATLAB/Simulink ou Mathcad. Ces simulateurs fournissent un support de simulation pour des composants électriques et mécaniques avec d'autres extensions pour l'optique, la RF, et les dispositifs fluidiques.

Actuellement, la création d'un flot de conception complet pour les systèmes MSMD pour de nouveaux dispositifs présente toujours de nombreux défis. Les simulateurs doivent utiliser des méthodologies de modélisation qui supportent les différents domaines à travers différents niveaux d'abstraction et assurer des simulations plus précises au niveau comportemental et système. D'un autre côté, les flots de conception dépendent typiquement de l'extraction des paramètres au niveau physique avec plusieurs exécutions de simulations des éléments finis ou des éléments aux frontières (FEM/BEM) pour les structures physiques et d'extractions de la liste des interconnexions pour les circuits numériques.

Une méthode de conception palliative aux contraintes déjà citées, doit supporter la modélisation et la simulation d'un système global multidisciplinaire réutilisable et intégrable dans d'autres systèmes. De ce fait la création de bibliothèques de composants est nécessaire pour les différents domaines à

<sup>5</sup> Exemple : une poutre peut avoir une longueur de l'ordre des micromètres alors qu'elle peut vibrer à une fréquence de l'ordre des kHz. D'un autre côté, des phénomènes et des applications se produisent à l'échelle des secondes alors que d'autres phénomènes se produisent sur une échelle de temps plus grande. Il est nécessaire de combiner la différence entre les temps et les échelles de longueurs

<sup>6</sup> **BEM**- de l'anglais **B**oundary **E**lement **M**ethod

<sup>7</sup> **FEM**- de l'anglais **F**inite **E**lement **M**ethod

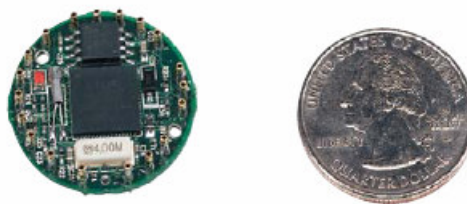
interfacer par la suite. La méthode doit aussi permettre l'élaboration de différents modèles basés sur des équations différentielles ordinaires et partielles (ODE<sup>8</sup> et PDE<sup>9</sup>). Une conception suffisamment expressive doit supporter aussi la non linéarité, et les systèmes hybrides continu-discret. Une telle approche permet de réaliser des prototypes virtuels pour un système exploitable et réutilisable. Un prototype virtuel fournit au concepteur un retour immédiat sur les décisions de conception et aussi une exploration compréhensible des performances du système. Finalement, l'utilisation du prototypage virtuel réduit de façon significative les cycles de conception et le coût.

## 2.2 Application aux microsystèmes autonomes

Beaucoup de travaux ont récemment prouvé qu'il est possible d'intégrer sur des surfaces centimétriques des systèmes complexes et hétérogènes en utilisant les technologies disponibles. Ceux-ci ont été possibles par la convergence rapide de trois principales technologies : circuits numériques, communications sans fil, et systèmes micro électromécaniques (MEMS)<sup>10</sup>.

Dans chaque secteur, les avancées technologiques au niveau du matériel et de la conception ont mené aux réductions de la taille, de l'énergie, et du coût des dispositifs. Ceci a permis, la réalisation de nœuds compacts et autonomes, pouvant chacun contenir un ou plusieurs capteurs, des possibilités de calcul, de communication, et une source d'énergie.

Parmi ces dispositifs, on trouve dans le commerce des capteurs miniatures intelligents<sup>11</sup>. La Figure II-2 montre le capteur miniature sans fil MICA2DOT de Crossbow Technology Inc [CRO], qui a la taille d'une pièce américaine d'un quart de dollar. Ce dispositif sans fil est équipé de processeurs miniatures et fonctionne avec un système d'exploitation réduit (TinyOS). Il est composé des capteurs miniatures (motes) de Berkeley et une batterie pour l'autonomie énergétique.



**Figure II-2 : Capteur sans fil MICA2DOT à côté d'une pièce d'un quart U.S (Image source : Crossbow Technology, Inc.).**

Cependant, plusieurs projets ont été lancés, pour étudier certains aspects de recherches dans la communication des réseaux de capteurs sans fils :

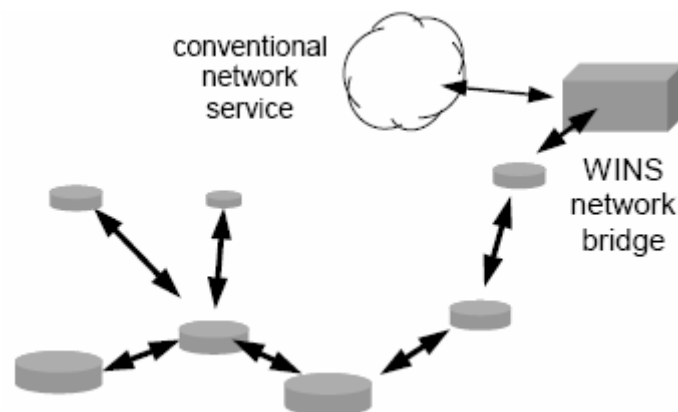
<sup>8</sup> **ODE**- de l'anglais **O**rdinary **D**ifferential **E**quation

<sup>9</sup> **PDE**- de l'anglais **P**artial **D**ifferential **E**quation

<sup>10</sup> **MEMS**- de l'anglais **M**icro**E**lectro**M**echanical **S**ystem

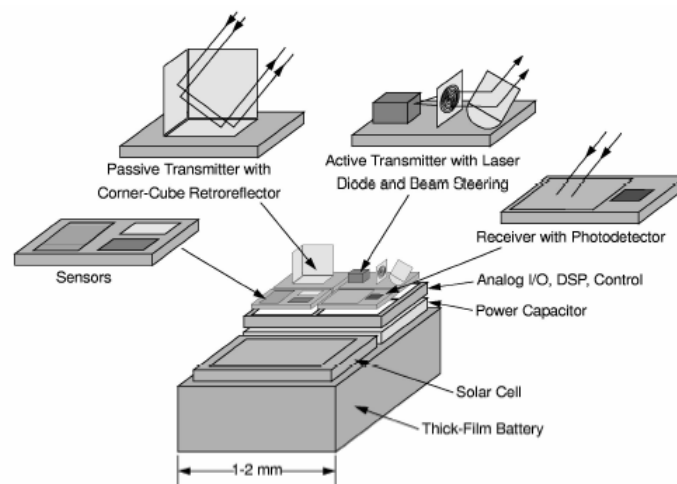
<sup>11</sup> Smart dust motes

- ⇒ Le projet WINS (Wireless Integrated Network Sensors) est développé par l'université de Californie (UCLA) en 1997, et constitue un réseau distribué avec accès Internet à faible consommation. Ce projet combine la technologie des MEMS, de traitement des signaux, de calcul, et des possibilités de gestion de réseaux sans fil dans les systèmes intégrés. Il est basé sur le développement de capteurs et d'actionneurs pouvant communiquer sur de petites distances. Les études ont été focalisées sur le développement des méthodes, des protocoles et des dispositifs de communications RF. Le réseau WINS est composé des nœuds en forme de disques dans la Figure II-3, un pont de réseau pour le transfert des informations entre les nœuds et le service réseau. Ce réseau de capteurs sans fil supporte la communication multi alimentation ce qui permet de réduire la consommation de l'énergie. Les données de chaque nœud sont transférées grâce à un lien sans fil asymétrique à un utilisateur ou à un service réseau conventionnel [Asa-98].



**Figure II-3 : Structure du réseau WINS : Les nœuds, le service réseau (ou l'utilisateur), le pont pour l'acheminement des données.**

- ⇒ Le projet **Smart Dust** est développé dans le même esprit que **WINS** par l'Université de Berkeley (1998). Il cible la fabrication d'un dispositif de capture et de communication isolé à l'échelle millimétrique et utilisé comme un nœud dans un réseau distribué. La taille du dispositif souhaitée est celle d'un grain de sable. Le système (Figure II-4) réalisé contiendra des capteurs, des dispositifs de calcul et de communications bidirectionnels, un générateur de puissance pour l'autonomie du dispositif. Le microcontrôleur embarqué permet l'exécution de tâches opératives, et la gestion du transfert de la puissance aux différents composants afin d'améliorer la consommation énergétique. La principale différence avec le projet WINS est que ce dernier est basé sur des communications RF sur de courtes distances, alors que le projet **Smart Dust** est étendu à des dispositifs optiques pouvant communiquer sur quelques centaines de mètres [Kah-00].



**Figure II-4 : Le dispositif Smart dust contient des microcapteurs, un récepteur optique, des transmetteurs passifs et actifs optiques, circuiterie de traitement des signaux et de contrôle, et une source d'énergie.**

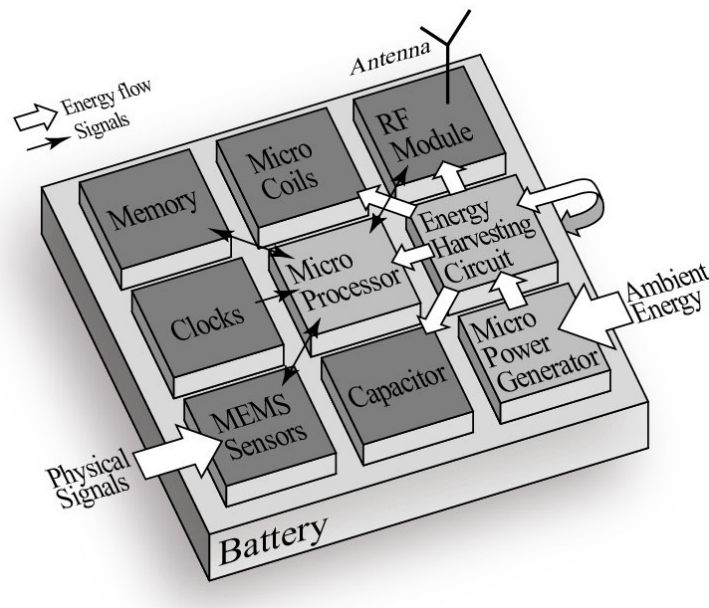
- ⇒ Le projet VIBES (Vibration Energy Scavenging) est développé par un groupe de laboratoires de recherches et d'entreprises en Europe<sup>12</sup>, et chapeauté par l'Université de Southampton UoS. Le projet se focalise sur l'étude d'un seul nœud dans le réseau sans fil. Il a pour but de développer un micro générateur de puissance capable de récupérer les vibrations environnantes (immeubles, machines, corps humain). Le dispositif produira à partir de ces vibrations une énergie électrique pouvant alimenter un microsystème. La Figure II-5 montre l'architecture d'un nœud sans fil. Le microsystème dit autonome se composera d'un micro générateur de puissance ( $\mu$ PG)<sup>13</sup>, d'un circuit de récolte d'énergie (EHC)<sup>14</sup>, d'un élément de stockage (batterie ou super capacité), d'un module de communication RF, plusieurs capteurs MEMS et microprocesseurs. Actuellement, les modules constituant le nœud du réseau sans fil en cours de développement sont : le micro générateur de puissance à partir des vibrations, le circuit de conversion et de récolte de l'énergie, l'élément de stockage et le microprocesseur à faible consommation. La conversion des vibrations et les mouvements existants dans l'environnement en une énergie électrique est effectuée par la transduction piézoélectrique. Ce mode de transduction a été choisi pour les caractéristiques physiques de ce type de matériaux ainsi que pour sa compatibilité avec les techniques de micro fabrication. Un tel dispositif permet de rendre le microsystème complètement autonome ce qui résout les problèmes de portabilité et de vie des sources d'énergie pour les réseaux de capteurs sans fil.

<sup>12</sup> TIMA (France), Tyndall (Irlande), University of Southampton (UoS), PHILIPS research et FEMTO-ST (France)

<sup>13</sup>  $\mu$ PG- pour Micro Power Generator

<sup>14</sup> EHC- de l'anglais Energy Harvesting Circuit





**Figure II-5 : Architecture du microsystème autonome du projet VIBES.**

Dans le cadre de cette thèse réalisée dans le groupe Micro et Nano Système au laboratoire TIMA, la partie de récupération de l'énergie du microsystème se compose d'un générateur de puissance, d'un circuit de récolte de l'énergie et d'un élément de stockage. Elle est étudiée et implémentée selon une méthodologie de conception multi domaines à signaux mixtes (MSMD) pour valider cette partie du microsystème. Nous allons donc la considérer comme un microsystème dans nos développements.

### 3. Contributions

Dans le cadre de cette thèse, nous avons participé à deux projets : VIBES<sup>15</sup>, et OSMOSE<sup>16</sup>. Le travail présente trois contributions principales :

- (1) Étude et analyse des différentes méthodes de conception des systèmes ; une étude qui a permis de définir la méthodologie utilisée tout au long de cette thèse pour la modélisation et la simulation d'un microsystème hétérogène sur puce. Ensuite, nous avons effectué la comparaison entre différents langages et outils de modélisation et de simulation.
- (2) Cette étude nous a amené à proposer une méthode de modélisation et de simulation des systèmes multi domaines à signaux mixtes.
- (3) Nous avons développé et appliqué la méthode proposée pour réaliser les modèles des sous-systèmes constituant le microsystème autonome. Ces modèles permettent de construire le prototype virtuel du microsystème. Ce prototype est basé sur une approche de modélisation collaborative nécessitant l'amélioration du modèle en plusieurs étapes.

<sup>15</sup> Projet européen

<sup>16</sup> Outils pour la simulation, synthèse et modélisation des systèmes sur puce (SoC) : projet région Rhône-Alpes

## **4. Plan du manuscrit**

La suite du manuscrit est structurée en deux grands chapitres composés de deux parties chacun. Dans la première partie du Chapitre II-, nous dressons une vue sur les méthodes de la modélisation des microsystemes et nous nous inspirons des méthodologies de gestion de projet pour proposer une approche de conception collaborative. Dans la deuxième partie du Chapitre II-, nous nous intéressons aux langages et aux outils de modélisation et de simulation pour proposer l'utilisation d'un environnement qui peut répondre aux enjeux de la validation fonctionnelle d'un microsysteme MSMD. Dans le Chapitre III-, nous développons le prototype virtuel du microsysteme qui est modélisé dans l'environnement choisi selon l'approche proposée. Pour ce faire, nous décomposons ce chapitre en deux parties. La première partie concerne le développement des modèles des sous-systèmes qui composent le microsysteme à différents niveaux d'abstraction. Dans la deuxième partie, nous réalisons le prototype virtuel du microsysteme autonome basé sur les modèles des sous-systèmes (Model Based Design) et nous procédons à sa validation selon deux stratégies de simulation ; la simulation globale et la cosimulation multi langages. Finalement, nous concluons la présentation des travaux réalisés lors de cette thèse et nous présentons quelques perspectives.



# Chapitre II- Méthodes de conception et outils de modélisation des microsystemes

---

A. Conception des microsystemes .....	12
1. Définition d'un microsysteme sur puce .....	12
2. Intégration des microsystemes sur puces .....	13
3. Description des microsystemes à différents niveaux d'abstraction .....	15
3.1 Modélisation au niveau système .....	16
3.2 Modélisation comportementale .....	16
3.3 Modélisation au niveau composant .....	20
3.4 Modélisation au niveau physique .....	20
3.5 Dispositif physique versus modèle des composants comportementaux .....	21
4. Flot de conception des microsystemes .....	21
5. Simulation des microsystemes .....	23
6. Prototypage virtuel .....	23
7. Méthodologies de conception .....	24
7.1 Cycle de conception en cascade .....	24
7.2 Cycle de conception en V .....	24
7.3 Méthodologie proposée : le cycle de conception en spirale .....	25
8. Bilan .....	27
B. Langages et environnement de modélisation .....	28
1. Définition des outils de modélisation .....	28
2. Modélisation collaborative .....	29
3. Langages et environnements de modélisation .....	30
3.1 Langage C++ .....	30
3.2 SystemC .....	30
3.3 VHDL-AMS et/ou Verilog-AMS .....	31
3.4 Modelica .....	32
4. Comparaison entre VHDL-AMS et Modelica .....	32
4.1 Simulateurs électriques .....	33
5. L'environnement MATLAB/SIMULINK .....	34
5.1 Processus de simulation sous Simulink .....	34
5.2 Outils de cosimulation avec MATLAB/Simulink .....	35
6. Bilan .....	38
7. Conclusion .....	39

## A. Conception des microsystemes

Les décennies de recherche et d'investissement de capitaux dans les technologies VLSI ont permis d'intégrer simplement des millions de transistors sur un millimètre carré de silicium. Ces investigations ont permis de plus d'étendre ces techniques de fabrication à des composants non électriques, comme les micro-capteurs et les micro-actionneurs. Ainsi, à partir des techniques de micro-usinage spécifiques, il est possible d'intégrer sur une même puce un système complet.

Dans ce chapitre, nous introduisons les microsystemes. Nous présentons un état de l'art sur l'intégration, la modélisation et la simulation des microsystemes complexes, pour montrer que les technologies actuelles peuvent supporter le processus de conception des microsystemes de type MSMD.

### 1. Définition d'un microsysteme sur puce

Les systèmes sur puce (SoC<sup>17</sup>) ou multi puces intégrant des MEMS sont appelés des **microsystemes**. Les microsystemes se composent de sous-systèmes qui appartiennent à différents domaines. La conception de ces microsystemes utilise conjointement diverses technologies telles les MEMS, l'électronique analogique et numérique, la RF...etc. La complexité de tels microsystemes multi-domaines est due à l'interaction existant entre les différents domaines couplés nécessitant diverses interfaces pour le conditionnement des signaux. Ces interfaces de communication entre les domaines sont directes ou indirectes selon les signaux impliqués, mais aussi selon la fonction de chaque sous-système dans le microsysteme. Le comportement de ces microsystemes est régi d'ailleurs par des phénomènes physiques qui évoluent de façon continue dans le temps et des événements qui se produisent à des espaces discrets et coordonnés dans le temps. Ces microsystemes sont à signaux mixtes ou des microsystemes hybrides. La Figure II-1 montre la composition hétérogène d'un microsysteme MSMD, avec l'interaction entre les différents domaines ainsi que les signaux échangés.

L'étude de la conception des microsystemes MSMD passe par les méthodologies, les outils et les langages qui leur sont propres. Pour notre part, nous nous sommes intéressés à définir une méthodologie générique permettant de réaliser un prototype virtuel pour des microsystemes MSMD. La méthodologie proposée est basée sur les outils dont nous disposons ; elle permet une étude fonctionnelle du prototype virtuel. L'étude fonctionnelle passe par la spécification, la modélisation et la simulation des sous-systèmes. Une telle étude prend en compte leurs natures hétérogènes (en termes d'énergie échangée et de signaux) afin de construire les interfaces lors de la construction des prototypes virtuels.

---

<sup>17</sup> SoC- de l'anglais **S**ystem **O**n **C**hip

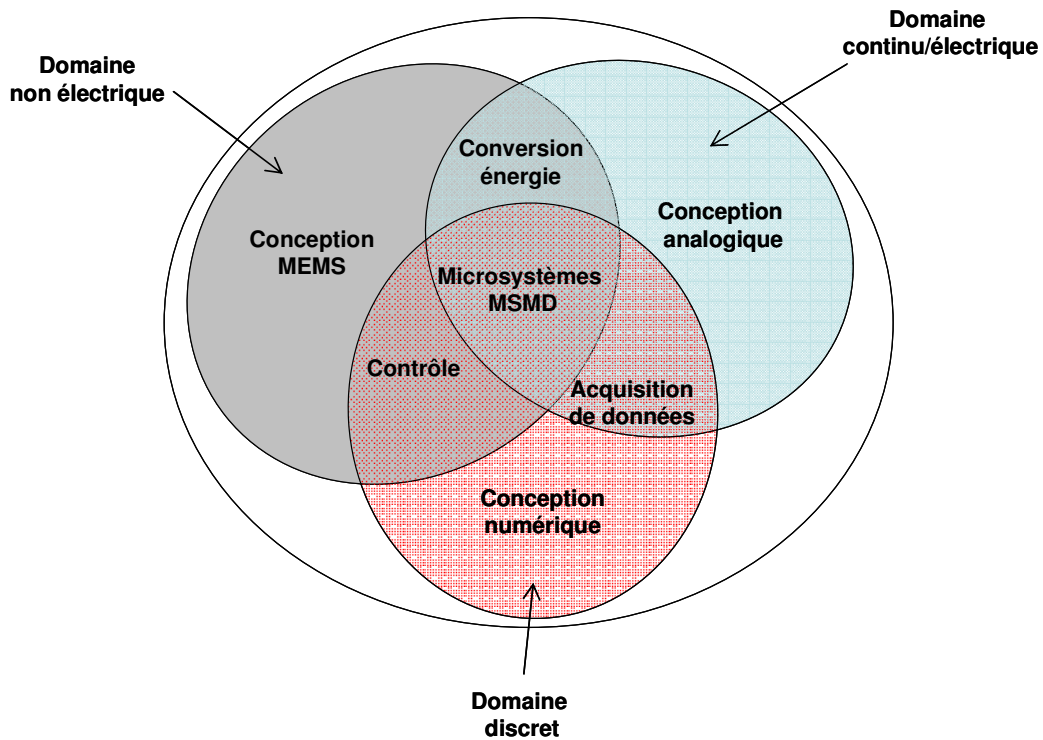


Figure II-1 : Diagramme d'un microsystème multi domaines à signaux mixtes

## 2. Intégration des microsystèmes sur puces

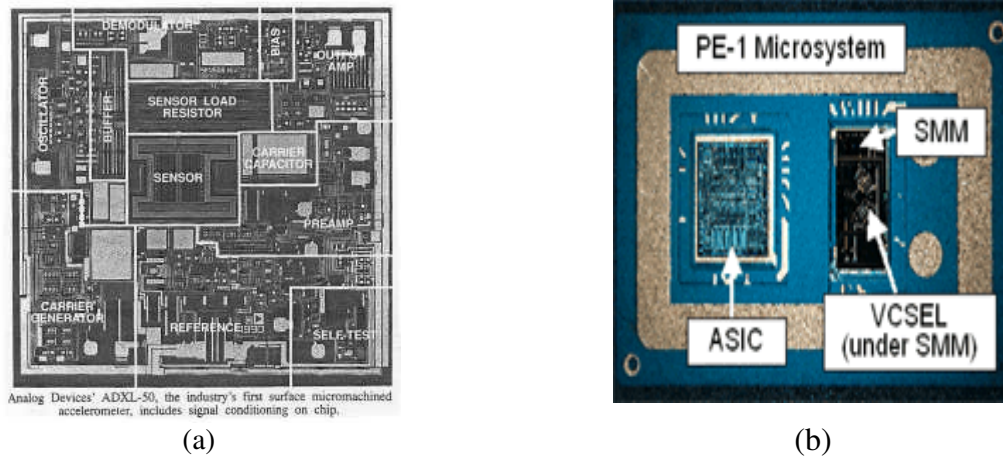
Bien que l'intégration des microsystèmes réduise les coûts de fabrication, il est encore difficile d'avoir tous les modules sur une même puce. L'avantage d'une telle intégration est d'éviter les circuits et les connexions additionnelles, ainsi que les interfaces de communication nécessaires au fonctionnement des puces.

Généralement, la fabrication efficace des circuits complexes sur un même substrat est incompatible avec celle des MEMS. Deux scénarios peuvent être considérés. Le premier scénario présente un SoC monolithique pour les composants électroniques (numérique et analogique), et les capteurs/actionneurs MEMS qui sont construits sur le même substrat (voir la Figure II-2 (a)). Le second scénario présente un SoC hybride, où les dispositifs MEMS et les composants électroniques sont construits sur des puces distinctes, selon des flots de fabrication différents (voir la Figure II-2 (b)). Ces systèmes peuvent être montés ensemble (1) sur un module multi puces (MCM)<sup>18</sup> ou un système dans un boîtier (SiP)<sup>19</sup>, séparément sur un PCB<sup>20</sup>.

<sup>18</sup> **MCM**- de l'anglais **M**ulti-**C**hip **M**odule

<sup>19</sup> **SiP**- de l'anglais **S**ystem in **P**ackage

<sup>20</sup> **PCB**- de l'anglais **P**rinted **C**ircuit **B**oard



**Figure II-2 : Microsystèmes monolithique (a), microsystème multi chip (b)**

Nous avons indiqué dans la section (1) qu'un microsystème est constitué de sous-systèmes. Ces sous-systèmes peuvent être classés en trois parties fonctionnelles. La première partie est constituée par l'ensemble des microcapteurs et des micro-actionneurs, qui peuvent être des MEMS. La seconde partie représente les circuits électroniques analogiques. Enfin, la troisième partie concerne les circuits numériques (FPGA, ASIC...etc.).

La réalisation des fonctions citées ci-dessus repose sur les outils et les techniques de fabrication déjà matures dans le monde industriel.

- Les circuits électroniques numériques sont décrits avec des langages de haut niveau comme VHDL ou Verilog. Les modèles sont compilés et synthétisés pour réaliser le layout du masque, dans une technologie de fabrication spécifique à chaque fois.
- Les circuits analogiques sont moins faciles à réaliser indépendamment de la technologie. Pour pallier ce problème, il convient d'enrichir le VHDL et le Verilog pour les signaux mixtes. Les circuits analogiques contiennent en général moins de composants que ceux numériques, et peuvent être adaptés manuellement pour les interfaces électroniques. Une autre possibilité consiste en la définition de tous les circuits d'interface sous la forme d'un ensemble de primitives analogiques génériques (amplificateur opérationnel, multiplexeurs, commutateurs, oscillateurs). Les circuits d'interface peuvent être rapidement portés à un processus de fabrication qui supporte l'implémentation des bibliothèques analogiques génériques. Actuellement, l'ensemble des primitives génériques n'est pas suffisamment mûre pour adopter cette approche.
- Les microcapteurs et micro-actionneurs actuels ne sont pas conçus avec une technologie unique. D'ailleurs, il n'y a pas vraiment de méthode générique pour spécifier ces dispositifs pouvant être implémentés dans de multiples technologies de fabrication. Actuellement, la conception de tels composants est réalisée manuellement et adaptée à chaque technologie. Pour les années à venir, on peut penser que chaque technologie de fabrication disposera de ses propres bibliothèques de composants.

### 3. Description des microsystèmes à différents niveaux d'abstraction

Pour la conception d'un système, il convient de considérer celui-ci à différents niveaux d'abstraction afin de réaliser différentes analyses. L'analyse haut niveau en premier, constitue l'étape conceptuelle où quelques détails seulement sont connus. Davantage de détails sont obtenus à la fin du processus de conception.

Dans le Tableau II-1, on présente les différents niveaux d'abstraction pour les trois domaines : le niveau système, le niveau comportemental, le niveau circuit (composant) et le niveau physique. Ces niveaux sont développés dans les sections suivantes. Nous insistons dans ce qui suit sur la modélisation des parties non électriques de type MEMS. En effet, nous nous sommes intéressés à l'intégration de ces sous-systèmes dans un flot de conception hétérogène en se basant sur les techniques qui existent déjà.

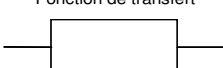
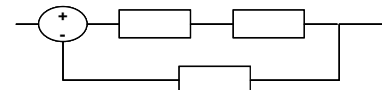
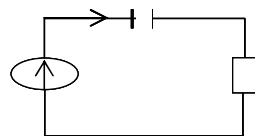
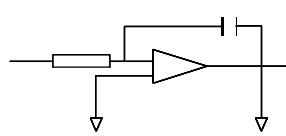
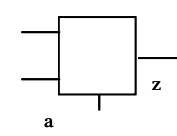
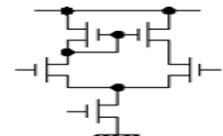
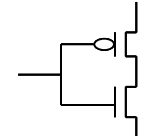
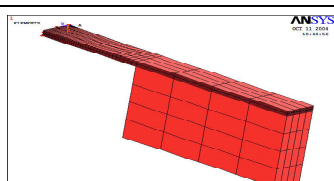

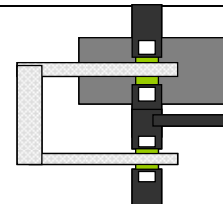
Niveaux	MEMS	Électronique analogique	Électronique numérique
système	Fonction de transfert 		if a='1' then z= x else z= not (y) end if
comportemental			x y 
Circuit	$\begin{cases} \ddot{\delta} = \frac{-K_{sp}}{m}\delta - \frac{b_m b^{**}}{m}\dot{\delta} + \frac{K_{sp}ad}{2mt_c}V + b^*\ddot{y} \\ \dot{V} = \frac{-2d\gamma_c t_c}{a\epsilon}\dot{\delta} \end{cases}$		
physique			

Tableau II-1 : Les niveaux d'abstraction pour les domaines mécanique, électronique analogique et électronique numérique



### **3.1 Modélisation au niveau système**

Le modèle au niveau système peut être vu comme un espace, où un certain nombre de fonctions et de relations logiques sont satisfaites. Dans cet espace, il est important de garantir un fonctionnement et des liens corrects entre les modèles des sous-systèmes qui représentent le comportement global du microsystème. Cet espace est partitionné et simplifié selon plusieurs sous-systèmes qui sont importants lors de la conception du microsystème. Le modèle global doit fournir les fonctions connectant les différents domaines d'énergie (ex : interfaçage entre les convertisseurs électriques et les modules non électriques incluant des fonctions internes).

A ce même niveau, les sous-systèmes peuvent être considérés comme des boîtes noires qui interagissent entre elles avec des interfaces discrètes. Ces sous-systèmes peuvent être aussi modélisés avec les équations différentielles (DAE ou ODE) [Asc-98], et/ou les spécifications à événements discrets [Zei-00]. Toutefois, l'utilisation d'un cadre orienté objet laisse un large degré de liberté d'abstraction et de flexibilité pour la simulation de tels systèmes [Buc-94].

### **3.2 Modélisation comportementale**

Une fois la méthodologie de la modélisation comportementale choisie, les interactions entre les composants des différents domaines sont prises en compte, ainsi que l'environnement de conception. Cette performance dépend de la méthode de simulation et des signaux de caractérisation choisis. Deux approches de modélisation et de simulation, au niveau comportemental, ont été définies : la modélisation fonctionnelle et la méthode par circuit équivalent. Nous les présentons dans ce qui suit.

#### **3.2.1 Modélisation fonctionnelle**

Cette approche de modélisation fonctionnelle est une alternative prometteuse pour la modélisation des systèmes hybrides. Elle tire profit des niveaux d'abstraction possibles et des langages offerts dans le cadre de la simulation à signaux mixtes. Il est cependant nécessaire de mettre le point sur les limitations que présente cette méthode. Durant la description d'un système, le concepteur doit spécifier, en suivant cette méthode, les relations définissant l'interaction entre les différents signaux dans le système. La définition de ces relations est non triviale dans le cas des microsystèmes multi domaines, ce qui implique aussi la caractérisation des interfaces aussi. La Figure II-3 montre l'interface existant entre les composants du modèle d'un système hybride qui sont dans des domaines d'énergie différents.

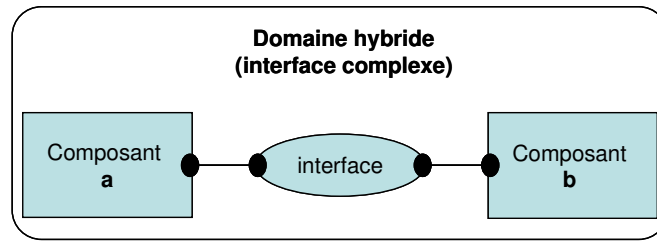


Figure II-3 : Interaction entre les composants dans les domaines hybrides

Actuellement, les dynamiques de ces interfaces doivent être modélisées explicitement par le concepteur. Cependant, les informations à extraire des modèles d'interfaces sont souvent fournies par le concepteur. Dans d'autres cas, l'information requise pour déterminer les modèles d'interfaces peut être obtenue automatiquement à partir des outils de conception d'un domaine spécifique qui sont intégrés dans l'environnement de modélisation.

### • Modélisation basée sur le graphe

Les graphes ont été utilisés pour représenter les systèmes interconnectés dans plusieurs domaines [Ses-61] [Tri-85]. Pour la modélisation des systèmes, il existe en effet trois types de paradigmes basés sur les graphes : les graphes de liaisons (bond-graph), les graphes linéaires, et les diagrammes blocs [Sin-01]. La Figure II-4 montre les modèles en diagramme bloc et en graphe de liaison d'un circuit électrique simple.

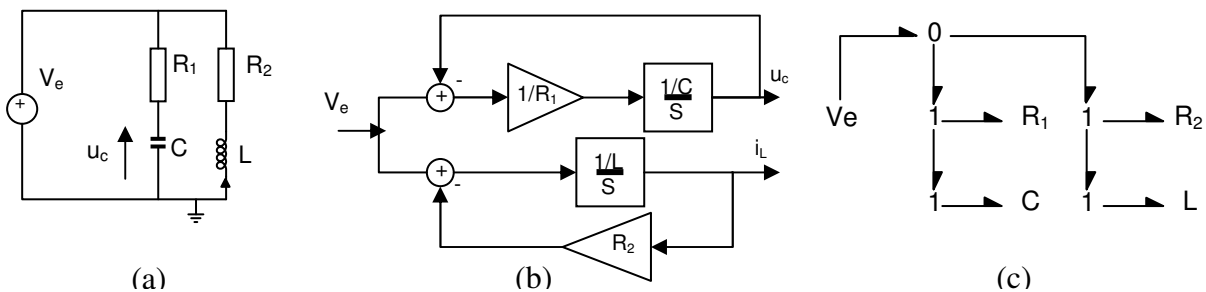


Figure II-4 : (a) Circuit électrique, (b) Diagramme bloc, (c) Graphe de liaison

### • Modélisation par graphe de liaison

La modélisation par le graphe de liaison (bond-graph) [Pay-61] [Ros-83] se situe entre le système physique et les modèles mathématiques classiques. La technique de modélisation par graphe de liaison est basée sur la caractérisation des phénomènes d'échange de puissance (par un lien) au sein du système. L'idée de la modélisation par graphe de liaison consiste en la représentation des puissances dissipées comme produit flux/effort et à leur liaison par les différentes jonctions pour reproduire le système. La méthode supporte les lois de Kirchhoff. D'autre part, la modélisation des éléments linéaires et non linéaires est supportée par cette méthode [Bre-97]. Cependant, le graphe de liaison ne convient pas à la modélisation au niveau physique des systèmes mécaniques [Tri-85] et les

systèmes hybrides (continu-discret) [Eds-99] [Fer-95]. La simulation dynamique avec les graphes de liaison n'est pas supportée.

- **Modélisation par diagramme bloc**

Les diagrammes blocs sont utilisés pour la modélisation des flots de signaux échangés en entrées/sorties d'un système. Les modèles créés contiennent des fonctions primitives décrivant les relations entre les entrées et les sorties (ex : Simulink, SPW...). Les systèmes complexes sont partitionnés en sous-systèmes selon une configuration hiérarchique.

Chaque modèle dans les diagrammes blocs est généralement défini comme une procédure. Ainsi, les solveurs ne peuvent pas rompre les boucles algébriques, et le concepteur doit dans ce cas réorganiser ces modèles manuellement pour éviter ce problème.

### 3.2.2 Modélisation comportementale basée sur le circuit électrique équivalent

Cette méthode de modélisation comportementale est basée sur des modèles définis sous la forme de circuits électriques équivalents. Ces circuits équivalents représentent les sous-systèmes dans d'autres domaines d'énergie. Comme le montre la Figure II-5, dans cette méthode de modélisation, l'interaction entre les composants est un lien naturel et l'interface est le nœud du réseau dans le domaine électrique. Par conséquent, le modèle équivalent électrique peut être simulé en utilisant n'importe quel simulateur de circuits électriques connu (ex : SPICE, iSMILE [Mor-97], SABER). Cette méthode est utilisée pour la simulation des dispositifs mécaniques, et dont l'équivalent représente la liste des interconnexions (netlist) sous SPICE. Divers exemples de dispositifs mécaniques utilisant cette méthode, ont été développés dans [Van-97] [Ana-96], et des interconnexions optoélectroniques ont été réalisées par [Yan-89]. Cette méthode présente toutefois des limitations pour la modélisation hiérarchique et la cosimulation.

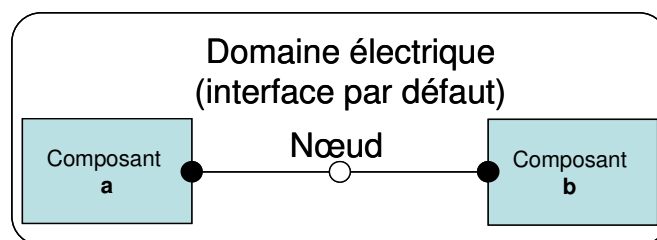


Figure II-5 : Interaction entre les composants dans le domaine électrique

- **Principe de la modélisation électrique**

Le comportement d'un composant peut être décrit grâce aux équations différentielles, dont la forme dépend de la nature physique de son activité. Cependant, pour les systèmes non électriques, les modèles mathématiques sont utilisés pour la génération des équivalences dans le domaine électrique. Pour ce fait, les lois de la physique sont appliquées à ces modèles, à partir desquelles des analogies

entre les domaines sont établies. On définit deux variables globales (voir la Figure II-6) qui sont le "*flux*" (correspond à : le courant, la vitesse, la température...) et "*l'effort*" (correspond à : la tension, la force,...). Les éléments non électriques peuvent être ainsi convertis en composants électriques équivalents. Pour résoudre le modèle électrique, des méthodes bien connues ont été élaborées permettant l'utilisation de ces circuits électriques.

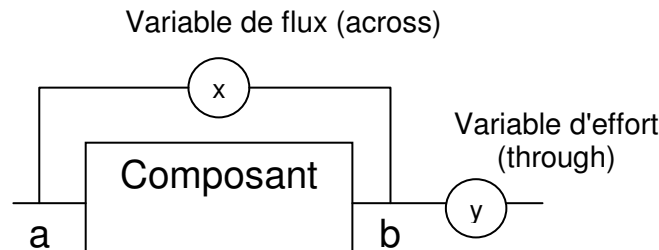


Figure II-6 : Variable terminale : flux et effort

### • Les réseaux de Kirchhoff généralisés

Les circuits électriques sont souvent modélisés grâce à des réseaux de Kirchhoff. De plus, le concept des réseaux de Kirchhoff est utilisé pour d'autres domaines [Ton-76]. Dans ce cas, ils sont appelés les réseaux de Kirchhoff généralisés, et le modèle réalisé représente le circuit électrique équivalent d'un composant [Voi-97]. La modélisation d'un système utilisant ces réseaux repose sur la manipulation des quantités *flux*<sup>21</sup> et *effort*<sup>22</sup>. Ces quantités sont définies pour plusieurs domaines par analogie (voir annexe). Les liens existant entre les composants sont les nœuds et leurs modèles sont des relations mathématiques entre les flux et les efforts qui sont formalisées par des diagrammes. Les équations des quantités sont implicites, explicites et différentielles ; elles obéissent aux lois de conservation de l'énergie (lois des nœuds et lois des mailles).

L'un des principaux avantages de la modélisation avec ces réseaux est la similitude entre la structure du système et la topologie du réseau, qui est facile à voir lorsque le réseau est ramené à une description schématique du système. Ceci mène directement au concept de la *modélisation orientée objet*, si cette notion est interprétée comme une méthode de modélisation hiérarchique et modulaire [Cel-91] [Sch-00]. Les informations masquées, et la commutation entre les différents niveaux du modèle, peuvent être réalisées par une claire distinction entre l'interface et le noyau du modèle, mais aussi par l'échange des données entre les modèles à travers leurs interfaces. Ce concept est indépendant des langages de programmation orientés objet ; il peut en revanche être supporté par de tels langages ou des langages de modélisation orientés objet.

La description mathématique unifiée est l'approche souhaitée, du fait que la modélisation est indépendante du simulateur [Sch-00]. Cependant, la résolution des équations différentielles des

<sup>21</sup> Through variable

<sup>22</sup> Across variable

circuits de Kirchhoff revient à employer l'analyse nodale modifiée (MNA) utilisée en électronique. Plusieurs simulateurs comme SPICE, ELDO [ELDO], SABER [SAB] utilisent l'approche MNA.

Les approches mathématiques utilisées pour la modélisation des MEMS sont les équations différentielles partielles (PDE), les équations différentielles ordinaires (ODE), ou les équations différentielles algébriques (DAE). L'application de ces approches dépend de la méthode de modélisation appropriée et le choix du simulateur [Sen-98] [Muk-00]. Par ailleurs, la modélisation par analogie entre les domaines physiques est utilisée pour réaliser le système complet. La technique se base sur le concept des réseaux de Kirchhoff généralisés. Ces réseaux n'ayant pas de dimensions, ils permettent l'analyse d'un système composé de divers éléments liés entre eux [Sch-00]. En effet, les réseaux du système sont construits puis implantés et simulés ensemble.

### 3.3 Modélisation au niveau composant

A ce niveau, le sous-système est décomposé en briques élémentaires basiques qui sont décrites par des modèles analytiques. Des approches numériques basées sur la discrétisation des équations partielles (PDE), peuvent aussi être appliquées (se référer au Chapitre II-A.3.4). Ceci mène à un système d'équations différentielles ordinaires (ODE), ou aux équations différentielles algébriques (DAE) utilisées pour des applications plus compliquées. Dans le cas des simulateurs par éléments finis (FEM), ces équations sont formulées et résolues automatiquement ; leur nombre est directement lié aux mailles discrétisées et aux dimensions des composants.

L'exportation des équations différentielles ordinaires est possible si l'ordre du système d'équations est réduit, afin de constituer le modèle simplifié du composant. Il est évident que la précision du modèle est réduite, mais l'erreur liée peut être acceptable si le modèle d'ordre réduit prévu n'est pas très petit.

Certains simulateurs, comme ANSYS, incorporent des algorithmes pour la réduction d'ordre du modèle, et offrent l'opportunité d'exporter les équations d'un modèle réduit. En effet, le modèle d'ordre réduit (ROM)<sup>23</sup> peut être utilisé pour la modélisation aux niveaux comportementaux ou systèmes. Cependant, il n'existe pas de modèles réduits pour tous les couplages possibles.

### 3.4 Modélisation au niveau physique

Le niveau d'abstraction physique (ou dispositif) est le plus proche de la réalité du fonctionnement d'un système. Il est évident que lors de la réalisation du modèle à ce niveau, les propriétés des matériaux employés et des structures doivent être introduites dans la description du comportement des dispositifs.

A ce niveau de modélisation, le comportement d'une structure, est décrit par exemple, par les équations différentielles partielles (PDE). Ici, les méthodes de résolution utilisées sont les éléments

---

<sup>23</sup> Reduced Order Modeling

finis (FEM) et les éléments aux limites (BEM). Ces méthodes de résolution sont précises, et conviennent au calcul des distributions des contraintes, des déformations et des fréquences naturelles des structures MEMS.

Cependant, les simulations FEM et BEM sont souvent des processus gourmands en temps et en ressources dans la conception des MEMS complexes.

La modélisation des propriétés physiques des dispositifs constitue nécessairement des paramètres d'entrée pour les modèles des niveaux supérieurs. Cependant, pour ce genre de développement, il est important d'extraire des résultats interprétables, afin d'exploiter l'impact des matériaux et des structures sur le comportement réel des dispositifs.

### **3.5 Dispositif physique versus modèle des composants comportementaux**

Les techniques de simulation au niveau composant offrent un degré de précision requis pour modéliser le régime temporel, et les dépendances géométriques de fabrication ainsi que le régime permanent [Mor-97]. Cependant, la modélisation de ces processus requière des ressources de calcul et des techniques spécialisées. D'autre part, les résultats de simulation obtenus ne sont pas en général compatibles avec d'autres simulateurs dans d'autres domaines.

Pour résoudre le problème de la simulation physique du dispositif dans un environnement multi domaines, il est possible d'utiliser un simulateur pour chaque domaine à part, couplé à d'autres domaines à travers un processus de coordination à haut niveau qui gère leur comportement en termes d'énergie et de temps de leur processus physique commun. Cependant, l'inconvénient de cette technique, comme nous l'avons déjà mentionné pour la simulation au niveau composant, en plus de la condition de coordination entre les simulateurs, est la convergence vers un point de fonctionnement [But-81] [Pin-84]. Pour la simulation au niveau système, les coûts en temps de cette méthode sont prohibitifs.

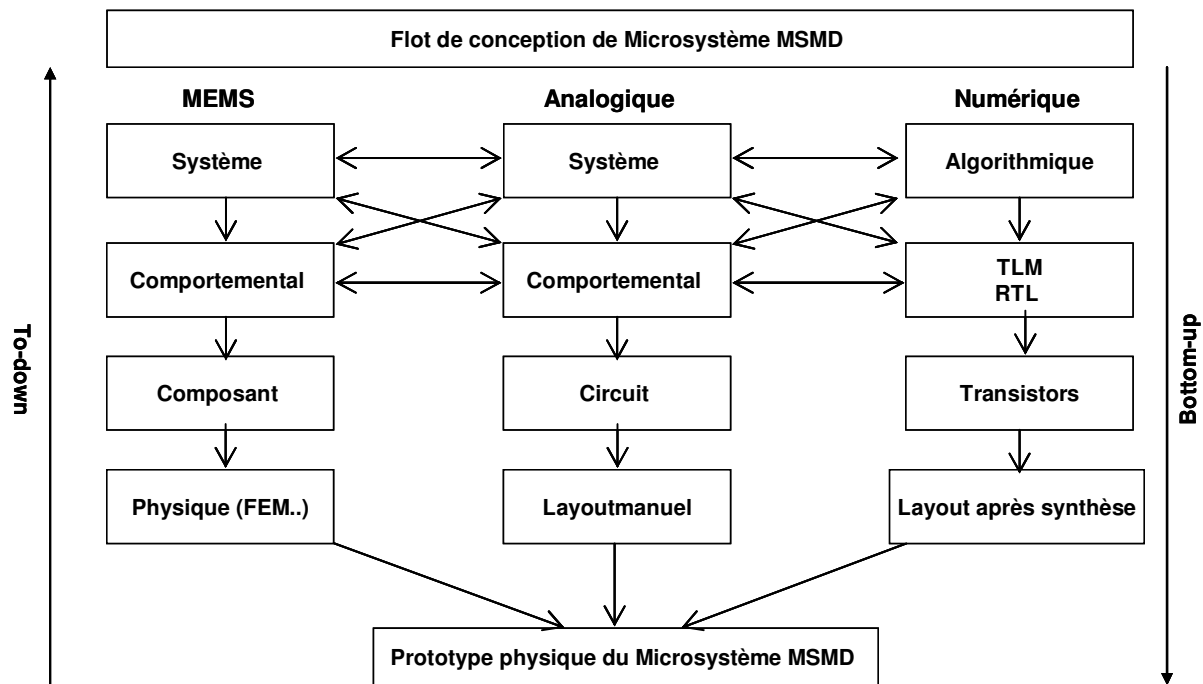
D'autre part, les modèles comportementaux peuvent être conçus en prélevant les principaux comportements du dispositif à partir des processus physiques. Tandis que cette technique ne permet pas une fidèle modélisation, la simulation au niveau système des modèles comportementaux peut fournir des résultats acceptables.

## **4. Flot de conception des microsystèmes**

Après avoir défini les différents niveaux d'abstraction, la conception des microsystèmes peut être réalisée selon deux approches. La première est appelée l'approche descendante (top-down) et la deuxième est l'approche ascendante (bottom-up).

Au premier stade du développement, il se trouve que les procédures de conception sont au niveau recherche, et par conséquent, l'approche ascendante est plutôt favorisée. Mais si les technologies utilisées sont maîtrisées, et leurs capacités et propriétés sont connues, l'approche descendante, dans ce

cas, est la mieux adaptée au processus du développement industriel. La Figure II-7 illustre ces deux approches dans un flot de conception.



**Figure II-7 : Flot de conception des éléments d'un microsystème à différents niveaux d'abstraction**

Le développement selon une approche de conception descendante, doit se faire du niveau système vers le niveau physique (voir la figure ci-dessus). Les modèles raffinés au niveau physique sont complétés, et la performance réalisée peut être testée et confrontée aux spécifications d'origine tout en utilisant les blocs abstraits et le modèle définis au niveau système. La vérification entre des domaines mixtes à différents niveaux d'abstraction doit être effectuée. La simulation multi niveaux est actuellement considérée comme la meilleure approche pour la vérification des systèmes à signaux mixtes complexes [Kun-00]. Une telle méthode permet de réduire le temps consommé dans la conception récurrente lors de l'assemblage du système.

Par ailleurs, la vérification d'un modèle avec les simulations est faite à tous les niveaux d'abstraction :

1. La fonctionnalité de chaque composant est vérifiée par un modèle idéal lors des simulations au niveau système.
2. Ensuite, le comportement du système est encore vérifié en remplaçant le modèle idéal par un modèle au niveau composant.
3. Cette vérification permet après d'observer l'effet des détails au niveau composant (ou circuit) sur les performances du système.
4. Enfin, le modèle physique est substitué au modèle composant. En comparant les résultats des simulations pour les modèles au niveau composant et au niveau physique (ou dispositif), le comportement et la précision du modèle doivent être vérifiés.

L'approche de conception ascendante (bottom-up) implique le développement de chaque composant au niveau physique jusqu'au niveau système. Les composants sont combinés pour former les blocs, et l'assemblage de ces derniers constitue le microsystème. Cette hiérarchisation ascendante, permet d'implémenter le comportement réel des sous-systèmes à partir de leurs domaines d'énergie.

Une fois le développement des modèles correctement réalisé, la vérification ascendante traite un système plus complexe. Au niveau comportemental, l'exécution des simulations reste rapide. Ceci se fait par l'abandon de l'implémentation des détails des niveaux inférieurs subordonnés aux détails du niveau comportement. Les modèles comportementaux produits lors de la vérification ascendante, peuvent être utiles d'un côté, pour la réutilisation et l'évaluation d'autres systèmes. D'un autre côté, les détails supplémentaires peuvent nécessiter l'implication de concepteurs spécialisés, qui généralement, ne sont pas au sein d'un même groupe de travail (MEMS, systèmes analogiques et mixtes, RF...etc.).

## **5. Simulation des microsystèmes**

La simulation est particulièrement importante pour la conception des systèmes hétérogène, ayant des composants dans différents domaines, et qui de plus sont couplés dans le but de réaliser des performances optimales.

De nos jours, la simulation des microsystèmes complexes est caractérisée par:

- La conception au niveau composant en utilisant des solveurs d'équations différentielles partielles (PDE), comme la méthode des éléments finis, employée pour les différents domaines (structures mécaniques, phénomènes thermiques, propagation des ondes...etc.).
- La conception des circuits analogiques et numériques basée sur la réutilisation de concepts pré-élaborés et automatisés selon une approche top down.
- La conception des systèmes basée sur les diagrammes blocs (ex : MATLAB/SIMULINK de Math Works, SPW de Cadence, System Studio de Synopsys)
- Et l'augmentation de l'intégration HW/SW (de FPGA, de microcontrôleurs, de DSP et microprocesseurs).

## **6. Prototypage virtuel**

La modélisation et la simulation constituent une méthode de "*prototypage virtuel*", qui permet au concepteur de tester les spécifications d'un concept. Le prototypage virtuel permet aussi de réduire considérablement le flot de conception, et d'éviter le prototypage physique répété. De plus, le prototypage virtuel permet au concepteur d'avoir un retour immédiat, lui permettant ainsi de prendre les décisions relatives à la conception d'un système. Une telle approche permet l'exploration d'alternatives lors de la conception et donc l'exécution efficace du concept final. Enfin le but essentiel du prototype virtuel est la validation d'un système avant de réaliser le prototype réel et la réduction du temps et du coût de conception.



## 7. Méthodologies de conception

Dans les sections précédentes, nous avons défini aussi bien les niveaux d'abstraction d'un système à travers lesquels un prototype complet de celui-ci doit se faire. Nous avons aussi défini le flot décrivant la méthode de conception d'un microsystème MSMD. Dans ce qui suit, le problème majeur lors du développement des modèles pour les microsystèmes complexes, est l'adaptation des phases entre les méthodes de conceptions, afin de tirer les meilleurs profits des ressources mise en œuvre. C'est pourquoi, nous avons étudié les méthodologies de conception telles que le cycle en cascade et le cycle en V, et nous préconisons dans cette thèse la méthodologie de conception collaborative (cycle en spiral) comme support au prototypage virtuel de microsystèmes MSMD.

### 7.1 Cycle de conception en cascade

On rappelle que c'est la méthodologie la plus couramment utilisée ; elle est basée sur les phases suivantes : spécification, conception du système, réalisation détaillée et tests unitaires, et enfin intégration ou prototypage. Comme le montre la Figure II-8, cette approche nécessite l'achèvement de chaque phase, ce qui implique que tout changement sera au niveau des spécifications initiales. Cette approche constitue une démarche basée sur l'intervention de spécialistes dans la conception, ce qui rend la tâche difficile lorsque le système est pluridisciplinaire. L'enchaînement qui reste simple et intuitif contient des risques et des coûts de conception non mesurés.

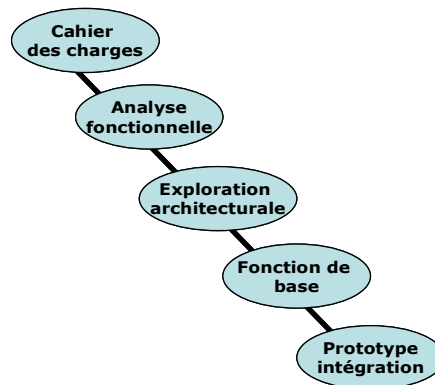


Figure II-8 : Les phases du cycle en cascade

### 7.2 Cycle de conception en V

Le principe de ce modèle (voir la Figure II-9) est la décomposition d'un projet de conception d'un système en phases distinctes. Toute phase de description d'un composant est accompagnée de tests permettant d'assurer les spécifications correspondantes. Ceci rend explicite la préparation des dernières phases (validation et vérification) par les premières phases (spécification et conception), et

permet ainsi d'éviter, à titre d'exemple, d'énoncer une propriété qui est impossible de vérifier objectivement après la réalisation.

Ce cycle en V convient à la réalisation de projets, dont la finalité est bien connue (tel que la conduite des projets en informatique (codage d'une application ou d'une fonction)). Dans ce cycle on distingue l'enchaînement et l'itération entre les phases de développement et de tests. Cette approche nécessite la préparation et la description complète des phases suivantes et les tests des systèmes. Cette méthode a pour avantage de proposer, au fur et à mesure, une démarche de réduction des risques, tout en minimisant l'impact des incertitudes le long du processus de réalisation. Par conséquent, dès les premières phases, la conception devient plus technique : d'où l'exclusion de personnes non spécialistes ou spécialistes dans d'autres domaines. En effet, une telle démarche suscite la décomposition et la distribution des tâches d'un système complexe en plusieurs sous-systèmes qui seront étudiés à part par des experts. La vérification du système global ne survient qu'en fin de projet. En plus de la définition des procédures ultérieures et des tests, le cycle en V présente deux autres inconvénients. D'une part, la communication et l'échange entre les projets qui ne peuvent se faire qu'à la fin. Et d'autre part, le changement des spécifications en cours d'un projet peut s'avérer difficile et nécessite le redémarrage du cycle de développement.

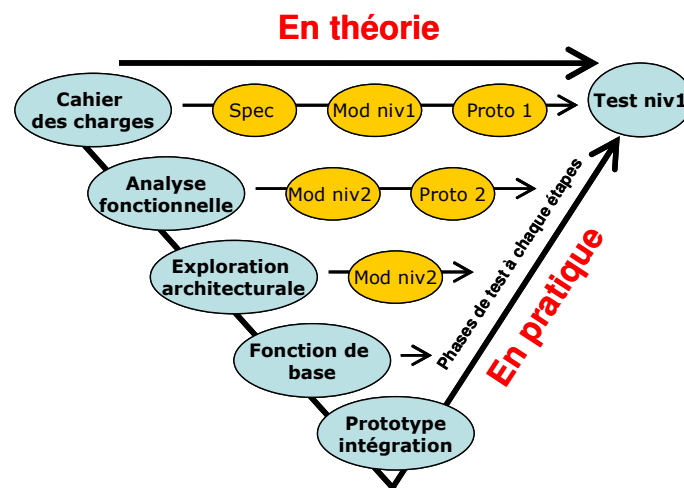


Figure II-9 : Le modèle de conception en V

### 7.3 Méthodologie proposée : le cycle de conception en spirale

Les bases du modèle en spirale ont été établies par Barry Boehm [Boe-88]. Cette méthode permet à l'intégralité de la démarche de conception d'être appliquée de nouveau sur le résultat obtenu par un cycle précédent. Dans le contexte de notre étude, la modélisation du microsystème représente la description des différents sous-systèmes d'une manière indépendante et couplée. L'indépendance concerne la possibilité d'appliquer pour chaque sous-système son propre cycle de conception et le couplage, lui ajoute le contrôle de son comportement. La prise en compte du couplage assure la

capacité de concevoir et d'exécuter les comportements entre les modèles des sous-systèmes en fonctions des évènements externes.

La Figure II-10 présente le cycle en spirale, tel que nous le préconisons dans une démarche de conception de microsystèmes MSMD. Ceci permet une grande réactivité aux changements des spécifications ainsi que la collaboration entre les différents partenaires. Pour respecter cette méthode, chaque cycle se déroule en quatre phases qui sont les suivantes.

- Détermination des spécifications, des alternatives et des contraintes d'un cycle, à partir des données du cycle précédent, et pour le premier cycle à partir des besoins et des cahiers de charge en début du projet. Pendant cette phase, les objectifs généraux sont définis en termes de caractéristiques fonctionnelles à supporter par la plate-forme de simulation (ex : MATLAB/Simulink) et en termes de résultats attendus pour le microsystème considéré. En ce sens, cette phase intègre la définition d'un ensemble de plans permettant de réviser et d'évaluer les spécifications dues à l'étude des phénomènes émergents dans les systèmes complexes.
- Développement et vérification des modèles et des tests unitaires (des sous-systèmes). Au niveau de cette phase, les raffinements successifs des descriptions relatives au système complexe sont réalisés par des spécialistes. Ceci, conduit au développement de prototypes décisifs au fur et à mesure de l'avancement du développement, par collaboration dans la prochaine phase. La phase se termine lorsque les expériences de validation peuvent se dérouler avec succès. Une modélisation adoptant le cycle en V pour chaque sous-système peut se superposer au sein de cette phase.
- Conception et développement de prototypes virtuels de plus en plus complexes (au premier niveau, c'est l'étude d'un prototype simulable), analyse des risques, et évaluation des alternatives à chaque niveau de la spirale. Au cours de cette phase, les concepteurs système implémentent les prototypes à partir des modèles développés dans la phase précédente. Seuls les éléments spécifiques au problème de simulation par domaine peuvent être retenus dans cette phase (plus haut niveaux d'abstraction). Les mécanismes génériques au domaine d'application considéré sont identifiés ; ils feront cependant l'objet d'une stratégie de validation définie dans la dernière phase. Cette phase conduit à fournir un nouveau prototype virtuel du microsystème à chaque cycle de la spirale.
- Revue des résultats et planification du cycle suivant. Cette phase intègre pour ce faire la définition d'un ensemble de stratégies de validation, permettant de valider les prototypes des sous-systèmes et du système final.

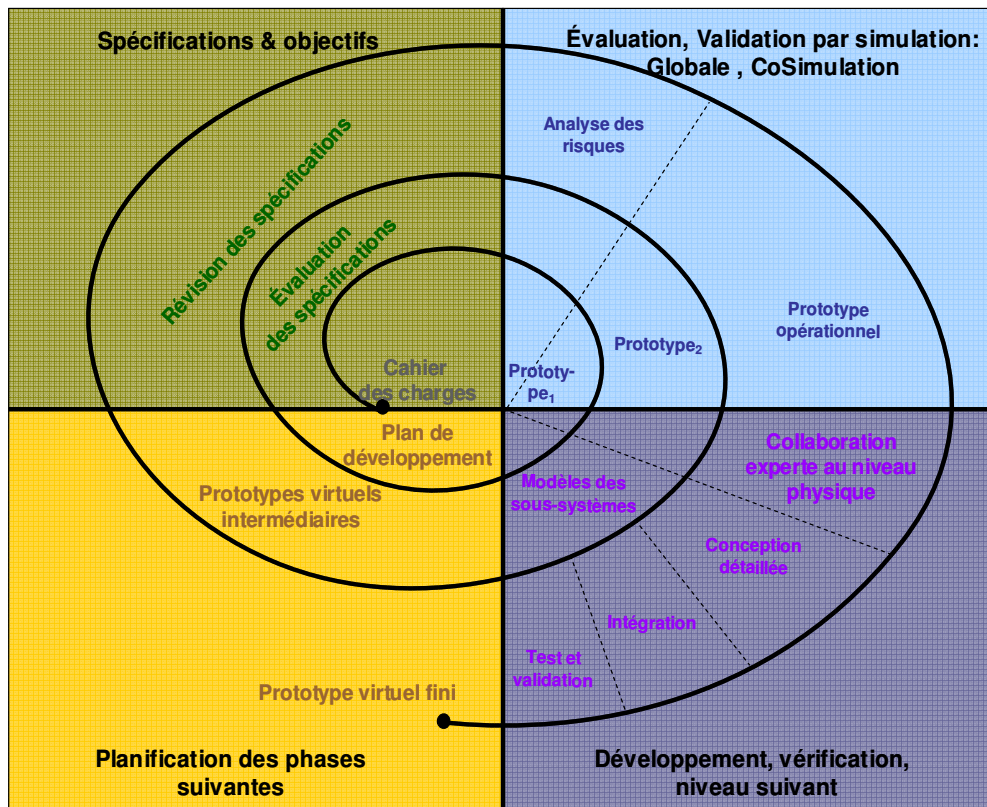


Figure II-10 : Les phases du cycle de conception en spirale

## 8. Bilan

Dans cette première partie, nous avons précisé la problématique liée à la conception des microsystemes MSMD, et présenté les différentes méthodes de modélisation à différents niveaux d'abstraction. Ces méthodes sont nécessaires lors de mise en œuvre de notre approche de modélisation collaborative. Cette approche permet d'élaborer un échange entre les différents concepteurs afin de réaliser le prototype virtuel décrivant au mieux le comportement d'un microsysteme. L'approche collaborative convient aux microsystemes complexes, dont les sous-systèmes les interfaces entre les domaines peuvent être développés et raffinés jusqu'à l'obtention des performances souhaitées. La simulation d'un système complet selon cette approche doit supporter l'échange de signaux différents et la réactualisation de ces derniers.

Dans la partie suivante, nous présentons les langages et les environnements, nous permettant d'appliquer l'approche de modélisation préconisée dans ce chapitre.

## **B. Langages et environnement de modélisation**

Dans le chapitre précédent, nous avons proposé une méthodologie de conception collaborative, dans le but d'obtenir un prototype virtuel d'un microsystème MSMD. La deuxième étape consiste à identifier des outils et des techniques nécessaires à la mise en œuvre de cette méthodologie. Ainsi, dans ce chapitre, nous faisons un état des lieux des langages, et des environnements de simulation. Nous déterminons par la suite, l'ensemble des méthodes à utiliser pour la modélisation des sous-systèmes d'un microsystème MSMD à différents niveaux d'abstraction.

### **1. Définition des outils de modélisation**

Le développement des microsystèmes est étroitement lié aux outils de conception et à l'environnement de modélisation et de simulation. Cependant, cet environnement doit simplifier la construction et la vérification des modèles des composants d'un système complexe. Un tel développement est réalisé dans la CAO<sup>24</sup> un domaine (exemple de la CAO électrique). Cette approche permet au sein d'une équipe de concepteurs, la création de prototypes virtuels en utilisant des environnements adaptés et maîtrisés en fonction des besoins.

On note que les outils de modélisation au niveau physique (ex : modèles en éléments Finis), ne sont actuellement pas intégrés dans les environnements de modélisation au niveau système. Ceci est dû aux ressources informatiques qui restent limitées, et à la difficulté d'interconnecter et d'interfacer les modèles à différents niveaux d'abstraction.

Un environnement de développement est parfois associé à des outils d'optimisation et de synthèse logique. Parmi les projets qui ont été développés dans ce sens, nous citons : Moscito [Sch-02] et RUNE [Tis-04]. L'environnement Moscito, implémenté sous JAVA, est une plate-forme basée sur la combinaison de plusieurs simulateurs (ANSYS, Saber, ELDO et MATLAB), et supporte les algorithmes d'optimisation (des méthodes avec et sans dérivations et des approches stochastiques). Cet outil fournit une interface ouverte basée sur la communication par sockets TCP/IP pour intégrer des outils externes facilitant aussi le choix d'un flot de travail personnalisé.

RUNE constitue une plate-forme automatisée supportant la modélisation des systèmes multi domaines. RUNE est réalisée aussi en JAVA. Un ensemble d'outils et de modèles pour l'intégration des composants optoélectroniques et photoniques est disponible. Ceci est valable dans un flot de conception de semiconducteurs. La plate-forme comporte des passerelles avec d'autres simulateurs (Spectre, ELDO et MATLAB). L'approche de modélisation permet la hiérarchisation de modèles en SystemC au niveau système, VHDL-AMS et MATLAB au niveau comportemental, et FDTD<sup>25</sup> et Spectre au niveau physique.

---

<sup>24</sup> CAO : Conception Assistée par Ordinateur

<sup>25</sup> FDTD : Finite Difference Time Domain

De plus en plus, les environnements de modélisation doivent répondre aux besoins de la spécification des systèmes qui se traduisent par:

- La modélisation et la prise en charge de la compatibilité des liens entre les domaines dans un système hybride (les grandeurs, les analogies et les relations entre les entrées/sorties),
- L'automatisation des processus d'instanciation et de connexion des modèles,
- L'accessibilité des modèles décrits dans un langage ou dans un environnement aux concepteurs non spécialisés, d'où la création de bibliothèques de composants développés par plusieurs collaborateurs,
- La conformité du système aux unités physiques (considérant particulièrement les problèmes numériques présentés par différentes grandeurs),
- L'existence des règles pour la transformation des modèles d'un langage à un autre,
- La préférence d'une approche graphique,
- L'obtention de modèles simulables.

## **2. Modélisation collaborative**

La conception de systèmes multidisciplinaires nécessite la collaboration entre les groupes de spécialistes qui peuvent être géographiquement séparés. Pour supporter la modélisation collaborative, les équipes de conception ont besoin d'une représentation commune du système, des informations sur les composants de ce dernier, et d'une capacité d'abstraction de ses modèles.

Dans le flot de conception VLSI, le VHDL constitue un exemple type de langage permettant une conception numérique automatisée. Cependant, il n'existe pas d'environnements ou de langages qui supportent tout le flot de conception des microsystèmes MSMD (voir la section A.4). Pour ce cas de figure, une méthode de modélisation collaborative doit être définie entre tous les spécialistes et le concepteur système. Ceci peut être réalisé par la définition des techniques et des outils de simulation, afin de proposer des solutions permettant d'intégrer des modèles dans des domaines d'énergie à des niveaux d'abstraction différents. Ceci signifie, qu'il faut trouver des issues pour le développement et l'interaction des modèles qui sont décrits dans les différents domaines, et qui sont analysés à plusieurs niveaux d'abstraction. Dans une approche de modélisation collaborative, il est aussi important d'organiser et de sauvegarder les informations sur les concepts qui peuvent être réutilisées.

Récemment, plusieurs efforts de standardisation et d'unification ont eu lieu dans les langages de modélisation et dans le cadre des simulations multi-domaines (ex : VHDL-AMS et Simulink). Il est néanmoins très difficile d'intégrer les environnements uni domaine utilisés au niveau physique (ex : ANSYS, Synopsys) qui sont nécessaires dans la modélisation collaborative. Ce problème vient des formats de données et des solveurs appropriés à ces environnements uni domaine. Ainsi, les environnements de modélisation et de simulation doivent tenir compte des interactions entre les domaines d'application, et pouvoir adapter les différents solveurs.

### 3. Langages et environnements de modélisation

Plusieurs langages de modélisation et environnements ont été utilisés pour supporter les diverses phases de spécification, de conception architecturale, et de conception fonctionnelle. Dans cette partie du chapitre, nous explorons les langages et les outils les plus utilisés et ceux qui sont susceptibles d'être de bons candidats, pour répondre à nos objectifs de modélisation collaborative et de prototypage virtuel pour des microsystèmes MSMD.

#### 3.1 Langage C++

Le langage C++ constitue la base de plusieurs environnements de modélisation et de simulation. Ce langage orienté objet est un langage puissant et flexible, et possède plusieurs compilateurs et des outils d'aide. Le C++ supporte la programmation impérative des états d'un système. L'état d'un système inclut les valeurs des variables et les données objets qui peuvent être des variables, des équations, ou des fonctions. Le nombre d'objets change de façon dynamique.

Lors de la modélisation des microsystèmes, il est possible de construire des modèles au niveau système, grâce à la flexibilité des sémantiques et des fonctions existantes. Cependant, pour la modélisation au niveau composant, le standard C++ présente une incapacité de description directe du comportement des sous-systèmes dans les domaines d'énergie couplés. Il ne supporte pas la notion de temps, ni de synchronisation et de parallélisme entre les processus, ni d'interaction entre les niveaux d'abstraction. Le C++ ne possède pas de solveur mais permet l'implémentation d'algorithmes de résolution des équations différentielles et de construire des environnements de développement. Toutefois, les concepteurs ont généralement recours à des outils du marché pour développer et simuler leurs modèles.

#### 3.2 SystemC

SystemC représente une librairie ouverte basée sur C++. Il ne constitue pas un langage à part entière mais se repose sur les fonctionnalités du C/C++ en termes de syntaxe et compilation. Donc, SystemC associé au standard C++ peut être utilisé pour construire des modèles à différents niveaux d'abstraction. Ces modèles représentant le comportement d'un système, sont définis en SystemC avec des "*Module*" et des "*Process*", en plus de canaux, de ports et d'interfaces. SystemC supporte la description d'architectures hiérarchiques et la conception multi niveaux [Bou-06]. Cette approche est destinée à la conception des systèmes numériques matériels et logiciels.

Un microsystème fluide a été développé avec SystemC, et des modèles comportementaux discrets sont présentés [Zha-04]. L'architecture d'un environnement de modélisation et de simulation hiérarchique associé à un solveur des équations différentielles ordinaires et algébriques est développée. Une extension du SystemC aux signaux analogiques et mixtes et les MEMS, est d'ores et

déjà utilisée [Gri-04]. Cet outil de modélisation est proposé comme un candidat potentiel pour la conception des systèmes multi domaines, mais il reste au stade de l'exploration et n'est pas encore standardisé. L'inconvénient de SystemC (AMS<sup>26</sup>) est l'absence de solveurs qu'il faut implémenter pour la simulation du comportement d'un système.

### 3.3 VHDL-AMS et/ou Verilog-AMS

Ces dernières années, les langages VHDL et Verilog qui sont destinés à la description et à la synthèse des circuits numériques, ont vu leur extension évoluer pour supporter la description des systèmes analogiques et mixtes. Ces extensions pour des systèmes AMS sont référencées par VHDL-AMS et Verilog-AMS successivement. Verilog-AMS est un bon concurrent de VHDL-AMS. Les deux langages sont efficaces pour la modélisation au niveau comportemental et au niveau circuit. Une comparaison entre les deux langages est donnée dans [Pêc-05]. Le récapitulatif de cette comparaison est présenté en Annexe A.

En effet, VHDL-AMS permet la modélisation du comportement des systèmes multi domaines et hybrides avec des temps continus et/ou discrets (système analogique/numérique). Ce langage est devenu un standard IEEE (1071.1.1-2004). La modélisation avec VHDL-AMS permet de décrire les systèmes indépendamment des logiciels.

Parmi les avantages du langage, les lois de conservation de l'énergie sont supportées automatiquement par les solveurs des outils de simulation. Ce langage offre aussi la possibilité de créer des modèles comportementaux et structuraux, et de type work-flow<sup>27</sup>. En plus de ces avantages, VHDL-AMS prend en charge les équations implicites dont la variable à calculer n'a pas besoin d'être annotée explicitement. Ceci facilite l'application des lois de Kirchhoff généralisées (se référer à la section A.3.2.2). Il est possible aussi d'exprimer des modèles génériques dont les paramètres ont été définis lors de leur exécution effective [Sna-04].

Toutefois, l'inconvénient majeur de VHDL-AMS est son incapacité à modéliser les sous-systèmes au niveau physique (ex : PDE). Par ailleurs, les outils de simulation présentent des limitations qui sont dues aux instructions non définies dans la norme et non supportées par les simulateurs. En effet, ces limitations peuvent toujours être contournées. Dans certains simulateurs les bugs et les limitations ne sont pas commentés ce qui rend la tâche plus difficile [Her-02]. Des exemples de simulateurs pour l'implémentation des modèles en VHDL-AMS, ainsi que les limitations associées sont donnés dans le tableau ci-dessous :

---

<sup>26</sup> AMS:- de l'anglais **A**nalogue and **M**ixed **S**ystem

<sup>27</sup> Ensemble de blocs fonctionnels successifs dont les entrées n'ont pas d'influence sur les sorties



<i>ADVanced -MS 2005.3</i> (Mentor Graphics)	Ne supporte pas les instructions "PROCEDURAL", "QUANTITY" dans les ports (pour la modélisation en signal flow), "BREAK" évolué.
<i>SMASH 5.6.2</i> (DOLPHIN INTEGRATION)	Environnement graphique avec pré et post traitement, Message d'erreurs incompréhensibles lié à la compilation en C++, Ne supporte pas la philosophie de bibliothèque définie dans la norme.
<i>SIMPLOER V.7.0</i> (ANSOFT)	Supporte l'instruction "PROCEDURAL" et "BREAK" dans une forme évoluée, Ne supporte pas les quantités vectorielles (type ARRAY).

Tableau II-2 : Exemples de simulateurs pour VHDL-AMS

### 3.4 Modelica

Récemment, il y a eu d'importants efforts pour définir un autre langage, capable de modéliser des systèmes multidisciplinaires : Modelica [Modelica] [Elm-98] a été développé au début pour les systèmes continus, alors que VHDL-AMS est une amélioration du langage VHDL pour supporter les composants à signaux AMS.

Modelica est un langage orienté objet qui a été développé pour définir un langage unifié, qui peut supporter la modélisation des systèmes physiques. Contrairement aux langages de modélisation orientée objet (C++, JAVA), le modèle sous Modelica est une description mathématique déclarative, où l'analyse et la résolution des équations sont supportées. Il supporte aussi la modélisation non causale, qui signifie que les terminaux d'un modèle n'ont pas besoin d'être assignés comme *entrée* ou *sortie*. Les modèles Modelica peuvent être introduits en tant qu'équations différentielles (ODE ou DAE), des machines d'état ou des diagrammes blocs. Modelica supporte la modélisation des systèmes hybrides [Ott-99]. Modelica est supporté actuellement par Dynasim [Dynasim].

## 4. Comparaison entre VHDL-AMS et Modelica

Les deux langages supportent la modélisation hybride (temps continu et discret) pour les différents domaines d'énergie. Ils supportent aussi la modélisation déclarative et la hiérarchisation des sous-systèmes. Le langage Modelica a l'avantage d'être orienté objet avec un support pour le concept d'héritage et les sous-types (encapsulation). Il permet aussi la définition des connexions agrégées de façon à ce que les variables du flux et de l'effort et les différents signaux peuvent être combinés à travers les domaines d'énergie.

Le VHDL-AMS dispose de primitives de modélisation pour les simulations à événement discret et à temps continu avec la possibilité d'analyser les petits signaux AC et le bruit.

Cependant, la différence majeure entre les deux langages consiste dans les solveurs utilisés pour l'évaluation des modèles concernant la résolution des problèmes appelés d'index-3 [Asc-97]. Ces problèmes existent dans les systèmes mécaniques multi corps [Raj-01].

Pour l'instant VHDL-AMS, Verilog-AMS ou Modelica sont à présent une bonne base pour l'intégration des systèmes MSMD. Mais, ils ne sont pas suffisants pour un usage efficace des outils de simulation et d'une réutilisation flexible des modèles.

Ces langages de modélisation sont utilisés pour la conception fonctionnelle à plusieurs niveaux d'abstraction. Cependant, cette approche de conception nécessite une expertise humaine spécialiste. Elle mène également à des problèmes d'interprétation des spécifications du système, lors de la translation entre les données des modèles et les outils. Ainsi, il est bénéfique d'avoir un environnement de modélisation et de simulation hiérarchique, en utilisant un langage de description commun, et un moteur de simulation associé, plutôt que plusieurs langages et simulateurs qui traitent différents niveaux d'abstraction. L'intérêt de cette approche est de permettre la vérification et la validation du système, et de répondre aux exigences de la conception en termes de réduction de temps et de coût.

## 4.1 Simulateurs électriques

La modélisation des systèmes analogiques s'appuie aussi sur des environnements uni-domaine comme les représentations en circuits électriques. Les outils existants reposent sur des bibliothèques de composants qui permettent la réalisation de schémas équivalents, et sur le développement de modèles utilisateurs en analogie au monde électrique (voir la section A.3.2.2). Parmi ces environnements, nous citons : SPICE<sup>28</sup> [SPI], Saber [SAB], ELDO [ELDO] et Spectre [SPE]. SPICE a été développé à l'Université de Berkeley à partir de la fin des années 60. Depuis le simulateur est figé, et sa dernière version SPICE3F5 est distribuée librement sur internet. SPICE permet l'analyse statique, et transitoire des circuits non linéaires et l'analyse petits signaux des circuits linéarisés autour d'un point de fonctionnement. Il dispose d'une bibliothèque de composants passifs et actifs.

SPICE est basé sur la modélisation structurelle contrairement aux autres simulateurs électriques qui eux sont plutôt basés sur une description comportementale. Le fonctionnement de cet outil consiste en l'extraction des équations du réseau électrique en appliquant les lois de Kirchhoff. Il repose sur quatre méthodes numériques pour la résolution des équations :

- L'analyse nodale modifiée (MNA),
- La décomposition des matrices d'états en LU pour résoudre les systèmes linéaires,
- L'intégration trapézoïdale,
- L'analyse non linéaire de Newton-Raphson,

Cependant, la construction de nouveaux modèles sous SPICE s'avère difficile, du fait que le concepteur doit étudier tous les détails à ce niveau. Cet aspect peut diminuer la vitesse des simulations et aussi accroître le temps de mise sur le marché contrairement à la modélisation comportementale. Ceci rend la tâche du concepteur plus difficile. En plus, la taille du système d'équations dépend de façon exponentielle du nombre de nœuds ce qui ralentit les temps de simulation. SPICE présente aussi

---

<sup>28</sup> SPICE- de l'anglais **S**imulation **P**rogramm with **I**ntegrated **C**ircuit **E**mphasis

un handicap pour la modélisation des systèmes comportant des fonctions de contrôle et de traitement numérique. Il est donc impossible de modéliser les parties numériques et d'intégrer les IP.

## **5. L'environnement MATLAB/SIMULINK**

MATLAB/Simulink sont des produits de MathWorks Inc., et sont utilisés dans une centaine de pays à travers le monde, pour le développement des techniques de calcul et de la conception basée sur les modèles. MATLAB/Simulink est le quatrième environnement utilisé dans la conception électronique (après Cadence Design Systems, Synopsys et Mentor Graphics) et présente une satisfaction d'utilisation de 65% [EET-05]. Avec l'ensemble des produits associés et complémentaires à MATLAB/Simulink, plusieurs possibilités et services permettent de concevoir et aussi d'accélérer l'innovation dans la conception et le test des systèmes.

MATLAB est destiné essentiellement à réaliser des calculs mathématiques, la visualisation et l'analyse des résultats, et l'édition de nouveaux programmes utilisateurs. Simulink est l'extension graphique de MATLAB et sert à la modélisation et la simulation des systèmes dynamiques. Cet environnement graphique est une plate forme de simulation multi domaines basée sur les flots de signaux. Il contient un ensemble de bibliothèques qui peuvent être adaptées et/ou enrichies aux besoins du concepteur. Sous Simulink, le concepteur peut exécuter des simulations complètes, et intégrer des modèles de composants de nature différente (analogique, numérique) [Zen-04]. Il permet également d'implémenter des équations différentielles (ODE et DAE) et de développer des fonctionnalités destinées à l'implémentation matérielle (DSP, FPGA, ASIC...etc.) [Zen-05]. Les solveurs disponibles dans cet environnement, amènent à des simulations avec des temps continus (analogique), discrets (numérique), et mixtes. Le modèle est exécuté jusqu'à ce que les performances soient atteintes.

Les modèles basés sur des diagrammes blocs se composent de blocs et de connexions (signaux). La relation entre les éléments du diagramme bloc et les signaux qui les connectent décrit le système dynamique. Cependant, dans la bibliothèque de Simulink, on retrouve deux classes de blocs. La première classe représente les éléments constituant le système même (blocs non virtuels) et la deuxième classe concerne les blocs qui n'ont pas de rôle dans le système (blocs virtuels : gain, multiplexeur...etc.).

### **5.1 Processus de simulation sous Simulink**

Dans cette section, nous donnons une brève description sur le fonctionnement de la simulation sous Simulink et aussi la toolbox SimPowerSystems.

La simulation des systèmes dynamiques sous Simulink consiste à calculer les états du système et les sorties pour une plage de temps. Simulink fait appel au compilateur qui convertit le modèle dans une forme exécutable et procède comme suit:

- Evaluation des expressions des modèles des blocs pour déterminer leurs valeurs,

- Détermination des attributs des signaux (nom, type de données et dimension) qui ne sont pas explicitement spécifiés dans le modèle, et test si chaque bloc peut accepter les signaux qui lui sont connectés,
- Réalisation des optimisations spécifiées dans le panneau pour la configuration des paramètres,
- Mettre à plat le modèle hiérarchique en remplaçant les sous-systèmes virtuels qui les composent,
- Détermination de l'ordre d'exécution des méthodes des modèles des blocs (Outputs, Update, Derivative),
- Détermination des temps d'échantillonnage qui ne sont spécifiés explicitement de tous les blocs dans le modèle,
- Après cette phase, Simulink alloue la mémoire nécessaire à l'exécution du modèle (signaux, les états et les paramètres d'exécution).

## 5.2 Outils de cosimulation avec MATLAB/Simulink

L'évolution de MATLAB/Simulink a permis d'intégrer et d'interfacer d'autres outils : Real-Time Workshop, Link for ModelSim, Xilinx System Generator, AccelChip, Altera's DSP Builder, Symplify DSP, Femlab, et dernièrement Cadence a annoncé la réalisation de l'interface SLPS entre OrCAD/PSpice pour la cosimulation de systèmes électriques. Depuis quelques temps, Simulink dispose de bibliothèques pour les composants électriques (SimPowerSystems) et les composants mécaniques (SimMechanics). Les liens avec d'autres outils permettent d'étendre l'utilisation de l'environnement MATLAB/Simulink à la modélisation spécifique, et à des tâches de conception et de génération de code, d'implémentation matérielle, de test et de vérification des systèmes.

L'évolution des différentes versions de MATLAB/Simulink a fait l'objet d'une exploration de très près lors de nos travaux. La méthodologie de conception des systèmes MSMD telle qu'elle est suggérée dans la section A.7.3 a permis de désigner cet environnement pour la modélisation du comportement dynamique de ces systèmes.

Dans cette partie du document, nous présentons le mécanisme de la cosimulation avec MATLAB/Simulink. La cosimulation peut être multi-domaines, multi-niveaux, et multi-langages.

### • **SimPowerSystems**

SimPowerSystems est un ensemble de composants électriques et mécaniques qui vient s'ajouter à la bibliothèque de Simulink. Les composants ajoutés sont à la base des éléments destinés à des simulations de l'électronique de puissance. SimPowerSystems repose sur l'utilisation des ports électriques au lieu des signaux. Les ports électriques traitent des courants et des tensions, et le schéma obtenu est celui d'un circuit électrique. Le compilateur de SimPowerSystems génère le modèle de l'espace d'état du circuit pendant la phase d'initialisation [Sim-05].

Les simulations en utilisant SimPowerSystems peuvent être exécutées une fois que le circuit électrique est construit et se font comme dans n'importe quel autre modèle Simulink. Chaque fois qu'une simulation est lancée, une routine d'initialisation spéciale est appelée. Ce processus d'initialisation calcule le modèle espace d'état du circuit électrique et construit le système équivalent qui peut être simulé par Simulink. La commande "*power\_analyze*" fait partie du processus. Elle permet l'obtention du modèle espace d'état et de créer le modèle Simulink du circuit. On peut aussi appeler "*power\_analyze*" à partir des lignes de commande pour obtenir l'espace d'état de la partie linéaire du circuit. Lorsque le processus d'initialisation est lancé, la commande «*power\_analyze*» réalise les cinq étapes suivantes : organiser tous les blocs, ensuite obtenir les paramètres et évaluer la topologie du réseau et enfin séparer les blocs linéaires et non linéaires, et attribuer automatiquement à chaque nœud électrique un nombre. Une fois la topologie d'un circuit obtenue, le modèle espace d'état de la partie linéaire du circuit est calculé. Tous les régimes permanents et les initialisations sont réalisés à ce stade.

Afin de réaliser des modèles avec SimPowerSystems, il est recommandé d'utiliser des solveurs supportant les problèmes de transition raide<sup>29</sup>. Ainsi, "ode23tb" et "ode15s" sont les solveurs préconisés à ce fait. Le solveur "ode23tb" est classé comme moins précis mais converge plus rapidement, alors que le solveur "ode15s" converge plus lentement mais classé plus précis.

Les problèmes de raideurs interviennent lorsque la solution recherchée varie lentement, alors que des solutions proches varient plus rapidement. La méthode numérique doit employer des petits pas de simulation pour converger vers des résultats satisfaisants. Ceci implique que les solveurs "*stiff*" réalisent plus de calculs à chaque pas et permettent de donner des résultats contrairement aux autres solveurs disponibles [Mol-04]. Par ailleurs, la précision des simulations peut être ajustée grâce à la *tolérance relative* et à la *tolérance absolue* dans le menu des paramètres de simulation sous Simulink. Par conséquent, la réduction des valeurs des tolérances contribue à l'augmentation de la précision mais aussi le temps de simulation. On peut aussi ajuster la taille minimale et maximale du pas de simulation.

Initialement, ces paramètres ont des valeurs par défaut et c'est la simulation à pas variable qui est exécutée. Cependant, le temps de simulation est nettement plus lent d'où la nécessité de varier les paramètres de simulation.

- **Interfaçage du circuit électrique et Simulink**

Les éléments de Simulink et de SimPowerSystems peuvent être connectés mais doivent être interfacés. Les lignes connectant les blocs Simulink sont des signaux directionnels, alors que les lignes qui relient les terminaux ne sont pas directionnelles et ne peuvent pas être directement connectées aux lignes des blocs Simulink. La conversion des signaux Simulink en signaux électriques et vice versa nécessite l'utilisation de blocs d'interface entre les ports Simulink et les terminaux. Pour ce faire, le bloc de mesure (voltage measurement et current measurement) est utilisé pour interfacier les blocs

---

<sup>29</sup> En anglais "*stiff*"

SimPowerSystems et les blocs Simulink. Ce bloc convertit la tension ou le courant en signaux Simulink.

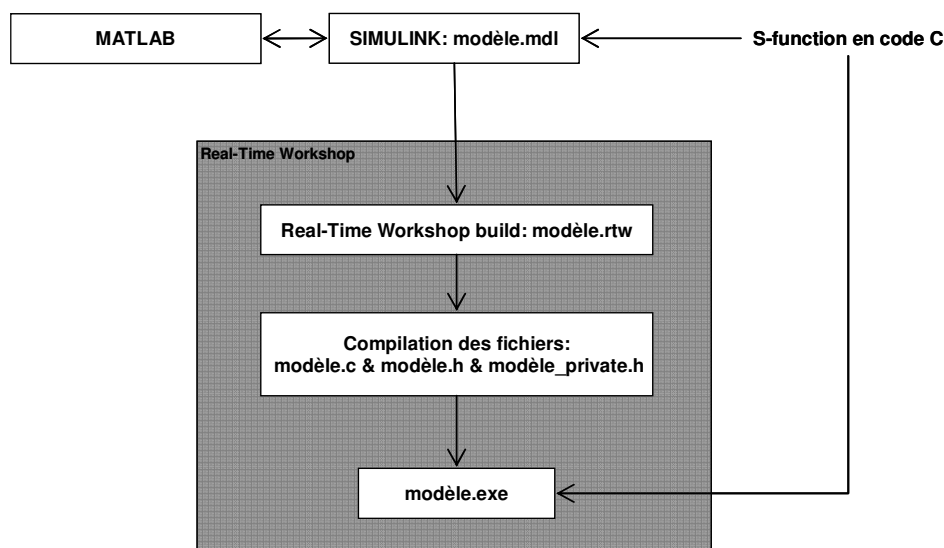
Les blocs Simulink sont aussi interfacés dans le système électrique grâce aux sources de tension et de courant contrôlées (controlled voltage source et controlled current source). Cette interface permet d'injecter un signal Simulink dans un circuit électrique. Nous donnons dans la section Chapitre III-A.2.3 un exemple d'interfaçage entre les blocs Simulink et les terminaux électriques.

- **S-funtion**

La fonction système permet de traduire un sous-système Simulink en S-function qui est constituée d'un ensemble de fichiers en C++. Le code généré pour cette S-function peut être réutilisé par la suite dans le cadre de la cosimulation multi-langages. Une S-function peut être construite à partir d'une description en langage informatique (fonction MATLAB ou fonction en C) et appelée sous forme d'un bloc Simulink. L'utilisation des S-function permet d'accélérer les simulations des systèmes.

- **Real Time Workshop**

Le Real Time Workshop est un outil très puissant dans Simulink. Le RTW génère du code C (ANSI) optimisé à partir de modèles Simulink. Il permet de créer et d'exécuter du code C et de tester aussi bien les sous-systèmes que le système entier. Le code résultant peut être réutilisé dans des applications à temps réel. Lors de l'exécution des modèles, cet outil est un moyen d'accélérer les simulations, et de réaliser le prototypage rapide pour les tests en boucle fermée. Le code généré peut être réutilisé sous forme d'un bloc Simulink, afin d'être analysé dans tout le système et peut être aussi exécuté en dehors de l'environnement MATLAB/Simulink. RTW permet de générer du code pour des systèmes continus, à événements discrets, et hybrides. La Figure II-11 présente le processus de génération d'un modèle (code) exécutable à partir d'un modèle Simulink (.mdl) ou d'une S-function en utilisant le RTW.



**Figure II-11 : Architecture de Real Time Workshop**

- **Link for ModelSim**

*Link for ModelSim* est une boîte à outils fournie par MathWorks dans l'environnement MATLAB/Simulink. Elle constitue une interface de cosimulation bidirectionnelle entre MATLAB/Simulink et le simulateur HDL ModelSim de Mentor Graphics® (voir la Figure II-12). La communication entre les deux environnements est réalisée par *sockets* ou par *mémoire partagée*. Avec cette interface, la modélisation des composants, le test, la vérification système, et l'analyse des résultats est réalisée au niveau des environnements (MATLAB/Simulink & ModelSim). La cosimulation est instantanée et continue pendant le temps de simulation. Cette interface permet l'intégration de code HDL automatiquement dans le système à modéliser, ce qui élimine les erreurs et le temps liés aux méthodes manuelles.

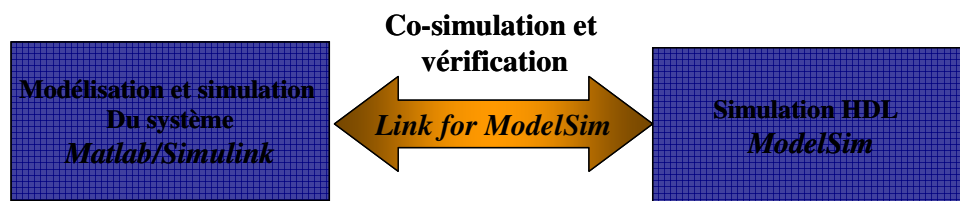


Figure II-12 : Interface Link for ModelSim/MATLAB/SIMULINK

- **Smash**

*Smash* est un logiciel développé par *Dolphin Integration* qui supporte la modélisation à signaux mixtes, multi niveaux et multi langages [SMASH]. Cet outil peut intégrer des modèles en VHDL, VHDL-AMS/Verilog-AMS et des netlist SPICE.

La cosimulation MATLAB/Simulink & *Smash* est basée sur une interface de type maître/esclave dont la communication est assurée via une S-function. Dans ce cas, MATLAB/Simulink joue le rôle du simulateur "maître" et fait appel à *Smash* à travers cette S-function exécutable. Durant les simulations, les noyaux des deux simulateurs sont synchronisés et partagent les données à travers les entrées/sorties.

## 6. Bilan

Nous avons préconisé, dans les parties précédentes, une méthodologie de conception collaborative afin de réaliser un prototype virtuel pour les microsystemes MSMD. Par la suite nous avons étudié les différents langages et environnements utilisés dans la conception des microsystemes. Dans ce paragraphe, nous déterminons notre choix de l'environnement MATLAB/Simulink pour mettre en œuvre notre approche de modélisation, de simulation et de vérification des microsystemes MSMD. Ce choix se justifie par la convivialité de l'environnement, par sa maturité en matière de pré et post traitement, les possibilités de développement des modèles dans différents domaines d'énergie, et par la cosimulation multi-langages. Ainsi, l'évolution de l'environnement MATLAB/ Simulink permet

de concilier l'efficacité des approches de conception hiérarchiques et l'intégration de plus en plus d'outils de développements pour la vérification à travers les niveaux d'abstraction. Notre intérêt pour le prototypage sous MATLAB/Simulink est aussi lié aux techniques de modélisations des microsystèmes que nous allons développer dans la suite du document pour les raisons suivantes:

- modélisation avec des diagrammes bloc. Cette technique permet la construction pertinente et l'utilisation facile des modèles. Elle supporte la modélisation hiérarchique, la réutilisation et l'évolution vers des modèles plus complexes couvrant différents domaines et niveaux d'abstraction,
- modélisation basée sur les équations et les descriptions mathématiques,
- modélisation basée sur les lois de conservation de l'énergie (ex : lois de Kirchhoff),
- réutilisation des modèles, qui peuvent être adaptés à d'autres applications. Cette généralisation permet des modèles simples et une simulation plus efficace,
- Cosimulation avec d'autres outils.

Ceci n'empêche pas que MATLAB/Simulink présente des inconvénients en comparaison à VHDL-AMS qui constitue un candidat potentiel pour la modélisation des microsystèmes MSMD. Ses faiblesses concernent les variables qui doivent être définies explicitement comme *entrées/sorties* lors de la réalisation d'un modèle. Le problème ne se pose pas lorsque la boîte à outil SimPowerSystems est utilisée où les signaux (tension et courant) sont définis implicitement.

L'exécution des modèles sous MATLAB/Simulink peut être affectée par des temps de simulations lents dû au fait que c'est un langage interprété.

Cependant, la comparaison réalisée dans [Oud-04] entre la modélisation VHDL-AMS basée sur l'utilisation des lois de la conservation de l'énergie et MATLAB/Simulink basé sur le flot de signal (signal flow) a été réalisé pour une version antérieure de MATLAB/Simulink (version 6.0 et 4.0 respectivement). L'inconvénient relaté concerne l'utilisation des réseaux de Kirchhoff pour résoudre les problèmes d'interaction entre les blocs constituant le système. Ce problème est dû à l'échange bidirectionnel des énergies. Cependant, l'évolution de MATLAB/Simulink permet de réaliser des interconnexions bidirectionnelles en définissant des interfaces entre les modèles. Le modèle fonctionnel du microsystème réalisé permet l'interaction entre toutes les parties de ce dernier [Zen-05].

## 7. Conclusion

Dans ce chapitre, nous avons défini les points essentiels pour la conception des microsystèmes. Tout au long de ce document, le microsystème fait référence à une méthodologie de conception, où la tâche primordiale du concepteur est d'implémenter un certain nombre de composants.

La modélisation et la simulation permettent de valider les spécifications d'un système donné avant le prototypage physique. Finalement, un modèle consistant d'un microsystème doit se rapprocher le plus possible du système réel, et supporter la réutilisation de modèles et d'environnements collaboratifs.



Nous avons élaboré une synthèse des techniques et les langages et les environnements qui vont nous permettre de modéliser un microsystème MSMD, en se basant sur une description avec les diagrammes blocs. Cette technique permet de décrire des modèles analytiques et d'introduire des équations différentielles, de développer des sous-systèmes à différents niveaux d'abstraction, et des processus de contrôle en boucle fermée.

L'environnement MATLAB/Simulink a retenu notre intérêt de part la maturité et la richesse de ses outils de développement et vérification. Cet environnement nous permettra de concevoir notre prototype virtuel d'un microsystème pour une première validation. Il supporte la modélisation déclarative et impérative et la modélisation orientée objet. Par ailleurs, la simulation uni domaine ou par analogie entre les domaines, et la simulation discrète et continue sont aussi supportées. Il est très possible avec cet environnement d'interagir avec les différents collaborateurs avec la possibilité de raffiner les sous-systèmes et de créer plusieurs modèles documentés et d'exécuter d'autres simulateurs en utilisant la cosimulation.

Niveaux d'abstraction	Description	VHDL-AMS	MATLAB/SIMULINK
Modélisation système	<ul style="list-style-type: none"> <li>✓ Comportement des composants</li> <li>✓ Fonctionnement du système</li> <li>✓ Simulations en mode mixte</li> </ul>	<ul style="list-style-type: none"> <li>✓ Réseaux de Kirchhoff</li> <li>✓ VHDL</li> <li>✓ Plusieurs environnements de simulation</li> <li>✓ Langage émergeant</li> </ul>	<ul style="list-style-type: none"> <li>✓ Systèmes: électrique, mécanique, thermique, contrôle</li> <li>✓ Résolution des lois des mailles et les lois des nœuds dans l'espace d'état (matrice d'état)</li> <li>✓ Diagrammes de blocs</li> <li>✓ Algorithmes et machines d'état</li> <li>✓ Cosimulation avec VHDL</li> </ul>
		<ul style="list-style-type: none"> <li>✓ Description comportementale, flot de signaux, flot de données, équations constitutives</li> <li>✓ Signaux continus/ discrets</li> <li>✓ Simulation temporelle, analyse fréquentielle, bruit</li> </ul>	
Modélisation circuit (conservative)	<ul style="list-style-type: none"> <li>✓ Basée sur la conservation de l'énergie, macro modèles</li> <li>✓ Analyse temporelle et stationnaire</li> <li>✓ Équations différentielles (ordinaires/ algébriques)</li> <li>✓ Analyse du couplage entre les composants</li> </ul>	Équation différentielles implicite/explicite Paquetage (électrique, mécanique)	Description comportementale et modélisation mathématique indépendamment du domaine
Modélisation physique	<ul style="list-style-type: none"> <li>✓ Influence des technologies utilisées</li> <li>✓ Propriétés des matériaux, géométrie</li> <li>✓ Équations différentielles Partielles</li> <li>✓ Synthèse logique</li> </ul>	Modèles d'ordre réduit	<ul style="list-style-type: none"> <li>✓ Outil PDE, cosimulation FEMLAB</li> <li>✓ Outils d'implémentation des DSP et FPGA (Xilinx System Generator, Altera DSP Builder)</li> </ul>
		Inconvénients: <ul style="list-style-type: none"> <li>✗ Pas de simulateur spécifique</li> <li>✗ La dérivation dans l'espace non disponible</li> <li>✗ Non portabilité de modèles à l'extérieur du langage (ex: les modèles C/C++)</li> <li>✗ Certaines instructions ne sont pas implémentés dû aux simulateurs (PROCEDURAL)</li> </ul>	Inconvénients: <ul style="list-style-type: none"> <li>✗ Langage interprété: temps d'analyse réduit &amp; temps de simulation augmentant</li> <li>✗ Passerelle directe entre les niveaux d'abstraction non disponible</li> </ul>



# Chapitre III- Modélisation du microsysteme autonome

---

A. Modélisation d'un microsysteme autonome .....	44
1. Présentation du microsysteme autonome .....	44
2. Générateurs piézoélectriques.....	46
2.1 Modèle générique de conversion des vibrations mécaniques en électricité.....	48
2.2 Modélisation du générateur de puissance.....	49
2.3 Construction d'une bibliothèque de générateurs piézoélectriques .....	71
2.4 Bilan .....	72
3. Modélisation du circuit de récolte d'énergie.....	74
3.1 Redresseur AC/DC.....	75
3.2 Convertisseur DC/DC .....	80
3.3 Génération de la commande du commutateur.....	93
3.4 Bilan .....	98
4. Elément de stockage : la batterie.....	100
4.1 Modèle comportemental dynamique .....	100
4.2 Modèle raffiné de la batterie .....	109
4.3 Bilan .....	115
5. Conclusion.....	116
B. Stratégie de la simulation .....	117
1. Simulation globale.....	117
1.1 Premier prototype virtuel .....	118
1.2 Exemple du prototype virtuel avec plusieurs niveaux d'abstraction .....	122
2. Cosimulation .....	124
3. Comparaison des temps de simulation .....	127
4. Validation expérimentale .....	129
5. Conclusion.....	130

Dans ce chapitre, nous appliquons la méthode de modélisation et les techniques détaillées précédemment au microsystème autonome (SPMS<sup>30</sup>) du projet VIBES (voir la section Chapitre I-A.2.2). Nous développons les différentes architectures pour la partie qui est relative à la récupération d'énergie. Ensuite, nous implémentons ces architectures dans l'environnement de simulation préconisé. Enfin, nous définissons la stratégie de simulation pour le prototype virtuel du microsystème autonome.

## A. Modélisation d'un microsystème autonome

### 1. Présentation du microsystème autonome

Le microsystème autonome est décomposé en trois domaines hétérogènes : la partie MEMS, la partie électronique analogique-numérique et la partie de stockage de l'énergie qui est une batterie électrochimique. Ces parties sont constituées de sous-systèmes. Cette décomposition permet de réaliser la modélisation hiérarchique des sous-systèmes comme le montre la Figure III-1. Le sous-système de génération de puissance sera un MEMS à base de matériau piézoélectrique. Le circuit de récolte d'énergie (EHC<sup>31</sup>) est composé de sous-systèmes analogiques et numériques. Le dernier sous-système est le réservoir d'énergie qui peut être une batterie ou éventuellement une super capacité. Ces sous-systèmes traitent de différents domaines physiques et des signaux de diverses natures (analogique et numérique) et sont issus de technologies différentes.

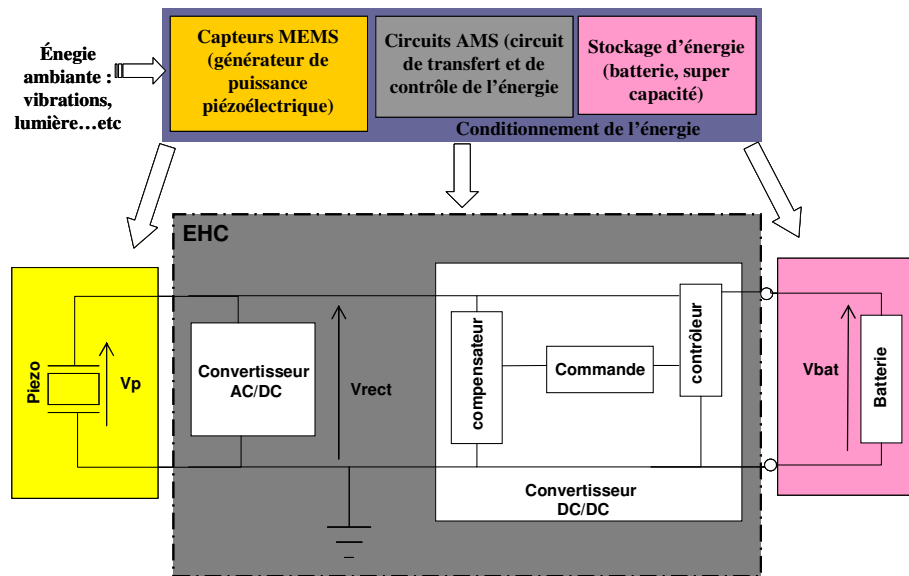


Figure III-1 : Schéma du microsystème autonome

<sup>30</sup> SPMS :- pour Self Powered MicroSystem

<sup>31</sup> EHC :- pour Energy Harvesting Circuit

Le générateur de puissance produit une tension alternative due aux vibrations mécaniques présentées dans l'environnement du capteur (ex : machine, outil,...etc.). Cette tension est acheminée vers la batterie via le circuit de récolte de l'énergie (EHC). Ce dernier se compose d'un redresseur AC/DC, et d'un convertisseur DC/DC. Le circuit DC/DC joue le rôle de générateur de courant continu pour la batterie, et se compose d'un compensateur commandé par une boucle de contrôle. La boucle de contrôle est une fonction implémentée pour générer un signal numérique destiné à l'activation du compensateur. Lors de la description du module DC/DC, la partie contrôle est définie comme un sous-système et est distincte de ce dernier. Les étages AC/DC et DC/DC sont vus comme une impédance par le générateur et sont utilisés pour la détermination et le transfert de la puissance maximale du générateur vers la batterie [Ott-02]. La batterie, dans ce cas, est utilisée pour le stockage de l'énergie afin d'alimenter les divers composants du nœud sans fil (microcontrôleur, module RF et capteur).

Une fois le microsysteme partitionné en sous-systèmes comme nous l'indiquons sur la Figure III-2, ces derniers sont modélisés individuellement. Les sous-systèmes sont décrits à différents niveaux d'abstraction. Les détails des descriptions peuvent émaner soit du raffinement des modèles des sous-systèmes au niveau local, soit des interactions dans le cadre de la modélisation collaborative où les sous-systèmes sont caractérisés individuellement par les experts (voir la section Chapitre II-A.7.3).

### *Modélisation multi domaine à signaux mixtes*

Sous-systèmes	MEMS (Piézoélectrique)		circuits analogiques (AC/DC & DC/DC)		circuit numérique (algorithme de contrôle)		Batterie	
Nature des signaux	Continu		Continu		Discret		Continu	
Description	Circuit électrique équivalent	Analytique -physique de la piézo- électricité-	Circuit électrique -composants électriques-	Analytique -lois de Kirchhoff-	Algorithme	VHDL	Circuit électrique équivalent	Analytique -expressions & LUT-
	Interface d'adaptation		Interface d'adaptation		Interface d'adaptation		Interface d'adaptation	

**Figure III-2 : Partitionnement du microsysteme autonome selon les domaines et la nature des signaux**

Les différentes descriptions des sous-systèmes à différents niveaux d'abstraction peuvent être confondues par la suite et rassemblées pour constituer le prototype virtuel qui représente le modèle fonctionnel du microsysteme. Ceci permet d'analyser et de valider le comportement global du microsysteme. La diversité des domaines et des signaux impliqués impose l'utilisation de modules d'interfaces. Ces interfaces assurent la communication entre les sous-systèmes par échange d'un ensemble

de paramètres et de signaux [Nec-02]. Selon le type de description, l'interface peut être directe ou indirecte (se référer à la section Modélisation comportementale basée sur le circuit électrique équivalent Chapitre II-A.3.2.2).

L'approche de modélisation et de simulation adoptée est réalisée sous l'environnement MATLAB/Simulink en exploitant les différents outils notamment la SimPowerSystems toolbox. Cette décision constitue un choix et un défi relevé dans le cadre de cette thèse en vue de profiter des possibilités offertes par cet environnement. Ces possibilités concernent tout d'abord, les différentes techniques que supporte l'environnement pour la modélisation tel que les équations différentielles, les représentations physiques de circuits électriques (avec SimPowerSystems), et la construction de programmes et de fonctions numériques. L'environnement offre aussi la possibilité de réaliser des simulations à plusieurs niveaux d'abstraction et possède des solveurs pour supporter les signaux mixtes pour les systèmes en boucle (résolution des équations, des non-linéarités, des discontinuités...etc.). Enfin, l'environnement dispose d'outils de calcul, de supervision et d'analyse permettant de concevoir un prototype virtuel pour le microsysteme autonome. Finalement, la validation du prototype virtuel, est supportée par l'environnement selon deux stratégies que nous détaillerons plus loin dans le manuscrit (voir la section du Chapitre III-B). La première stratégie est basée sur la validation par simulation globale. La deuxième stratégie est basée sur la validation par cosimulation multi-langages et multi-niveaux.

## 2. Générateurs piézoélectriques

Dans cette partie du document, nous décrivons le sous-système de génération de puissance qui constitue un élément essentiel dans le microsysteme sans fil. Les modèles de ce sous-système sont réalisés en collaboration avec M. Marzencki qui a étudié les microgénérateurs piézoélectriques pendant sa thèse [Mar-07].

Trois principaux types de générateurs existent et ont fait l'objet de travaux de recherche. Les générateurs électromagnétiques permettent d'obtenir des puissances intéressantes, mais ce sont des dispositifs macroscopiques, puisque leur miniaturisation et leur intégration sont toujours des défis à relever [She-97] [Lee-03] [Gly-04] [Ser-06] [Spr-06] [Bee-06] [O'Do-06]. Les générateurs électrostatiques sont des dispositifs microscopiques aussi et nécessitent des techniques d'implémentations ardues [Men-01] [Ste-03] [Ste-03] [Des-05] [Pea-05] [Chi-06]. La dernière catégorie concerne les générateurs piézoélectriques qui constituent une alternative pour l'intégration microscopique pour ce type de dispositifs. Les générateurs piézoélectriques peuvent produire des puissances relativement élevées pour des structures de grandes dimensions [Rou-03] ou excitées avec des amplitudes d'accélération élevées [Mar-07].

Dans cette section, nous nous focalisons sur la modélisation des générateurs piézoélectriques. De plus en plus, des microsystèmes à base de matériaux piézoélectriques en couches minces sont fabriqués [Guo-05]. Cet intérêt est dû à l'amélioration des propriétés des couches minces piézoélectriques [Sem-01] [Lan-02]. Nous rappelons que le principe de l'effet piézoélectrique direct est la génération de charges électriques sous l'effet d'une contrainte mécanique. Inversement, la structure se déforme sous l'effet d'un champ électrique. Ce phénomène est dû à la formation d'un dipôle lors de la séparation des charges dans les structures cristallines (voir Figure III-3).

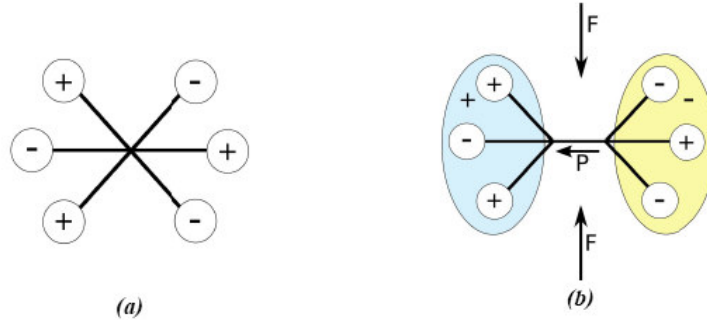


Figure III-3 : Principe de la piézoélectricité

Nous nous sommes intéressés à des structures simples, dites bimorphes, formées d'un substrat et de couches de matériau piézoélectrique [Gly-01]. Dans nos modèles, nous avons utilisé les propriétés des matériaux de type PZT<sup>32</sup> qui présentent des propriétés électromécaniques très intéressantes (couplage et pertes) et qui peuvent être déposées en couche minces [Lan-02]. Pour ces structures bimorphes, deux modes de fonctionnement sont définis. Les deux modes dépendent des directions de la tension et de la force générées qui peuvent être perpendiculaires (mode 31) ou parallèles (mode 33) comme illustré sur la Figure III-4. En général, le mode utilisé pour les structures en flexion est le mode 31 à cause de la faible épaisseur des couches PZT mise en œuvre.

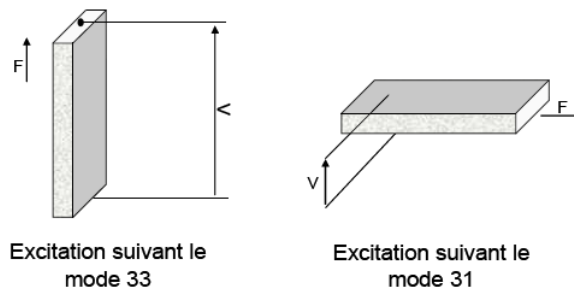


Figure III-4 : Modes d'excitation d'un barreau piézoélectrique

<sup>32</sup>PZT : Titanate Zirconate de Plomb



Sur la Figure III-5 nous donnons plusieurs exemples de structures de type bimorphe dont les détails du développement analytique sont présentés dans les travaux de Smits [Smi-94], Brissaud [Bri-04] et Roundy [Rou-03]. Nous précisons que nos travaux ont porté sur la quatrième structure. Cette structure est constituée d'une cale entre deux couches de PZT avec une masse à l'extrémité de la poutre. Les modèles pour cette structure sont développés à partir de la section 2.2.2 de ce chapitre.

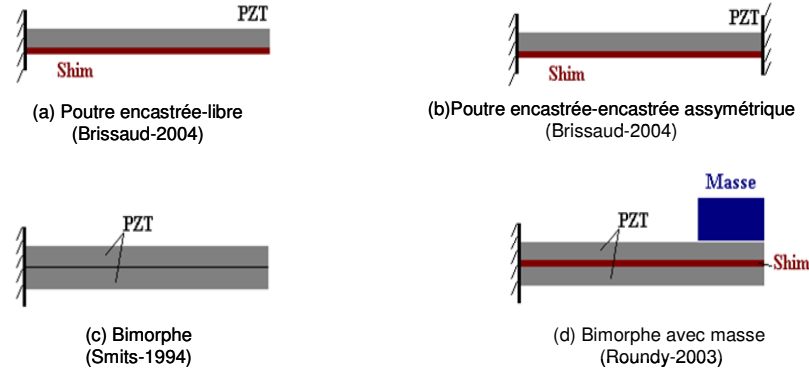


Figure III-5 : Différentes structures de type bimorphe

## 2.1 Modèle générique de conversion des vibrations mécaniques en électricité

Cette section est un bref rappel sur le mécanisme de conversion des vibrations mécaniques en une énergie électrique. Le modèle de conversion est proposé par le modèle de Williams et Yates [Wil-95]. Ce modèle générique à une dimension simplifié est constitué d'une masse connectée à un ressort confiné dans une cage rigide. Lorsque la cage est soumise à des vibrations, la masse se déplace et l'énergie est restituée dans le système masse-ressort. Pour ce modèle, la conversion des vibrations en énergie électrique est vue comme un amortissement linéaire par le système. Malgré le fait que le modèle reste indépendant des méthodes de conversion des oscillations en un signal électrique par transduction électromagnétique, électrostatique, ou piézoélectrique, quelques modifications se sont imposées pour les transducteurs électrostatiques et piézoélectriques [Rou-03].

Pour ces types de transducteurs, l'effet du système électrique sur le système mécanique est non linéaire mais aussi non proportionnel à la vitesse. La perte de l'énergie cinétique dans le système est représentée par l'effet de l'amortissement dans le modèle. Cet amortissement se traduit par l'ensemble des pertes mécaniques et de l'énergie électrique créée, représenté respectivement par les facteurs  $b_m$  et  $b_e$  sur la Figure III-6. L'équation différentielle de ce modèle générique est donnée ci-dessous :

$$m\ddot{z} + (b_e + b_m)\dot{z} + kz = -m\ddot{y}$$

III-1

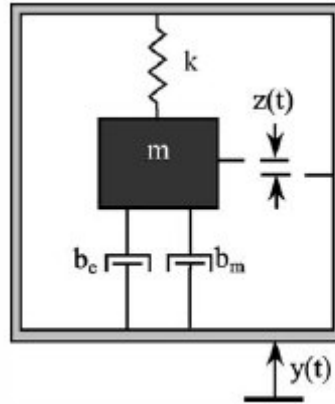


Figure III-6 : Schéma fonctionnel d'un transducteur des vibrations en électricité

où  $z(t)$  est le déplacement de la masse mesuré dans le référentiel de la cage et du ressort dû au déplacement de cette dernière selon  $y$ , avec  $m$  la masse de la charge,  $b_e$  est le coefficient d'amortissement électrique,  $b_m$  est le coefficient d'amortissement mécanique, et  $k$  la raideur du ressort.

La puissance électrique est la puissance engendrée par l'amortissement du système mécanique. Cette puissance est proportionnelle au coefficient d'amortissement électrique  $b_e$  et au carré de la vitesse de déplacement du système masse-ressort  $\dot{z}$ .

$$P = \frac{1}{2} b_e \dot{z}^2$$

III-2

## 2.2 Modélisation du générateur de puissance

Lors du partitionnement du microsystème multi domaines, le générateur de puissance utilisant la piézoélectricité constitue le sous-système de type MEMS. La puissance électrique issue du générateur piézoélectrique est récoltée dans la batterie qui est la source d'alimentation du nœud sans fil. Pour ce type de système, la quantité de puissance dépend de la charge connectée aux bornes du générateur lorsque le couplage est fort.

Du fait qu'à l'entrée du transducteur, l'énergie est mécanique, et qu'à la sortie, l'énergie est électrique; les modèles **doivent supporter l'échange entre ces domaines d'énergie**. L'autre aspect est de considérer l'impact bidirectionnel avec le reste des sous-systèmes. Nous avons donc étudié le dispositif afin de proposer différents modèles. Ces modèles sont intégrés dans le modèle global du système pour vérifier le fonctionnement du prototype virtuel.

Pour modéliser le générateur de puissance piézoélectrique, nous nous sommes basés sur les techniques développées dans le chapitre précédent. Dans un premier temps, nous avons utilisé le modèle électrique équivalent présenté par Ottman [Ott-02]. Ensuite, à partir des équations constitutives d'une structure bimorphe piézoélectrique [Rou-03], nous avons développé un modèle analytique du générateur en prenant en compte les caractéristiques géométriques et physiques. Ensuite, nous avons, construit le modèle du générateur développé par M. Marzencki dans sa thèse [Mar-07]. Enfin, nous avons introduit le modèle au niveau d'abstraction physique issu de la simulation en éléments finis sous ANSYS.

### 2.2.1 Modèle en circuit électrique équivalent

Le comportement du générateur de puissance est décrit par les relations entre le courant et la tension. A ce niveau d'abstraction, les détails physiques ne sont pas considérés, et la nature des vibrations (amplitude et fréquence) est incluse dans la formule du courant produit par le transducteur. Lorsque le générateur piézoélectrique est soumis aux vibrations, des charges électriques se forment sur les surfaces du matériau pour charger une capacité  $C_{piezo}$ . La tension fournie par le générateur est capturée aux bornes de ces électrodes.

Ainsi au niveau comportemental, le transducteur piézoélectrique est un modèle unidimensionnel qui peut être décrit par son signal de sortie  $i_p(t)$  donné par l'équation III-3.

$$i_p(t) = I_p \sin(\omega t) \quad \text{III-3}$$

Le courant  $i_p(t)$  est un signal sinusoïdal dont l'amplitude dépend de l'excitation mécanique  $\ddot{y}(t)$  de la structure piézoélectrique mais aussi de sa fréquence de résonance. Les valeurs du courant, de la capacité et de la fréquence peuvent être obtenues à partir des mesures expérimentales [Ott-02].

La Figure III-7 montre le modèle du générateur de puissance sous la forme d'un circuit électrique équivalent. Un tel modèle décrit la sortie du générateur en circuit ouvert qui est représentée par la source de courant et la tension qui est mesurée aux bornes de la capacité.

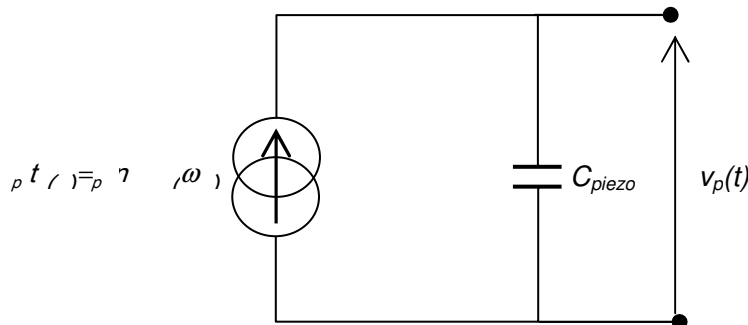


Figure III-7 : Circuit électrique équivalent du transducteur en circuit ouvert

Le modèle ci-dessus peut être amélioré en introduisant les pertes diélectriques représentées par une résistance équivalente  $R_{piezo}$  en parallèle avec la capacité  $C_{piezo}$ .

Afin de produire une puissance électrique, le générateur piézoélectrique est connecté à une charge. La puissance générée est maximale pour  $\omega=\omega_p$  (où  $\omega_p$  est la pulsation de résonance) et une charge résistive optimale donnée par l'expression ci-dessous.

$$R = \frac{1}{\omega_p C_{piezo}} \quad III-4$$

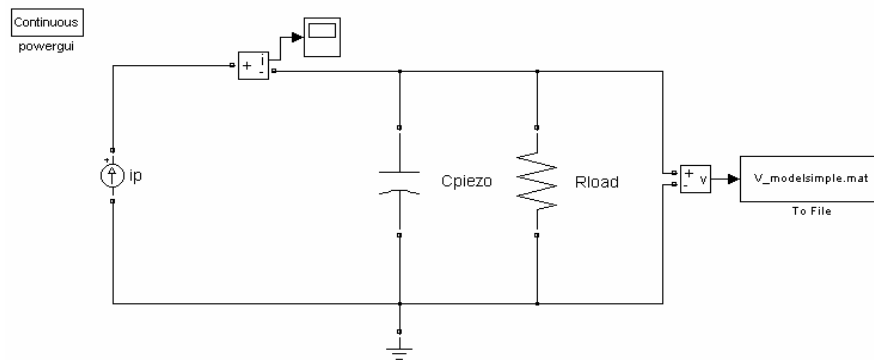
Ainsi, le courant fourni par le générateur est subdivisé en deux parties. Une partie va dans la capacité et l'autre partie dans la résistance de charge connectée à ces bornes, d'où l'expression ci-dessous.

$$i_p(t) = C_{piezo} \frac{\partial v_p}{\partial t} + \frac{v_p}{R} \quad III-5$$

Nous pouvons noter que la représentation par une source de courant fait que le modèle est correct lorsque le **couplage entre le générateur et la charge électrique est faible**. Ainsi, le modèle au niveau comportemental présente des limitations, d'où l'intérêt de développer un modèle qui tient compte de l'impact de la charge électrique sur la partie mécanique du modèle.

### • Implémentation du modèle en circuit électrique équivalent

Ce modèle étant simple, il peut facilement être implémenté sous la forme de circuit électrique en utilisant la boîte à outils SimPowerSystems sous Simulink (voir la Figure II-8). Cette boîte à outils nous permet de construire des circuits électriques en utilisant les éléments prédéfinis dans la librairie.



**Figure III-8 : Implémentation du modèle comportemental sous Simulink à l'aide de la boîte à outil SimPowerSystems**

Dans ce cas, SimPowerSystems est utilisée pour remplacer les simulations qui peuvent être exécutées dans un simulateur électrique tel que SPICE, et par ailleurs construire des modèles par analogie ou simuler des circuits. Cependant, cet outil disposant d'une librairie de composants électriques

génériques, le concepteur doit introduire les divers paramètres déclarés dans les modèles relatifs aux composants choisis (diodes, transistors...etc.).

Nous avons vérifié le modèle au niveau comportemental du générateur de puissance en utilisant ce circuit électrique équivalent. Pour ce faire, nous supposons qu'un courant sinusoïdal  $i_p(t)$  d'une amplitude de  $660\mu A$  et une fréquence de  $120Hz$  est fourni par une structure piézoélectrique. La capacité mesurée est  $C_{piezo}=2,07nF$  et une résistance  $R=200k\Omega$  est connectée aux bornes du générateur [Amm-07]. Nous avons observé la variation de la tension de sortie du générateur piézoélectrique en fonction du temps. En régime permanent, la tension  $V_p$  maximale est égale à  $12.6V$  (voir la Figure III-9). Nous avons obtenu le même résultat en implémentant la fonction de transfert du dispositif sous Simulink.

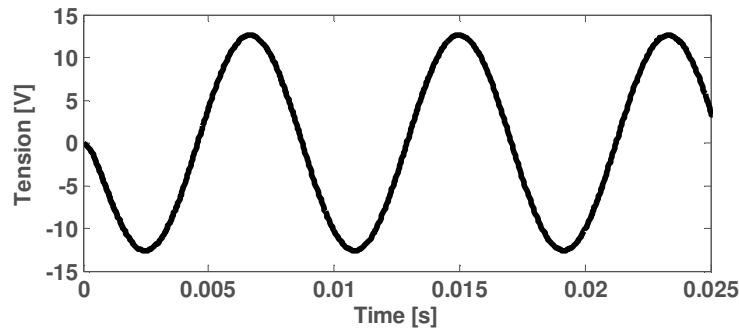


Figure III-9: Tension fournie par le générateur aux bornes de la résistance optimale

Ainsi pour cette configuration, la puissance maximale récupérée selon l'équation ci-dessous est  $397\mu W$ .

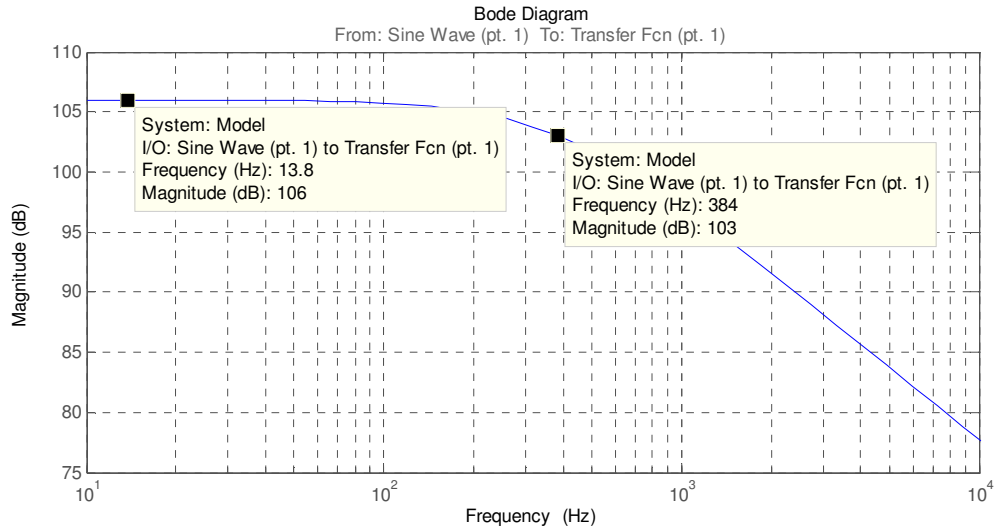
$$P_{moy} = \frac{V_p^2}{2R_{load}} \quad III-6$$

### • Fonction de transfert du générateur piézoélectrique

Nous pouvons donc déduire la fonction de transfert du dispositif à partir de l'équation III-5:

$$H(s) = \frac{V_p}{I_p} = \frac{R}{RC_{piezo}s + 1} \quad III-7$$

Nous pouvons déterminer la fréquence de coupure du générateur piézoélectrique et la charge en utilisant l'outil d'analyse linéaire *LTI Viewer* sous Simulink. Dans ce cas, la fréquence de coupure à  $-3dB$  est  $f_c=384Hz$  (voir la Figure III-10).



**Figure III-10 : Réponse de la fonction de transfert du modèle comportemental**

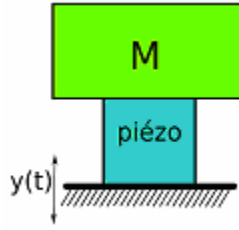
L'avantage de la description du générateur à ce niveau avec un circuit électrique équivalent est que le modèle peut être facilement implémenté et intégré avec d'autres éléments électriques dans un environnement de modélisation en s'appuyant sur la loi de la conservation de l'énergie. Cependant, la modélisation du générateur de puissance à ce niveau d'abstraction n'est pas toujours représentative de la réalité compte tenu des hypothèses simplificatrices qui ne tiennent pas compte par exemple du couplage avec le reste du système ni des divers types de pertes diélectrique, structurale et visqueuse présentes dans un tel composant. En effet, le courant est mesuré, puis maintenu fixe tout au long de la vérification. Dans la section qui suit, nous introduisons plus de détails qui permettent de raffiner ce générateur de puissance. Le modèle est basé sur une description mathématique qui prend en compte les caractéristiques physiques et dimensionnelles du générateur.

### 2.2.2 Modèles fonctionnels basés sur une approche analytique

Une approche raffinée du modèle électrique consiste à étendre l'étude à une description qui se rapproche le plus du modèle physique et à une plus large plage de fonctionnement. En raffinant le modèle, l'impact des caractéristiques physiques et géométriques sur le comportement du générateur sont considérées. La prise en compte de ces détails constitue un nouveau niveau d'abstraction.

Dans ce cas, la construction des modèles se fait par description analytique en utilisant les équations différentielles ordinaires (ODE), en développant les relations entre les variables du système à modéliser. Les variables dépendent naturellement de leurs variations dans le temps en fonction des structures et des matériaux impliqués.

Pour construire le modèle analytique, les équations différentielles du système sont développées pour la structure unidimensionnelle de la Figure III-11 à partir des équations constitutives de la piézoélectricité qui sont données ci-dessous.



**Figure III-11 : Structure simple à modéliser**

$$\delta = \frac{\sigma}{Y} + dE$$

$$D = d\sigma + \epsilon E$$

III-8

$\delta$  : La déformation dans la direction de l'application de la force.

$\sigma$  : La contrainte mécanique appliquée ( $\text{N.m}^{-2}$ ).

$E$  : Le champ électrique ( $\text{V.m}^{-1}$ ).

$D$  : Le déplacement électrique ( $\text{C.m}^{-2}$ ).

$Y$  : Le module d'élasticité ( $\text{N.m}^{-2}$ ).

$d$  : Le coefficient de couplage piézoélectrique ( $\text{C.N}^{-1}$ ).

$\epsilon$  : La constante diélectrique du matériau piézoélectrique ( $\text{F.m}^{-1}$ ).

La structure pour laquelle nous avons construit le modèle analytique est une poutre bimorphe simple. Une vue plus détaillée de la structure est donnée par la Figure III-12. La structure est composée de deux couches de matériau de type PZT symétriques et d'une cale intermédiaire en laiton. L'extrémité libre de la poutre supporte une masse et l'autre extrémité est encastree. Dans nos travaux, nous nous sommes basés sur la structure schématisée par la Figure III-12 qui a été étudiée par S. Roundy dans sa thèse [Rou-04].

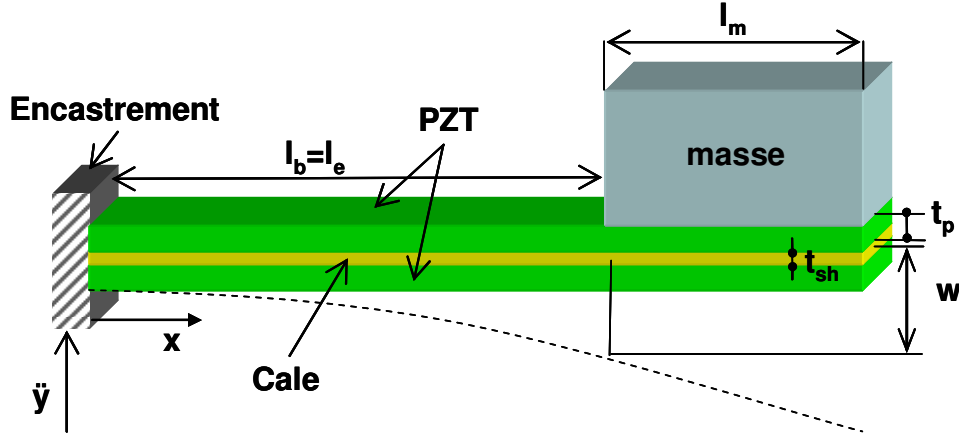


Figure III-12 : Structure du transducteur piézoélectrique

Nous reportons ci-dessous les équations obtenues à partir des équations constitutives de la mécanique des milieux continus et de la piézoélectricité. Elles seront utilisées pour extraire le système d'équations différentielles du modèle analytique de ce type de générateur électrique.

Le moment de rotation de la masse par rapport à l'encastrement est donné par l'équation suivante:

$$M(x) = m(\ddot{y} + \ddot{w})(l_b + \frac{l}{2}l_m - x) \quad \text{III-9}$$

Le moment d'inertie par rapport à l'axe de la poutre est :

$$I = 2 \left[ \frac{w_b t_p^3}{12} + w_b t_p b^2 \right] + \frac{\eta_s w t_{sh}^3}{12} \quad \text{III-10}$$

Avec  $b$  la distance entre le centre de la couche piézoélectrique et le centre de la cale

$$b = \frac{t_p + t_c}{2} \quad \text{III-11}$$

$t_p$  et  $t_{sh}$  sont les épaisseurs de la couche piézoélectrique et la cale.

Le rapport de la constante d'élasticité de la couche PZT "Y" sur celle de la cale "Y<sub>c</sub>" :

$$\eta_s = \frac{Y}{Y_c} \quad \text{III-12}$$

La contrainte dans le matériau piézoélectrique est :

$$\sigma = \frac{I}{l_e} \int_0^{l_e} \frac{b_l M(x)}{I} dx \Rightarrow \sigma = \frac{m}{b_2} (\ddot{y} + \ddot{w}) \quad \text{III-13}$$

Avec



$$b_1 = \frac{3b}{l_b^2} \left( \frac{2l_b + l_m - l_e}{2l_b + \frac{3}{2}l_m} \right) \quad III-14$$

$$\text{et } b_2 = \frac{2I}{b(2l_b + l_m - l_e)} \quad III-15$$

$l_b, l_m, l_e$  sont les longueurs de la poutre, la masse et les électrodes de  $C_{piezo}$  respectivement.

La contrainte extérieure appliquée à la structure piézoélectrique est :

$$\sigma_{in} = \frac{m}{b_2} \ddot{y} \quad III-16$$

La contrainte due à la masse est :

$$\sigma_m = \frac{m}{b_2} \ddot{w} \quad III-17$$

L'équation associée à la déformation d'une poutre de type Euler-Bernouilli est :

$$\frac{d^2 w}{dx^2} = \frac{M(x)}{YI} \quad III-18$$

La déformation en fonction de la contrainte et de la constante d'élasticité est :

$$\delta = \frac{\sigma}{Y} \quad III-19$$

Le déplacement vertical du centre de la masse est donné par :

$$w = \delta \frac{l_b^2}{3b} \left( \frac{2l_b + \frac{3}{2}l_m}{2l_b + l_m - l_e} \right) \Rightarrow w = \frac{\delta}{b_1} \quad III-20$$

La capacité électrique associée aux deux couches piézoélectriques est :

$$C_{piezo} = \frac{a^2 \epsilon w_b l_e}{2t_p} \quad III-21$$

avec  $a = 1$  si les électrodes sont en séries et  $a = 2$  si elles sont en parallèles

L'équation différentielle du déplacement est :

$$\ddot{\delta} = \frac{-Yb_l b_2}{m} \delta - \frac{b_m}{m} \dot{\delta} + \frac{Yb_l b_2}{m} \frac{ad_{31}}{2t_p} v_p + b_1 \ddot{y} \quad III-22$$

avec  $b_m$  l'amortissement mécanique

On introduit la constante effective de raideur  $K_{sp}$  :

$$K_{sp} = Yb_l b_2 \quad III-23$$

En remplaçant  $K_{sp}$  dans l'équation III-22 nous obtenons les équations différentielles ci-dessous :

$$\ddot{\delta} = \frac{-K_{sp}}{m} \delta - \frac{b_m}{m} \dot{\delta} + \frac{K_{sp} a d_{31}}{2 m t_p} v_p + b_l \ddot{y} \quad III-24$$

$$\dot{v}_p = \frac{-2 d_{31} Y t_p}{a \epsilon} \dot{\delta} \quad III-25$$

Nous pouvons déduire l'expression du courant ci-dessous :

$$i_p = -a w_p l_e d_{31} Y \dot{\delta} \quad III-26$$

A partir de ces équations différentielles, nous allons construire les modèles du générateur de puissance en les implémentant sous la forme de diagramme blocs ou en circuit électrique équivalent. Nous posons les hypothèses suivantes :

- La masse agit comme un point sur l'extrémité de la poutre,
- L'effet rotation de la masse sur la poutre est négligeable par rapport au premier mode de vibration,
- La masse de la poutre est négligeable comparée à la masse de la charge.
- L'écart entre l'amortissement mécanique et électrique du système est faible. Ceci, permet d'obtenir une puissance maximale (pas de pertes).
- La puissance est maximale lorsque  $\omega = \omega_p$ .

Nous rappelons que dans nos travaux, nous avons utilisé les modèles développés par des spécialistes afin de valider notre approche de modélisation collaborative multi domaines et multi niveaux. Pour plus de détails concernant ces modèles, nous invitons le lecteur à consulter les travaux de [Rou-04] et [Mar-07].

Nous mettons le système d'équations différentielles décrivant le générateur piézoélectrique en circuit ouvert sous la forme ci-dessous :

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \\ \dot{v}_p \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{K_{sp}}{m} & -\frac{b_m}{m} & \frac{K_{sp} a d_{31}}{2 t_c m} \\ 0 & \frac{-2 t_c d_{31} Y}{a \epsilon} & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \\ v_p \end{bmatrix} + \begin{bmatrix} 0 \\ b_l \\ 0 \end{bmatrix} \ddot{y} \quad III-27$$

Le transfert de l'énergie se fait par une charge. Nous devons donc introduire la composante de la charge électrique dans le modèle. En effet, le courant  $i_p$  fourni par le générateur se divise en deux : le courant dans la capacité  $C_{piezo}$  et le courant aux bornes de la charge externe  $R$ , d'où l'expression du courant  $i_p$  donnée ci-dessous :

$$i_p = C_{piezo} \dot{v}_p + \frac{v_p}{R} \quad III-28$$

En substituant cette expression dans la matrice précédente, nous obtenons l'expression de la dérivée de la tension ainsi que le nouveau système d'équations en fonction de la résistance de charge (III-29).

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \\ \dot{v}_p \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{K_{sp}}{m} & -\frac{b_m}{m} & \frac{K_{sp}ad_{31}}{2t_c m} \\ 0 & \frac{-2t_c d_{31} Y}{a\epsilon} & \frac{-1}{RC_p} \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \\ v_p \end{bmatrix} + \begin{bmatrix} 0 \\ b_1 \\ 0 \end{bmatrix} \ddot{y} \quad III-29$$

En posant  $\omega_n^2 = \frac{K_{sp}}{m}$  qui est la pulsation propre du transducteur et  $\xi = \frac{b_m}{2m\omega_n}$  l'amortissement mécanique du système, nous pouvons réécrire le système ci-dessus comme suit:

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \\ \dot{v}_p \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\omega_n^2 & -2\omega_n \xi & \frac{ad_{31}}{2t_c} \omega_n^2 \\ 0 & \frac{-2t_c d_{31} Y}{a\epsilon} & \frac{-1}{RC_p} \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \\ v_p \end{bmatrix} + \begin{bmatrix} 0 \\ b_1 \\ 0 \end{bmatrix} \ddot{y} \quad III-30$$

Nous pouvons aussi déterminer facilement la transformée de Laplace de la tension en circuit fermé (équation III-32) à partir des équations différentielles ci-dessous :

$$\begin{bmatrix} s\Delta \\ s^2\Delta \\ sV_p \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\omega_n^2 & -2\omega_n \xi & \frac{ad_{31}}{2t_c} \omega_n^2 \\ 0 & \frac{-2t_c d_{31} Y}{a\epsilon} & \frac{-1}{RC_p} \end{bmatrix} \begin{bmatrix} \Delta \\ \Delta s \\ V_p \end{bmatrix} + \begin{bmatrix} 0 \\ b_1 \\ 0 \end{bmatrix} A_{in} \quad III-31$$

Où  $A_{in}$  est la transformée de Laplace du signal d'entrée qui est l'accélération des vibrations.

$s$  est la variable de Laplace

$\Delta$  et  $V_p$  sont les transformées de Laplace de  $\delta$  et  $v_p$  respectivement.

$$V_p = \frac{-2d_{31}Y_t b_m}{a\epsilon} s \quad III-32$$

$$s^3 + \left[ \frac{I}{RC_{piezo}} + 2\xi\omega_n \right] s^2 + \left[ \omega_n^2 (1 + K_{sp}^2) + 2\xi \frac{\omega_n}{RC_{piezo}} \right] s + \frac{\omega_n^2}{RC_{piezo}} A_{in}$$

La transformée de Laplace de la tension permet de passer des équations différentielles à une équation algébrique d'un système non linéaire dynamique et de retrouver la puissance délivrée par le générateur de puissance qui est donnée par l'expression ci-dessous:

$$P_{moy} = \frac{V_p^2}{2R_{load}} = \frac{RC_{piezo}^2 \left( \frac{d_{31}Y_t b_m}{\epsilon} \right)^2}{(2\omega^2) \left[ (4\xi^2 + K_{sp}^4) (RC_{piezo}\omega)^2 + 4\xi K_{sp}^2 (RC_{piezo}\omega) + 2\xi \right]} A_{in}^2 \quad III-33$$

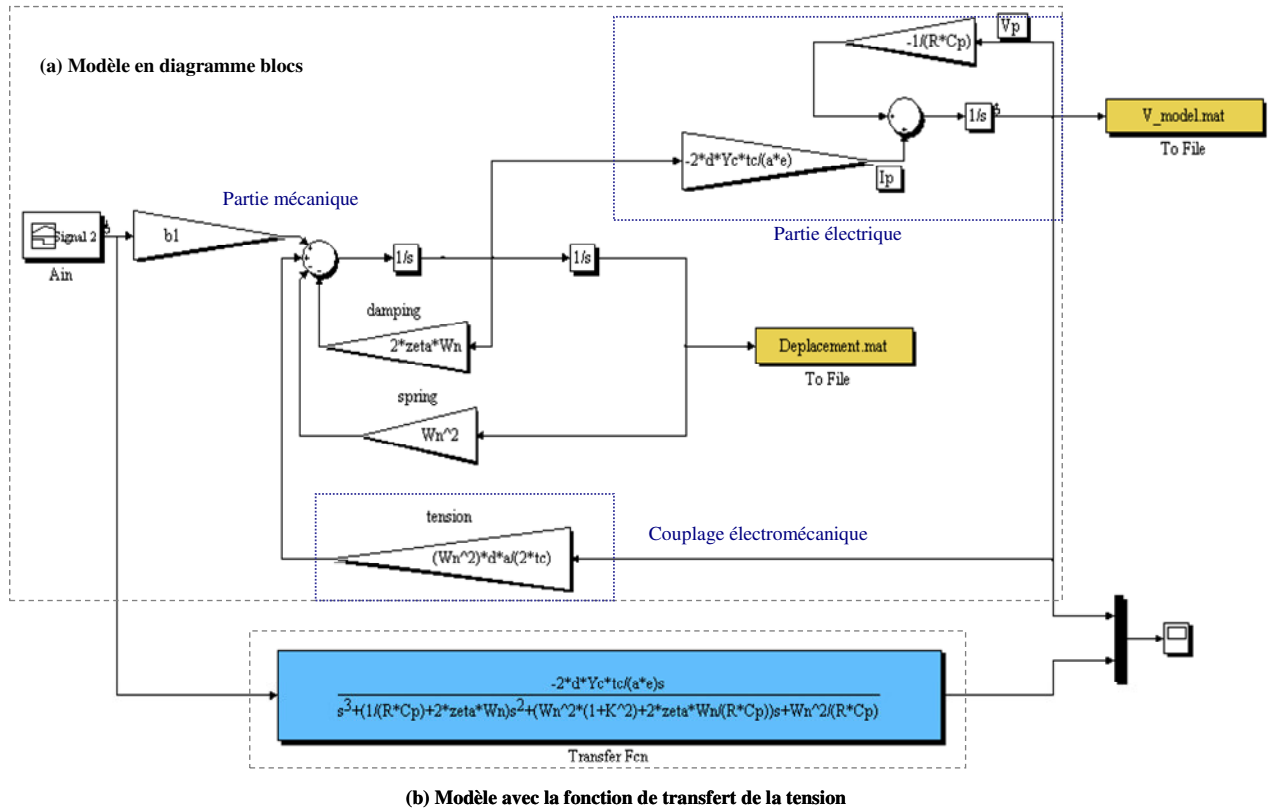
### • Modèle en diagramme blocs

Afin de justifier notre approche de modélisation, nous vérifions chaque modèle par comparaison : soit dans un environnement dédié (ex : SPICE) pour le modèle équivalent électrique, soit analytiquement par des routines développées.

Nous avons implémenté le système d'équations différentielles ordinaires (III-30) sous Simulink. Le modèle est construit avec les éléments prédéfinis de Simulink (Figure III-13) et aussi en association des composants de SimPowerSystems pour modéliser la partie électrique du générateur piézoélectrique (Figure III-14).

La méthode d'implémentation du modèle ci-dessus permet de passer de la matrice d'un système dynamique à un modèle sous une forme de diagramme bloc. Les différentes variables peuvent être observées (la déformation  $\delta$ , le courant  $i_p$  et la tension  $v_p$ ) et le fort couplage physique de la transduction piézoélectrique est réalisé par la réinjection de la tension de sortie  $v_p$  pour recalculer la nouvelle valeur du déplacement.

Sur la Figure III-13, la partie mécanique (calcul de la déformation mécanique) et la partie électrique (calcul du courant et de la tension) sont représentées en clair sur le modèle. Le couplage est marqué par le retour de la tension dans la partie mécanique ce qui modifie la valeur de la déformation, et aussi par la dérivée première de la déformation mécanique qui est proportionnelle au courant généré sur les électrodes de  $C_{piezo}$ .



**Figure III-13 : Implémentation du modèle analytique du microgénérateur sous Simulink : (a) modèle en diagramme blocs, (b) fonction de transfert de la tension**

Le sous-système du générateur modélisé sous la forme ci-dessus peut être aussi implémenté comme le montre la Figure III-14. Sur cette figure, nous montrons qu'il est possible de mélanger des diagrammes blocs et des éléments électriques sous Simulink, en utilisant les composants de SimPowerSystems. Pour la même raison du fort couplage entre la partie mécanique et la partie électrique, la tension est réinjectée mais cette fois-ci en utilisant l'interfaçage entre les deux représentations. L'interface entre les modèles Simulink et les composants de SimPowerSystems est réalisée à travers les sources de *tension/courant contrôlée* et le *bloc de mesure tension/courant*. Cette approche sera utilisée lors de la concaténation des modèles des parties analogiques que nous décrirons plus tard dans le chapitre.

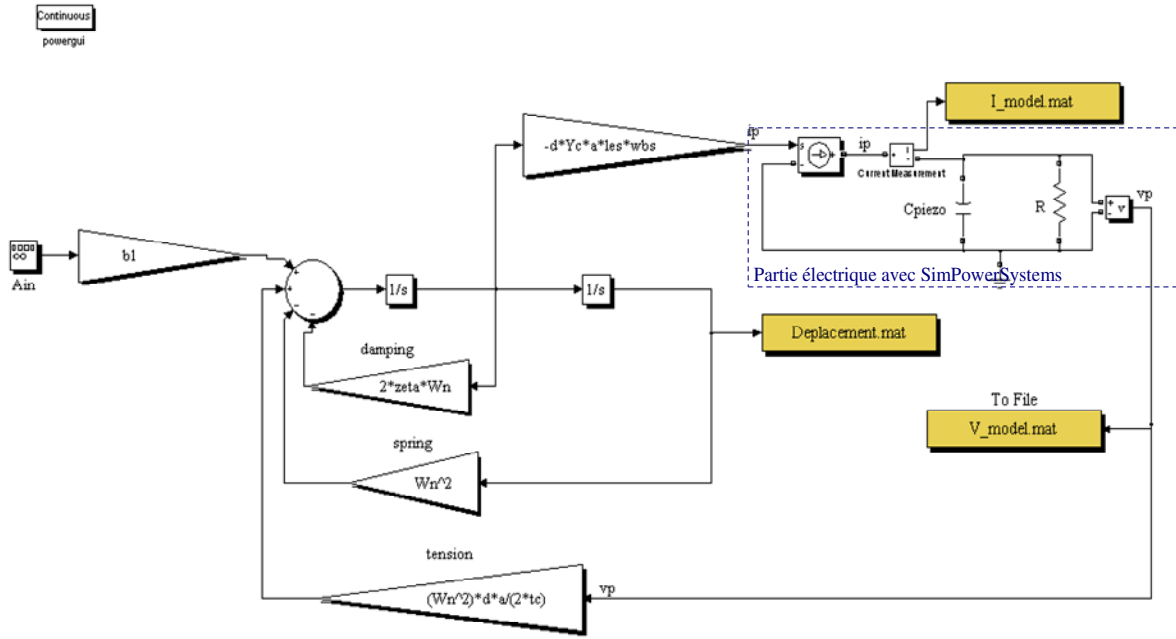


Figure III-14 : Modèle mixte : analytique et électrique (Simulink/SimPowerSystems)

Dans le but de valider le modèle du générateur de puissance, nous avons travaillé sur une structure bimorphe dont les propriétés physiques et géométriques sont données dans le Tableau III-1 [Rou-03].

$l_m$ (mm)	$w_m$ (mm)	$h_m$ (mm)	$l_c$ (mm)	$l_b$ (mm)	$w_b$ (mm)	$t_p$ (mm)	$t_{sh}$ (mm)	$d_{31}$ (C/N)	$Y_c$ (N/m <sup>2</sup> )	$Y_{sh}$ (N/m <sup>2</sup> )	$\epsilon/\epsilon_0$
17,3	3,6	7,7	10,7	10,7	3,2	0,278	0,102	$270 \times 10^{-12}$	$5 \times 10^{10}$	$130 \times 10^9$	3800

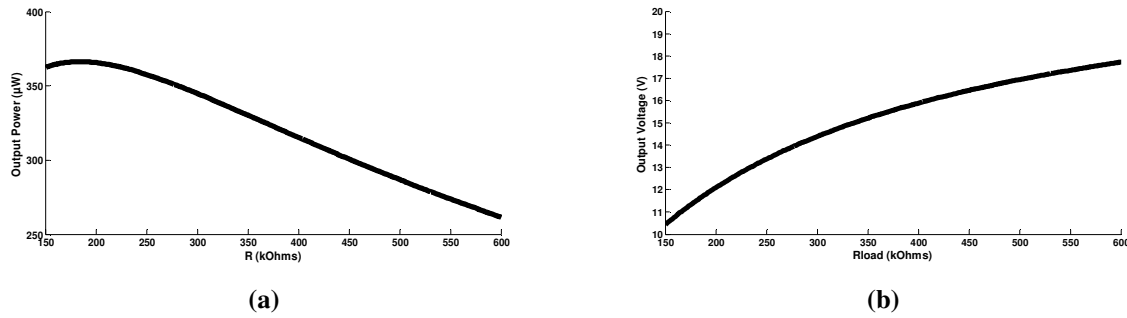
Tableau III-1 : Propriétés du transducteur piézoélectrique à base de matériau PZT-5H

La première étape consistait à vérifier les résultats de la puissance et de la tension et de les comparer avec ceux de l'article [Rou-03]. Pour ce faire, nous utilisons une structure de bimorphe qui est excitée à une fréquence de  $f=120\text{Hz}$  avec une accélération d'une amplitude  $A_{in}=2,25\text{ms}^{-2}$  et un coefficient d'amortissement mécanique  $b_m=0,2$ .

Nous avons développé des routines sous MATLAB pour calculer la puissance et la tension (en circuit ouvert et fermé). Ces calculs permettent la détermination des plages de valeurs de la résistance de charge  $R$  et de la fréquence de résonance  $f_n$  pour lesquelles la tension et la puissance sont maximales.

Sur la Figure III-15-a, nous donnons la courbe de la puissance pour différentes valeurs de la charge. Sur cette courbe nous pouvons prélever la charge optimale pour laquelle la puissance est maximale. Dans ce cas, la puissance est égale à  $366,3\mu\text{W}$  pour  $R=184\text{k}\Omega$ .

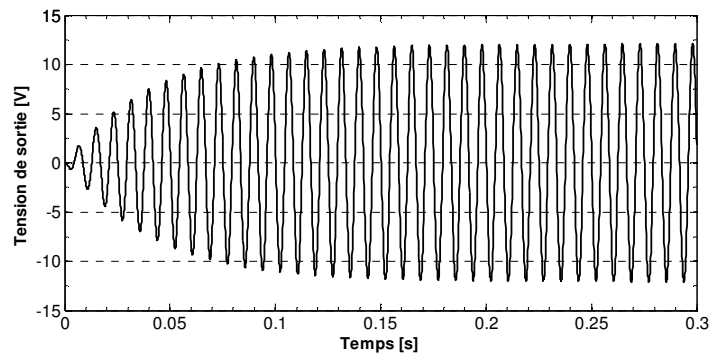
La Figure III-15-b représente la courbe de la tension à la sortie du transducteur piézoélectrique. La tension maximale pour cette résistance optimale est  $V_p = 12,09V$ .



**Figure III-15 : Puissance et tension en fonction de la charge  $R$  (a) résultats publiés [Rou-03] (b) résultats obtenus**

Dans une seconde étape, nous nous référons aux résultats obtenus ci-dessus pour vérifier le modèle décrit par la matrice (III-30) et implémenté sous Simulink (voir Figure III-13 et Figure III-14). Ces étapes de modélisations, nous permettent d'utiliser des modèles pour des conditions idéales afin d'obtenir la performance désirée : **puissance maximale à la sortie du générateur piézoélectrique**. Dans le cas du couplage qui varie, la puissance générée sera en conséquence (d'où l'intérêt du circuit DC/DC que nous allons étudier plus loin).

Nous avons exécuté le modèle pour l'ensemble des propriétés du Tableau III-1 et nous observons la réponse temporelle du générateur comme montré sur la Figure III-16. Nous avons obtenu une tension d'amplitude  $V_p = 12,08V$  et une puissance maximale  $P_{max} = 364,91 \mu W$  pour  $R = 200 k\Omega$  et  $f = 120 Hz$  que nous avons choisi. Nous comparons ce résultat avec la puissance calculée par l'expression III-33 est  $P_{max} = 365,94 \mu W$ , et sur la Figure III-15  $V_p = 12,09V$  et  $P_{max} = 365,61 \mu W$ . l'erreur obtenue entre les différentes méthodes de calcul est due à la résolution des solveurs et aussi aux approximations considérées.



**Figure III-16 : Allure de la tension  $v_p(t)$**

## • Modèle équivalent électrique raffiné

Lors de la modélisation des transducteurs piézoélectriques, le couplage entre les deux aspects mécanique et électrique doit être pris en compte. Le modèle en circuit équivalent électrique comprend dans ce cas les deux aspects et le couplage électromécanique est modélisé par un transformateur [Sher-99] [Pou-04].

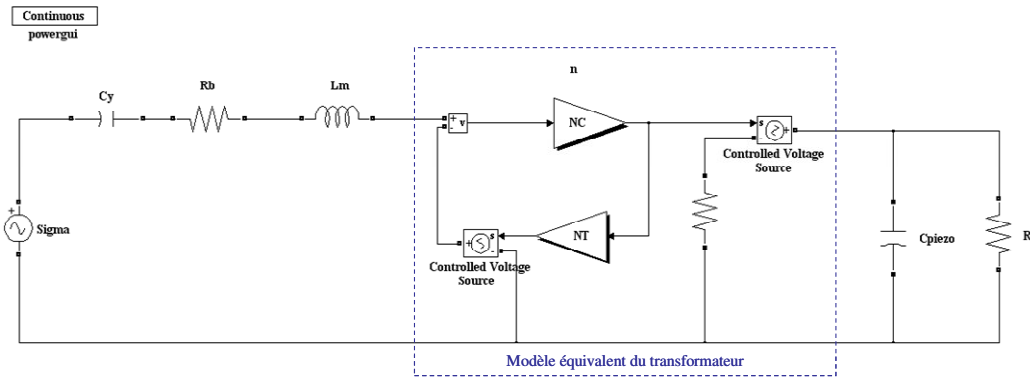
Le circuit électrique se compose d'une inductance équivalente  $L_m$  représentant l'inertie de la masse du générateur, d'une résistance  $R_b$  représentant l'amortissement mécanique, et d'un condensateur  $C_Y$  pour la raideur mécanique. Ces trois éléments constituent le modèle de la partie mécanique décrite par analogie au domaine électrique. Le signal d'entrée est l'accélération appliquée au matériau piézoélectrique qui est représenté par une source de tension. Par ailleurs, la déformation est équivalente à un courant dans le circuit. Pour la partie électrique, le circuit se compose de la capacité  $C_{piezo}$  traversée par le courant  $i_p$  avec  $v_p$  la tension de sortie du transducteur. Le couplage, dans ce cas, est modélisé par le ratio du primaire au secondaire d'un transformateur [Rou-04]. Par application de la loi de Kirchhoff, nous obtenons les relations du déplacement et de la tension de sortie  $v_p$  ainsi que le courant  $i_p$  ci-dessous.

$$\begin{aligned}\sigma_{in} &= L_m \ddot{\delta} + R_b \dot{\delta} + \frac{I}{C_Y} \delta + n v_p \\ i_p &= C_{piezo} \dot{v}_p\end{aligned}\tag{III-34}$$

La Figure III-17 montre le modèle électrique équivalent en utilisant les éléments de la SimPowerSystems implémenté sous Simulink. Les caractéristiques des différents composants du modèle sont données par l'ensemble des expressions ci-dessous. Le transformateur est modélisé comme étant deux contrôleurs de tension réalisés par des blocs de gain ( $N_T$  et  $N_C$ ) associés aux blocs de sources de tension contrôlées de Simulink/SimPowerSystems.

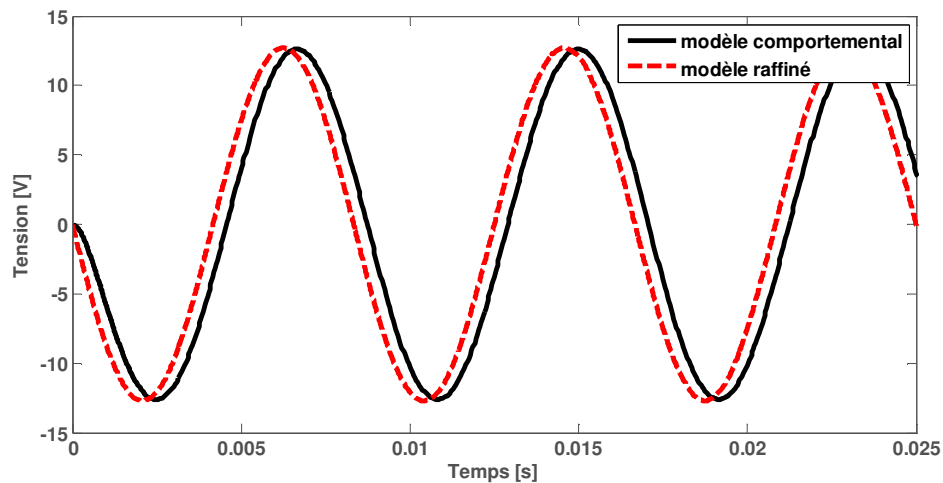
$$\begin{aligned}L_m &= \frac{m}{b_1 b_2}, R_b = \frac{b_m}{b_1 b_2}, C_Y = \frac{I}{Y_c}, N_T = \frac{a d_{31} Y}{2 t_p}, N_C = a \omega l_e d Y_c \\ C_{piezo} &= \frac{a^2 \epsilon \omega_b l_e}{2 t_p}, \\ \sigma_{in} &= \frac{m}{b_2} \ddot{y}, \\ i_p &= a \omega l_e d Y_c \dot{\delta}\end{aligned}\tag{III-35}$$





**Figure III-17 : Schéma électrique du générateur piézoélectrique implémenté sous Simulink à l'aide de la boîte SimPowerSystems**

L'exécution du modèle Figure III-17 sous l'environnement Simulink et SPICE (le modèle sous SPICE permet de valider le modèle avec Simulink/SimPowerSystems) donne une tension maximale  $V_p = 12,69V$  qui a pour allure la courbe temporelle montrée sur la Figure III-18. La puissance maximale récoltée à la sortie du transducteur piézoélectrique soumit une accélération  $A_{in} = 2,25ms^{-2}$  d'une fréquence  $f = 120Hz$  pour une charge résistive  $R = 200k\Omega$  est  $402,6\mu W$ . Ces résultats sont comparables à ceux obtenus pour le modèle comportemental étudié dans la section 2.2.1 précédente.



**Figure III-18: Comparaison entre la tension de sortie du modèle électrique comportemental et raffiné**

La technique de modélisation en circuit électrique équivalent est certainement intéressante, mais à notre sens, nécessite une investigation supplémentaire des domaines d'énergie afin de construire les analogies. Une solution est proposée dans ce qui suit où le modèle peut être construit directement par implémentation des équations différentielles (ODE ou DAE). En effet le modèle mathématique reste la description la plus représentative d'un système. Pour ce faire, nous allons montrer dans la section suivante

que le modèle au niveau circuit (ou composant) est construit sous la forme de diagramme bloc en se basant sur les équations différentielles. Cette technique est utilisée souvent par les automaticiens pour développer leurs modèles.

### 2.2.3 Modèle analytique unidimensionnel d'un générateur MEMS piézoélectrique

Contrairement au modèle précédent, la structure modélisée dans cette partie est microscopique. Le modèle analytique de la structure piézoélectrique unidimensionnelle représentée par le schéma de la Figure III-19 est exploité dans cette partie afin de justifier l'approche d'intégration multi niveaux d'abstraction et aussi d'enrichir la librairie des générateurs de puissance.

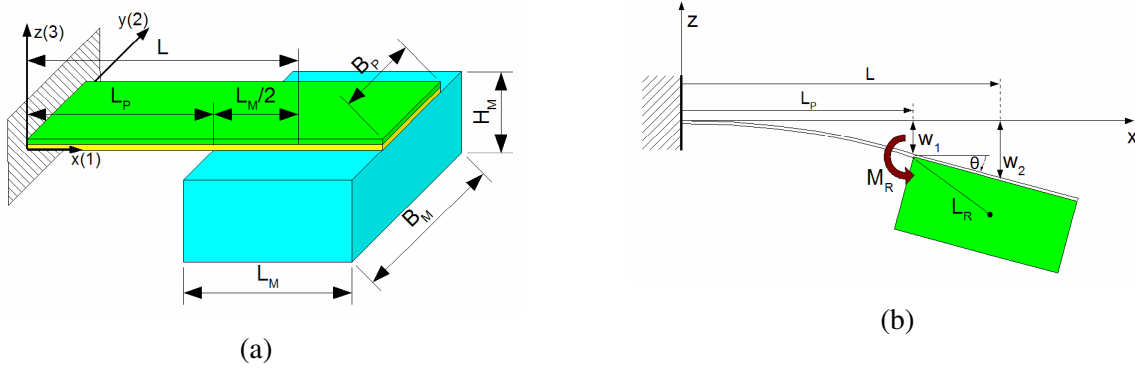


Figure III-19 : (a) Structure de la poutre encastrée-libre (b) Mouvement de la poutre décrite

Le modèle décrit une structure bimorphe asymétrique composée d'une poutre encastrée-libre et d'une grande masse rigide. Le modèle mathématique représente une description dynamique précise en 1D de la structure. Nous n'allons pas nous étaler sur les détails du développement des équations du système qui sont données ci-dessous et qui peuvent être consultés en détail dans la thèse de M. Marzencki [Mar-07].

$$\begin{aligned}
 m\ddot{w}_2 + \frac{B_p D_G''}{L_p L_{eq}^2} w_2 - \frac{\zeta B_p D_G''}{L_{eq}^2} \left( L - \frac{L_p}{2} \right) u + \frac{J_0}{L_{eq}^2} \ddot{\theta} \left( L - \frac{L_p}{2} \right) &= m A_{in} \\
 L_p \dot{u} + \frac{\beta}{\zeta B_p Z D_G''} u - \beta \dot{\theta} &= 0 \\
 \theta &= \left[ \frac{\partial w}{\partial x} \right]_{x=L_p} = \frac{L_p}{B_p D_G''} \left[ m \left( A_{in} - \ddot{w} \right) \left( L - \frac{L_p}{2} \right) - M_R \right] + \zeta L_p u
 \end{aligned}
 \tag{III-36}$$

Nous donnons dans le tableau Tableau III-2 la signification des facteurs constituant le système d'équations.

Désignation	Signification
$w_2$	Variable du déplacement
$u$	Tension électrique aux bornes du système
$\theta$	Angle de rotation de la masse
$m$	Masse
$B_p, L_p$	Largeur et longueur de la poutre
$L, L_{eq}$	Constantes équivalentes intermédiaires pour les longueurs
$D_G, D_G'', \zeta, \beta$	Constantes auxiliaires
$Z$	Charge électrique

Tableau III-2 : Définition des facteurs des équations différentielles

Le système d'équations est défini comme suit:

$$\begin{aligned} \theta &= \frac{L_p}{B_p D_G''} [mb(A_{in} - \ddot{w}_2 - J\ddot{\theta}) + \xi L_p u_0] \\ \dot{u}_0 &= \frac{-\beta}{\xi B_p D_G'' Z L_p} u_0 - \frac{\beta}{L_p} \dot{\theta} \\ \ddot{w}_2 &= A_{in} - \frac{a}{m L_p} w_2 + \frac{\xi}{m} \frac{B_p D_G''}{L_{eq}^2} \left( L - \frac{L_p}{2} \right) \end{aligned} \quad III-37$$

Nous avons procédé à l'implémentation de la matrice des variables couplées sous la forme de diagramme bloc sous Simulink (même technique que le modèle précédent) comme le montre la Figure III-20.

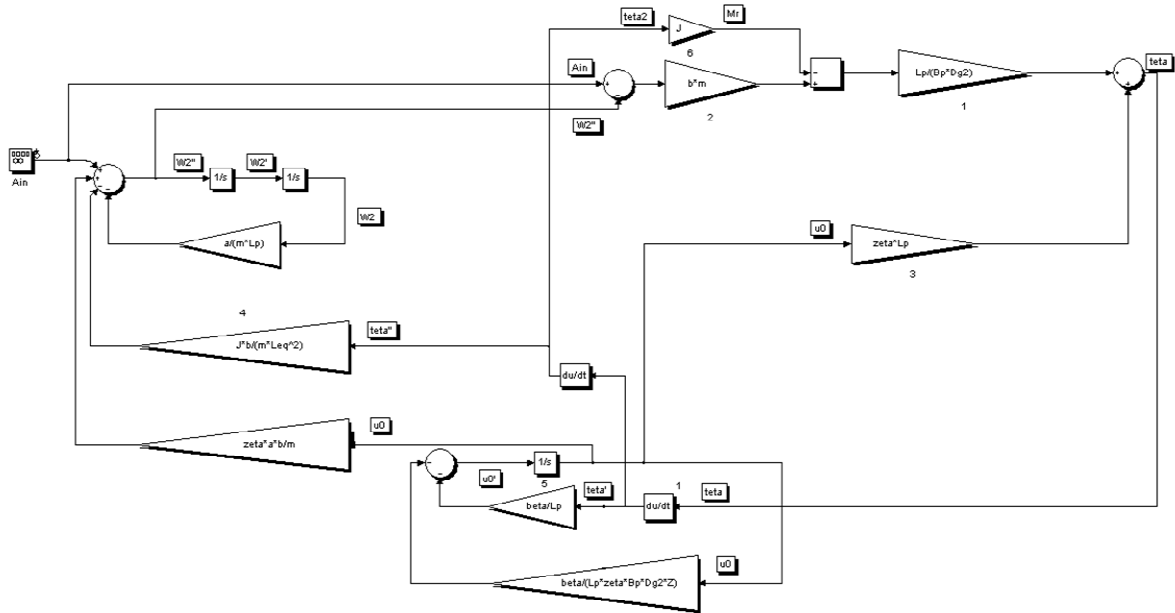
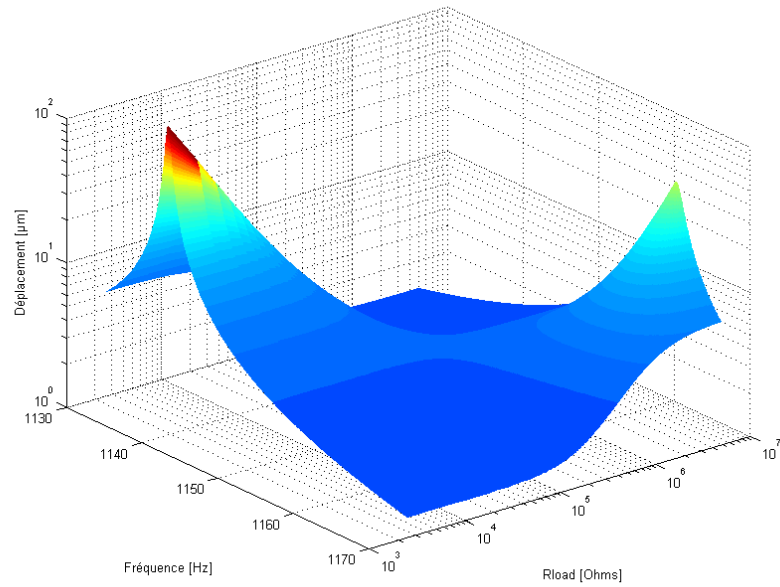


Figure III-20 : Modèle unidimensionnel développé par M. Marzencki [Mar-07]

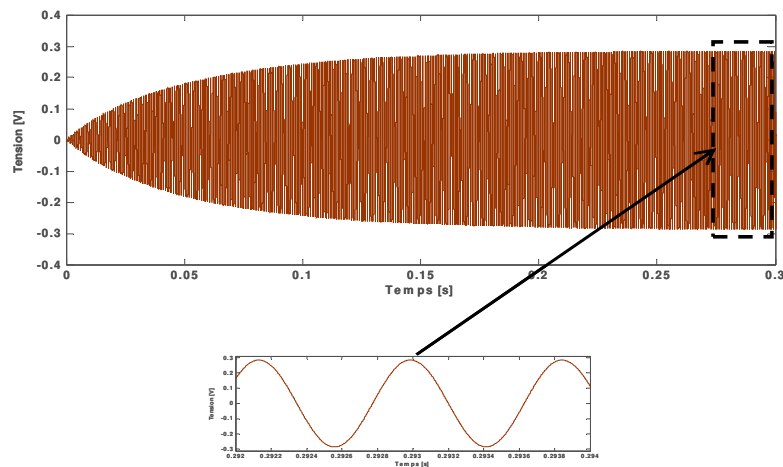
Le générateur piézoélectrique fournit une puissance maximale autour de sa fréquence de résonance  $f_r$  et pour une charge optimale  $R_{opt}$ . Afin de trouver ces paramètres ( $f_{res}$ ;  $R_{opt}$ ), nous utilisons des routines que

nous avons développées sous MATLAB. Les résultats obtenus ci-dessous ne prennent pas en considération les pertes. Pour cette partie nous utilisons les paramètres de l'Annexe E.



**Figure III-21 : Représentation du déplacement en fonction de la fréquence et de la résistance de charge**

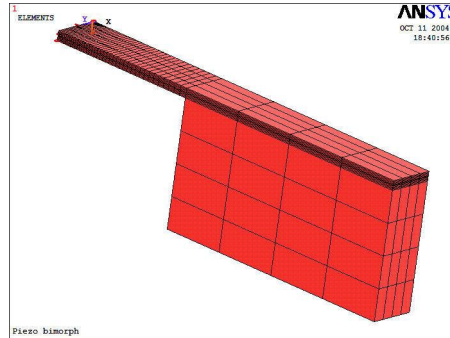
Les deux pics sur cette courbe représentent la fréquence de résonance et la fréquence d'antirésonance. Nous retrouvons pour  $A_{in}=10ms^{-2}$ ,  $f_{res}=1143Hz$  et  $R_{opt}=1k\Omega$  une tension à la sortie du générateur maximale  $V_p=284,3mV$  donc une puissance de  $53,3\mu W$ . l'allure de la tension est donnée par la Figure III-22.



**Figure III-22 : Evolution de la tension en fonction du temps**

## 2.2.4 Modèle physique (EF)

Dans le cadre de ce travail, nous avons fait appel au travail d'un collègue du groupe MNS : [Mar-07] pour la conception au niveau physique du générateur de puissance. Le but de cette collaboration est d'intégrer le modèle au niveau physique du générateur sous l'environnement Simulink. Pour ce faire, le collègue dans le cadre de sa thèse utilise **ANSYS** comme outil de modélisation en utilisant l'analyse en éléments finis (FEM<sup>33</sup>). Comme modèle de départ, nous avons utilisé la structure bimorphe [Rou-03] dont la représentation dans l'environnement ANSYS est donnée par La Figure III-23.



**Figure III-23 : Exemple de modélisation avec les EF**

Nous rappelons que nous avons besoin de faire des études dans le domaine temporel à chaque fois, afin d'observer et de valider le comportement du microsystème. En réalité la structure ci-dessus doit être connectée au reste du système (c'est-à-dire une charge électrique). Actuellement, un outil supportant la modélisation d'un microsystème MSMD complet n'existe pas. Ainsi, pour ce niveau d'abstraction, le développement des modèles reste propre à chaque domaine. Toutefois, nous proposons dans le cadre de la modélisation collaborative de réaliser des simulations d'une façon itérative entre ANSYS et Simulink. La méthode consiste à intégrer les résultats obtenus pour une charge électrique donnée sous ANSYS dans le modèle Simulink.

Tout d'abord, nous devons valider cette approche qui sera valable seulement pour les hypothèses suivantes : l'impédance vue par le générateur de puissance est connue et le couplage est faible. Cette approche peut être intéressante pour les systèmes dont le modèle analytique ou le circuit équivalent électrique sont difficiles à développer (exemple des structures complexes) du type de celle étudiée au paragraphe 2.2.3.

- **Comparaison du modèle FEM avec le modèle analytique**

<sup>33</sup> **FEM**- de l'anglais Finite Element Method

Nous avons procédé à la comparaison du modèle analytique raffiné développé précédemment et le modèle en éléments finis de la Figure III-23 fourni par M. Marzencki en utilisant les mêmes paramètres géométriques et physiques.

Le résultat de la simulation sous ANSYS est obtenu pour la structure ci-dessus. Dans cette étude, la structure est soumise à un choc d'une amplitude de  $A_{in}=2,5ms^{-2}$  d'une durée de  $10ms$ . Une charge résistive  $R_{load}=151k\Omega$  est connectée à cette structure afin d'évaluer la tension à ces bornes. Ici, nous considérons l'amortissement mécanique  $b_m=2\%$ . Les résultats sous ANSYS sont sauvegardés à la fin des simulations dans un fichier texte. Ce fichier texte est converti en un fichier de données (.mat) qui peut être appelé sous l'environnement de Simulink (voir la Figure III-24). Le fichier de données est composé de deux lignes : la première correspond au temps et la deuxième à la tension correspondante. La notion du temps est préservée dans le fichier avec l'extension .mat. A chaque pas de simulation, la valeur de la tension est interpolée et est convertie en son équivalent électrique (au delà des points existant dans le fichier .mat le modèle diverge). L'interface entre le fichier et le circuit électrique peut être une simple source de tension contrôlée (controlled voltage source).

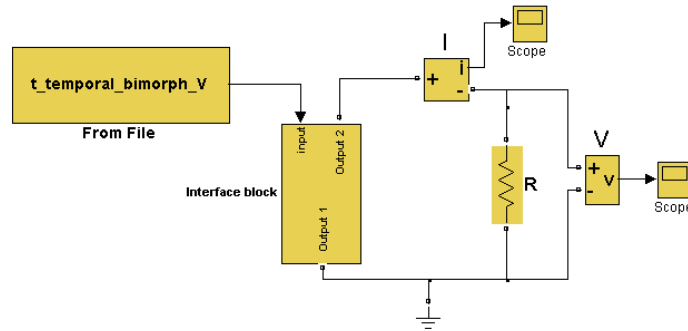


Figure III-24 : Exemple d'un fichier de résultats EF implémenté sous Simulink

Nous avons utilisé l'outil d'analyse linéaire *LTI Viewer* pour analyser la fonction de transfert du système et obtenir sa fréquence propre qui est  $f_0=114Hz$  (voir la Figure III-25). La fréquence propre déterminée par EF est quant à elle  $f_0\approx 113Hz$ , qui est très légèrement différente de celle du modèle analytique. L'erreur est inférieure à 1%.

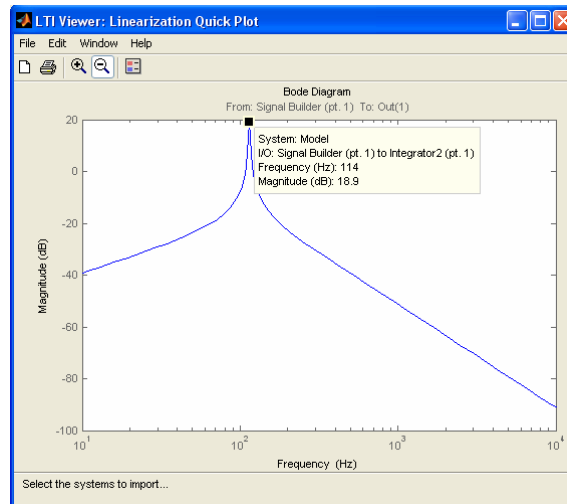


Figure III-25 : Diagramme de Bode de la tension

La Figure III-26 illustre l'évolution dans le temps de la tension du générateur piézoélectrique soumis à un choc. Les deux courbes sont légèrement déphasées à cause de l'erreur de la fréquence propre.

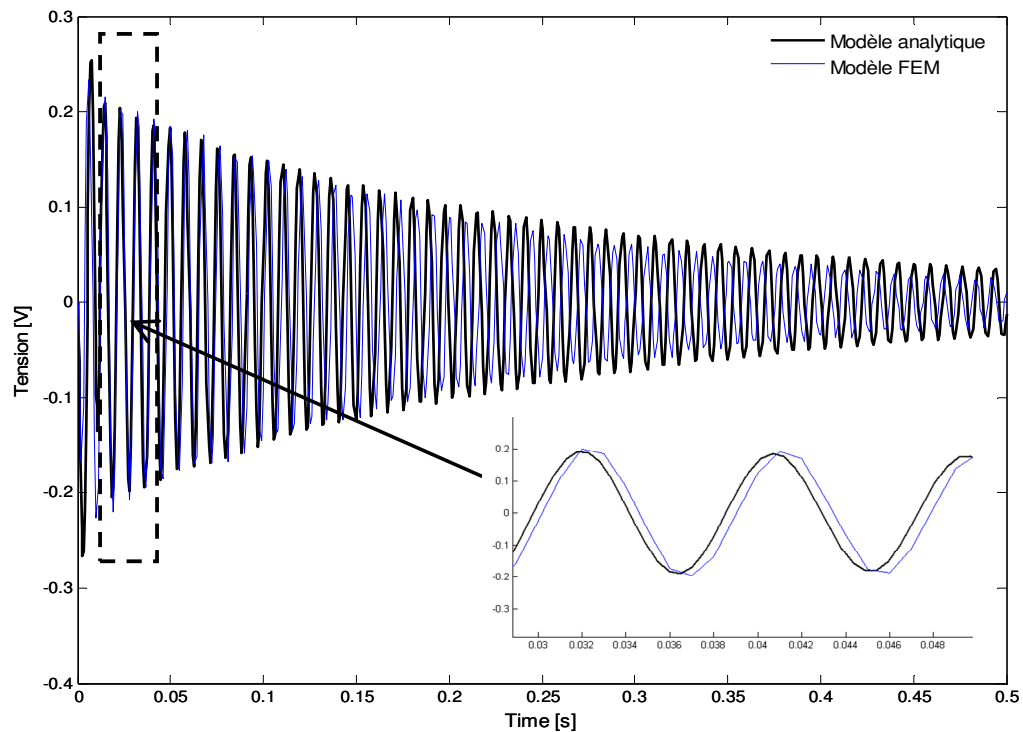


Figure III-26 : Comparaison entre le modèle analytique sous Simulink et le modèle en EF

Ce modèle reste valable pour un couplage faible entre le générateur de puissance et la charge connectée à ses bornes. Toutefois, cette approche serait intéressante si la cosimulation entre ANSYS ou un

autre outil de modélisation au niveau physique (PDE, FEM) et MATLAB/Simulink existait. Dans ce cas, le fichier de données serait actualisé en temps réel tout au long de l'exécution des simulations en fonction de la charge connectée aux bornes du transducteur.

## 2.3 Construction d'une bibliothèque de générateurs piézoélectriques

Simulink offre la possibilité de construire des bibliothèques de modèles personnalisés. Le modèle est construit graphiquement ou par programmation (en C ou en MATLAB), puis encapsulé sous son *masque*. La création du masque rend le modèle paramétrable. La boîte de dialogue du masque s'ouvre par un simple clic et propose des paramètres du modèle. Ils peuvent être modifiés en cours de simulation et sont associés aux variables génériques des blocs subordonnés.

La Figure III-27 illustre la fonction du générateur de puissance sous la forme d'un bloc paramétré. Lors de la création du masque, seuls les paramètres modifiables sont accessibles et peuvent être modifiés en temps réel. En revanche, le modèle peut être amélioré en éditant son masque (*Edit Mask*) ou en regardant sous le masque (*Look Under Mask*). Pour interfacer la partie Simulink et la partie SimPowerSystems, un module qui a pour rôle l'adaptation des signaux, est inséré. Dans ce cas, le signal d'entrée est le courant  $i_p$ .

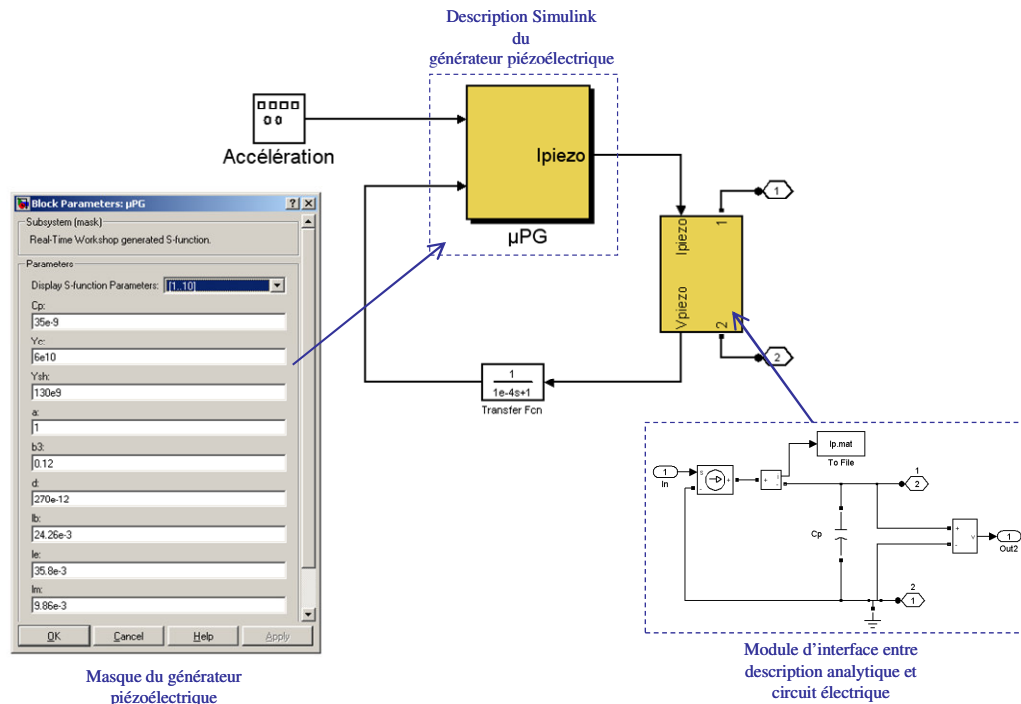


Figure III-27 : Modèle paramétrable du générateur de puissance



La boîte de dialogue du masque permet d'introduire les valeurs des paramètres d'un modèle comme n'importe quel autre bloc dans les bibliothèques de Simulink. L'avantage de cette technique consiste en la réutilisation des modèles personnalisés sans se préoccuper de ce qu'il y a à l'intérieur. Une fiche technique peut être éditée à titre explicatif.

La création de masque pour le modèle analytique et son interface avec les parties électriques conduit à un système sans état qui génère un problème de boucle algébrique. La résolution de la boucle algébrique passe par l'introduction de blocs ou de fonctions de transfert qui introduisent un retard. Ici, la fonction de transfert du premier ordre  $H(s) = \frac{I}{I + s\tau}$  est utilisée. La constante de temps  $\tau$  doit être suffisamment petite par rapport à la fréquence du générateur piézoélectrique (exemple :  $\tau=0,1ms$  pour  $f_{piezo}=120Hz$ ).

## 2.4 Bilan

Dans cette partie du manuscrit, nous avons proposé sous MATLAB/Simulink différentes descriptions de générateurs piézoélectriques à différents niveaux d'abstraction. Dans le tableau ci-dessous, sont présentées les points forts et les points faibles de chaque méthode pour les trois niveaux de modélisation abordés pour le sous-système de génération de puissance piézoélectrique.

Type de modélisation	Niveaux de modélisation		
	Comportemental	Circuit/composant	Physique
Description Mathématique (Simulink)	<b>Avantages:</b> <ul style="list-style-type: none"> <li>➤ Implémentation simple des relations entre courant et tension fourni par le générateur: fonction de transfert, circuit électrique</li> <li>➤ Obtention du comportement et des résultats rapidement</li> </ul> <b>Limitations:</b> <ul style="list-style-type: none"> <li>➤ Aucune notion sur l'influence des paramètres physiques et structurels</li> <li>➤ Le fort couplage dans la piézoélectricité n'est pas représenté</li> </ul>	<b>Avantages:</b> <ul style="list-style-type: none"> <li>➤ Implémentation directe des ODE en diagramme blocs</li> <li>➤ Modélisation du couplage fort électromécanique</li> <li>➤ Interfaçage avec SimPowerSystems</li> <li>➤ Précision du modèle: visibilité sur les paramètres structurels et physiques</li> <li>➤ Etudes de la performance recherchée: puissance maximale en fonction de la fréquence et de la charge optimale</li> </ul> <b>Limitations:</b> <ul style="list-style-type: none"> <li>➤ Pas d'étude paramétrable dans le même modèle: développement de routines supplémentaires</li> </ul>	<b>Avantages:</b> <ul style="list-style-type: none"> <li>➤ Modèle précis</li> </ul> <b>Limitations:</b> <ul style="list-style-type: none"> <li>➤ Pas de lien en temps réel avec une étude en EF</li> <li>➤ Le fort couplage n'est pas représenté:</li> </ul>
Circuit électrique (SimPowerSystems)		<b>Avantages:</b> <ul style="list-style-type: none"> <li>➤ Réutilisation de modèles existant</li> <li>➤ Modélisation du couplage fort électromécanique</li> </ul> <b>Limitations:</b> <ul style="list-style-type: none"> <li>➤ Réaliser l'analogie entre les domaines</li> <li>➤ Nécessité de développer des composants électriques</li> </ul>	N'est pas réalisé

**Figure III-28 : Tableau récapitulatif des avantages et des inconvénients de la modélisations d'un MEMS sous Simulink/SimPowerSystems**

Pour le niveau de modélisation physique, nous pouvons facilement imaginer que si l'on dispose d'un lien direct entre MATLAB/Simulink et un environnement dédié à l'analyse en éléments finis ou des équations différentielles partielles, il serait difficile de vérifier le comportement d'un microsystème complet. La limitation peut être vue en termes de temps de simulation et de ressources de stockage temporaire par exemple. Il n'est donc pas nécessaire d'intégrer les descriptions au niveau physique pour tous les sous-systèmes à chaque fois, car ceci rend le modèle plus compliqué et les simulations peuvent être difficiles à réaliser. En effet, au lieu de simuler tout le système avec toute sa complexité, il est possible de concentrer l'étude sur un aspect particulier du fonctionnement.

Nous rappelons que dans la suite de nos travaux, nous utilisons des dispositifs macroscopiques afin de réaliser le prototype virtuel du microsystème (voir la partie B de ce chapitre).

### 3. Modélisation du circuit de récolte d'énergie

Le circuit de récolte d'énergie (EHC<sup>34</sup>) représente la partie de gestion de l'énergie et est schématisé par la Figure III-29. Ce circuit a pour but de redresser la tension alternative  $v_p$  du générateur piézoélectrique avec le convertisseur AC/DC, puis fournir l'énergie récoltée à la batterie à travers le circuit d'adaptation de la tension DC/DC.

Lors du partitionnement du microsysteme autonome, le circuit de récolte d'énergie est le sous-système constituant la partie électronique analogique/numérique. Ce circuit se compose d'éléments électriques linéaires (résistances, capacités, inductance) et non linéaires (diodes, MOSFET) et d'une partie de contrôle numérique. La conception de circuits analogique/numérique fait partie des domaines les mieux maîtrisés en termes de méthodes et d'outils de développement et de simulation.

Dans cette section, nous présentons les modèles à différents niveaux d'abstraction des sous-systèmes du circuit de récolte de l'énergie. Les modèles sont développés en utilisant MATLAB/Simulink et plus particulièrement la boîte à outil SimPowerSystems destinée aux circuits de l'électronique de puissance et que nous adaptons pour notre application. Des modèles plus raffinés pour les composants non linéaires de cette toolbox ont fait l'objet d'une étude au sein du groupe MNS. Ces modèles sont intégrés afin d'améliorer la précision des éléments prédéfinis de SimPowerSystems.

Nous avons mentionné précédemment que la tension alternative produite par le générateur doit être adaptée pour alimenter efficacement avec la batterie le reste des modules du nœud sans fil. La fonction principale du circuit de récolte de l'énergie est de conditionner le signal du générateur, et de le transformer en un courant continu (DC) pour charger l'élément de stockage.

Le circuit de récolte d'énergie réalise deux tâches:

- La conversion du signal alternatif en sortie du générateur piézoélectrique en un signal continu par redressement avec un convertisseur AC/DC.
- L'adaptation de la tension fournie par l'AC/DC pour une batterie spécifique par transfert du maximum de puissance. Cette fonction est réalisée par un convertisseur DC/DC composé par les circuits compensateur, commande et contrôleur<sup>35</sup>.

Le schéma bloc du circuit de récolte d'énergie est donné par la Figure III-29 (bloc encadré). La tension alternative du générateur piézoélectrique  $v_p$  est redressée en une tension continue  $v_{rect}$ , qui à son tour est adaptée pour obtenir la tension aux bornes de la batterie  $v_{bat}$ . Dans ce cas de figure, le circuit de récolte d'énergie joue le rôle de source de courant pour la batterie.

<sup>34</sup> **EHC** – de l'anglais **E**nergy **H**arvesting **C**ircuit

<sup>35</sup> Le contrôleur constitue la partie numérique du DC/DC

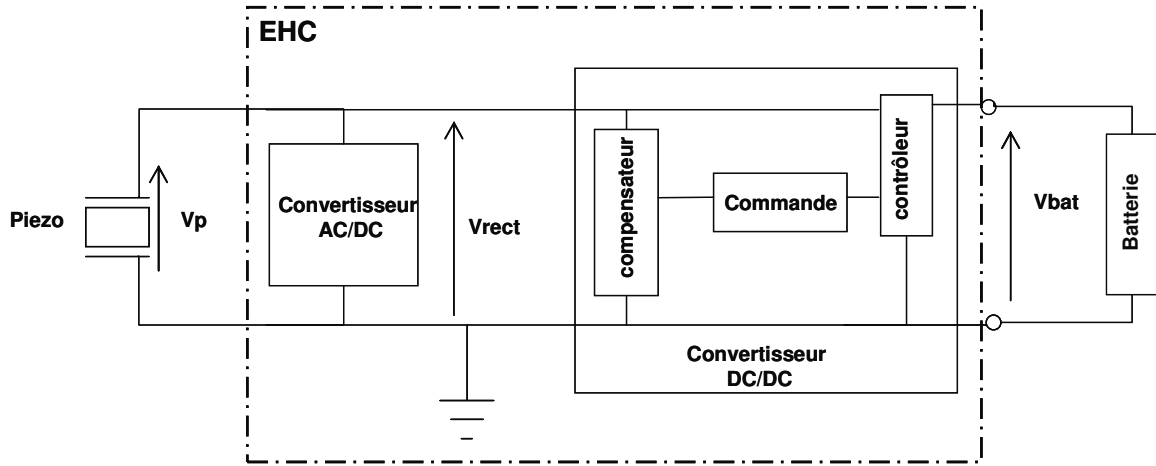


Figure III-29 : Schéma bloc du circuit de récolte d'énergie

Dans cette étude, nous allons considérer des architectures de convertisseurs AC/DC et DC/DC classiques qui sont bien adaptées à des générateurs fournissant suffisamment d'énergie par rapport aux pertes intrinsèques de ces deux modules. Il est à noter que dans le groupe MNS, la thèse d'Y. Ammar a porté sur des architectures originales ultra faible consommation de ces convertisseurs [Amm-07].

### 3.1 Redresseur AC/DC

Le redresseur AC/DC double alternance est composé d'un pont de Graëtz et d'un condensateur  $C_{rect}$  (voir la Figure III-30). Pour que les diodes du pont de Graëtz conduisent, il est nécessaire que la tension délivrée par le transducteur  $v_p$  soit supérieure à la tension de seuil des diodes. Ce problème ne se pose pas dès lors que nous utilisons des générateurs de grandes dimensions qui fournissent des tensions élevées et des puissances plus importantes que les pertes introduites par les diodes.

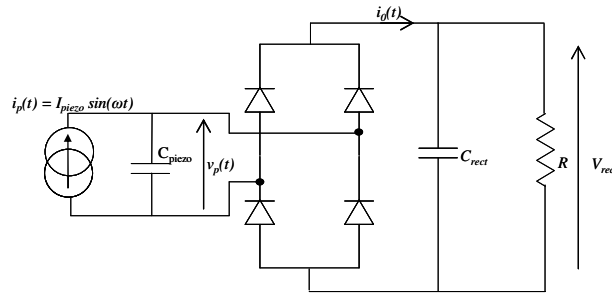


Figure III-30 : Schéma électrique du redresseur AC/DC

Le courant du générateur piézoélectrique  $i_p$  est un signal sinusoïdal représenté sur la Figure III-31 (a). Tant que la tension aux bornes de  $C_{piezo}$  est inférieure à la tension  $V_{rect}$  les diodes restent bloquées et le courant  $i_o(t)$  est nul. Lorsque cette tension devient supérieure, le courant circule vers la capacité  $C_{rect}$  et la charge.

La tension du générateur a l'allure reportée sur la figure b.

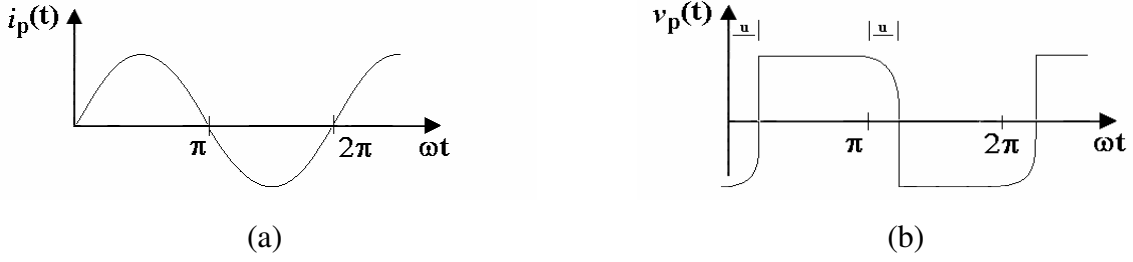


Figure III-31 : (a) Allure du courant du générateur, (b) Allure de la tension

Le courant  $i_o(t)$  peut être décrit comme suit :

$$\begin{aligned} i_o(t) &= 0 & 0 < \omega t < u \\ i_o(t) &= \frac{C_{rect}}{C_{rect} + C_p} I_p \sin(\omega t) & u < \omega t < T \end{aligned} \quad \text{III-38}$$

Et l'allure du courant  $i_o(t)$  est donnée par la figure ci-dessous:

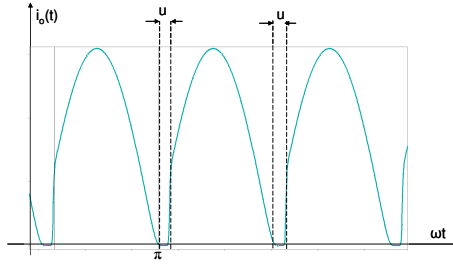


Figure III-32 : Allure du courant de l'AC/DC

Dans ce cas, nous faisons l'hypothèse que la capacité  $C_{rect}$  est beaucoup plus importante que la capacité  $C_{piezo}$  ce qui permet un bon filtrage de la tension alternative et que la tension  $v_{rect}$  peut être considérée comme constante.

L'expression du courant moyen  $i_o(t)$  en fonction de l'amplitude du courant du générateur  $I_p$  et de la tension de sortie  $V_{rect}$  ainsi que la puissance moyenne récupérée  $P_{moy}$  sont données par les expressions III-39 et III-40 respectivement.

$$\langle i_o(t) \rangle = \frac{2I_p}{\pi} - \frac{2V_{rect}\omega C_{piezo}}{\pi} \quad \text{III-39}$$

$$P_{moy} = \frac{2V_{rect}}{\pi} (I_p - V_{rect}\omega C_{piezo}) \quad \text{III-40}$$

Les expressions ci-dessus sont une description de **haut niveau** du convertisseur AC/DC. Dans la suite, nous nous intéressons à modéliser ce dispositif pour l'intégrer d'une façon hiérarchique dans la chaîne des sous-systèmes constituant le microsystème.

### 3.1.1 Implémentation de l'AC/DC

Nous avons implémenté le circuit redresseur sous Simulink en utilisant les composants électriques de la toolbox SimPowerSystems. Afin d'observer l'évolution de la tension issue du générateur piézoélectrique, le courant et la tension à la sortie du convertisseur AC/DC, nous avons utilisé les paramètres du Tableau III-3. Ces paramètres sont extraits de la publication d'Ottman et al. [Ott-02].

Composant/paramètre	Valeur
$I_p$	3,1mA
$C_{piezo}$	0,184 $\mu$ F
<i>Diodes</i>	1N4002
$C_{rect}$	50 $\mu$ F
$R$	24k $\Omega$

Tableau III-3 : Composants utilisés pour modéliser le AC/DC d'après [Ott-02]

La Figure III-33 représente le circuit AC/DC sous Simulink en utilisant les composants prédéfinis de la toolbox SimPowerSystems.

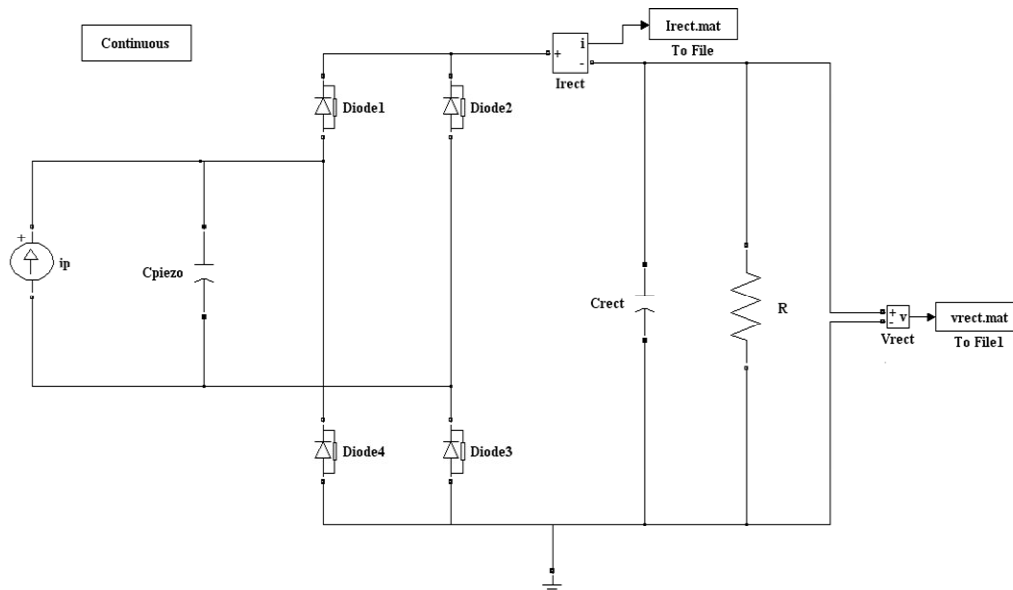
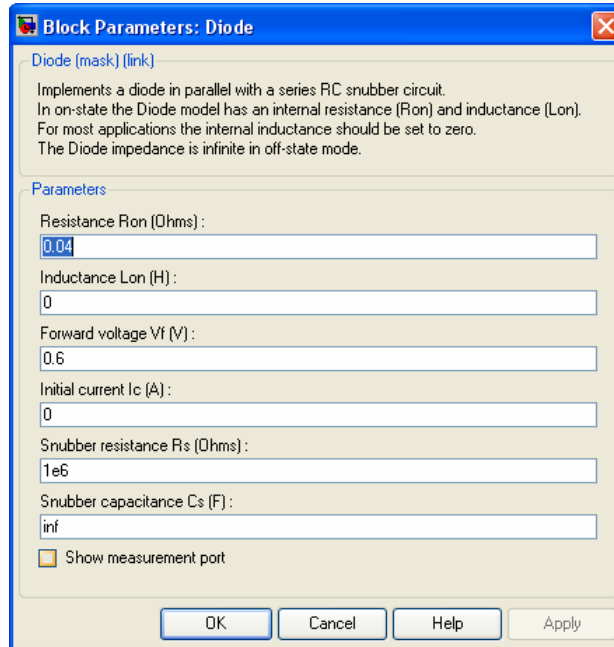


Figure III-33 : Schéma de l'AC/DC sous Simulink avec les composants de SimPowerSystems

Les paramètres d'une diode sous SimPowerSystems sont introduits comme le montre la Figure III-34. Nous avons extrait des caractéristiques de la diode 1N4002  $R_{on}$  et  $V_f$ . La résistance et la capacité de

l'effet appelé snubber ont les valeurs :  $C_s = \text{inf}$  et  $R_s = 10^6$ . Nous étudierons cet effet dans la section pour le sous-système DC/DC.



**Figure III-34 : Exemple de la diode 1N4002 sous SimPowerSystems.**

Pour exécuter ce modèle qui contient des composants non linéaires sous Simulink, il est préconisé d'utiliser les solveurs ode13tb, ode13s et ode15s [Sim-05]. Ces solveurs sont utilisés pour les systèmes avec des équations différentielles non linéaires. Nous avons essayé chacun de ces solveurs avec différents paramètres de simulations. Finalement, nous avons opté pour la configuration donnée dans le Tableau III-4. Le modèle ci-dessous est construit avec les composants de SimPowerSystems et les paramètres internes des diodes pouvant être ajustés en fonction des composants réels choisis.

Paramètres	Valeur
Solveur	Ode15s (stiff/NDF)
Pas initial	$5 \times 10^{-6}$
Tolérance relative	$1 \times 10^{-4}$
Méthode d'initialisation	Robust

**Tableau III-4 : Paramètres de simulation de l'AC/DC**

Nous reportons sur la Figure III-35, l'allure du courant de L'AC/DC. Afin de vérifier le modèle, nous avons aussi comparé les résultats avec ceux d'un modèle construit sous SPICE.

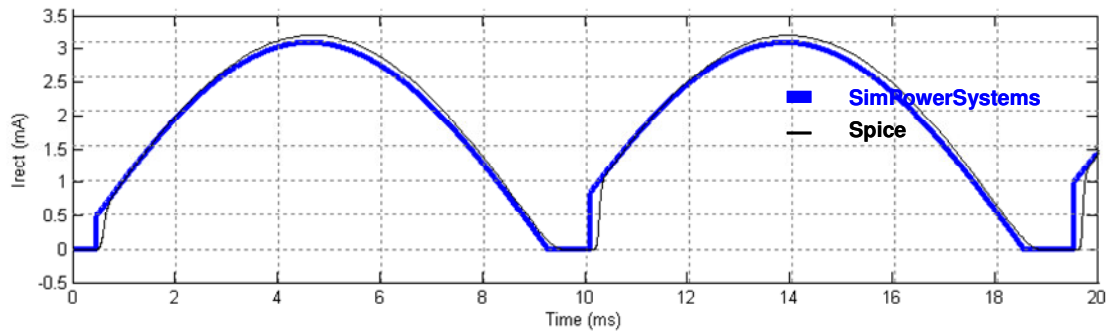


Figure III-35 : Comparaison des allures du courant  $I_{rect}$  à la sortie de l'AC/DC sous SimPowerSystems et SPICE

Pour obtenir ce bon accord, nous avons repris dans la bibliothèque de SPICE, les paramètres pertinents de la diode que nous avons ensuite introduits sous SimPowerSystems. **Il est à noter que les temps de simulation sous l'environnement MATLAB/Simulink sont à ce stade plus lent que ceux sous SPICE (1mn pour Simulink et quelques secondes pour SPICE).**

La Figure III-37 montre l'évolution de la tension de sortie du générateur de puissance ( $v_p$ ) et le courant ( $i_p$ ) (modèle de la section 2.2.1). Cette courbe a la même allure que la Figure III-31 du développement théorique de l'AC/DC.

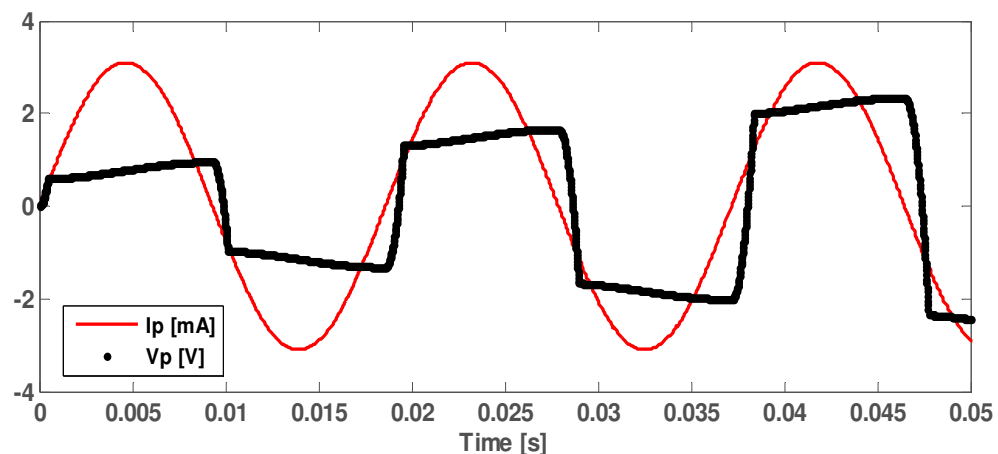


Figure III-36 : Allure de la tension du générateur de puissance à l'entrée de l'AC/DC avec SimPowerSystems

Nous reportons aussi la courbe de la tension de sortie de l'AC/DC sur Figure III-37. En régime permanent cette tension aux bornes d'une charge résistive (24k $\Omega$ ) est égale à 22,81V.



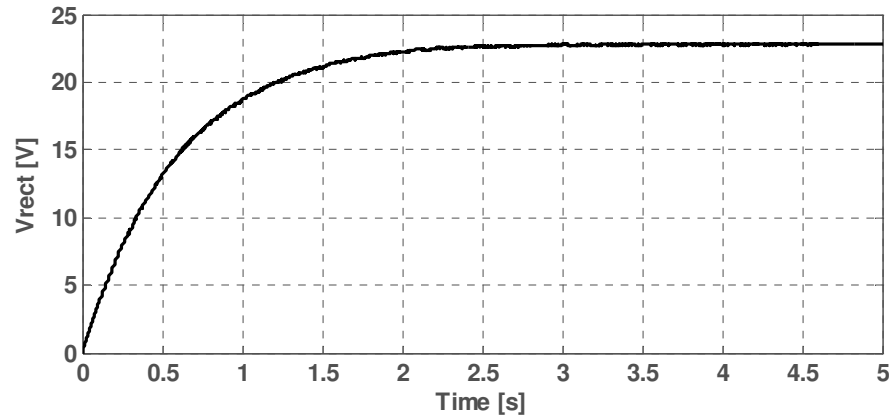


Figure III-37 : Allure de la tension de sortie de l'AC/DC avec SimPowerSystems

Pour le circuit AC/DC, nous avons eu de bons résultats en comparaison avec le modèle sous SPICE. Nous pouvons à ce stade valider le modèle de l'AC/DC sous Simulink et l'utiliser ultérieurement dans la suite de nos travaux.

### 3.2 Convertisseur DC/DC

Le convertisseur DC/DC est le circuit qui transfère la puissance maximale du générateur piézoélectrique vers la charge. Ce dispositif adapte la tension d'entrée pour une charge donnée (ex : une batterie).

Ce circuit est nécessaire pour un transfert optimum de l'énergie et a pour rôle principal l'adaptation des signaux entre le rectifieur AC/DC et la batterie. Dans nos travaux, nous nous sommes intéressés aux convertisseurs DC/DC à découpage alors que plusieurs autres types sont exploités dans la littérature [Amm-07].

Il existe plusieurs types de convertisseurs DC/DC à découpage (BUCK/abaisseur, BOOST/élevateur, BUCK-BOOST/inverseur). Dans notre cas, nous supposons que le transducteur piézoélectrique délivre une tension supérieure à celle de la batterie. Pour adapter, la tension d'entrée  $V_{rect}$  à la tension de la batterie  $V_{bat}$ , nous utilisons un abaisseur de tension. Ce circuit représenté par la Figure III-38 permet aussi d'éviter le passage du courant dans l'étage AC/DC. En régime permanent, la tension aux bornes de la batterie est quasiment constante, et donc le transfert de la puissance maximale se fait par le courant.

Nous retrouvons dans la littérature, deux types de fonctionnement de l'abaisseur de tension : le mode continu et le mode discontinu. Le mode discontinu correspond à l'annulation du courant dans l'inductance pendant une période de fonctionnement. Cette situation se produit lorsque le courant moyen absorbé par la charge est inférieur à la moitié de l'ondulation du courant dans l'inductance. Le mode de fonctionnement continu a comme caractéristiques un courant moins ondulé en sortie, des contraintes

moins fortes sur les dispositifs, mais un temps de réaction plus long que le mode discontinu. Dans les deux modes le convertisseur est caractérisé par un courant d'entrée discontinu, ce qui implique une perturbation harmonique dans la source de puissance.

Par définition, l'abaisseur de tension diminue la tension appliquée à l'entrée au profit de la charge. Le montage de base du dispositif se compose d'un commutateur (transistor MOS), d'une diode dite "de roue libre", d'une inductance et d'un condensateur.

L'inductance et le condensateur jouent le rôle de générateur de courant et de tension respectivement. La diode de roue libre permet d'éviter le courant de fuite. Le mode de fonctionnement du commutateur est directement lié à la détermination de la puissance maximale à transférer. Les cycles d'ouverture et de fermeture du commutateur définissent la charge vue par le générateur piézoélectrique.

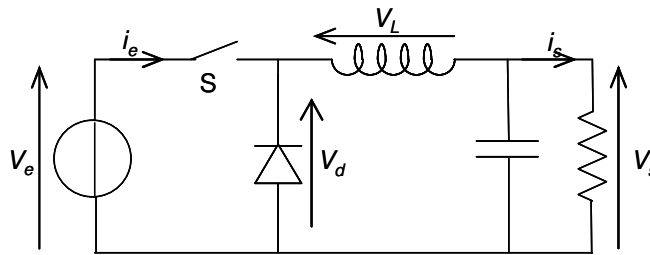


Figure III-38 : Schéma du convertisseur abaisseur de tension (Buck)

### 3.2.1 Principe de fonctionnement du convertisseur DC/DC en mode continu

Dans cette section, nous développons les équations du convertisseur DC/DC. Cette étude nous sert à dimensionner les composants pour une application donnée et de déterminer le rapport cyclique optimum.

Lors du fonctionnement en mode continu du convertisseur DC/DC, le commutateur ou le transistor MOS est fermé sur l'intervalle de temps  $\alpha T$ . A la sortie du convertisseur AC/DC, l'énergie est fournie à l'inductance et à la charge. Pendant la phase de l'ouverture du commutateur, c'est à travers la diode de roue libre que la continuité du courant est assurée.

La fonction de transfert du circuit DC/DC fonctionnant en mode continu est :

$$\frac{V_s}{V_e} = \alpha \quad \text{III-41}$$

Les expressions de la tension et du courant de l'inductance sont données ci-dessous :

$$\begin{aligned} v_L(t) &= L \frac{di(t)}{dt} = v_e(t) - v_s(t) \\ i_L(t) &= \frac{v_e(t) - v_s(t)}{L} t + I_m \end{aligned} \quad \text{III-42}$$

Pendant la deuxième phase  $\alpha T < t < T$ , le commutateur est ouvert, et c'est à travers la diode de roue libre que le courant circule. Dans ce cas, l'inductance restitue le courant accumulé pendant la phase

précédente pour continuer à alimenter la charge. L'ensemble des équations correspondantes est donné par III-43.

$$\begin{aligned}
 v_L &= -v_s(t) \\
 L \frac{di(t)}{dt} &= -v_s(t) \\
 i_L(t) &= \frac{v_s(t)}{L}(\alpha T - t) + I_M
 \end{aligned}
 \tag{III-43}$$

### • Ondulations du courant $i_L$ et de la tension $v_c$

L'ondulation du courant traversant  $i_L$  l'inductance est déterminée lorsque la tension dans cette dernière est nulle. Ceci implique  $\Delta i_L = \Delta i_L^{\alpha T} = -\Delta i_L^{(1-\alpha)T}$  d'où l'expression ci-dessous. Cette ondulation est maximale pour  $\alpha=0,5$ .

$$\Delta I = I_M - I_m = \frac{v_e(\alpha T) - v_s(\alpha T)}{L} \alpha T = -\frac{v_s(T)}{L} T(1-\alpha)
 \tag{III-44}$$

Dans la Figure III-39 nous reportons l'allure de la tension et du courant aux bornes de l'inductance.

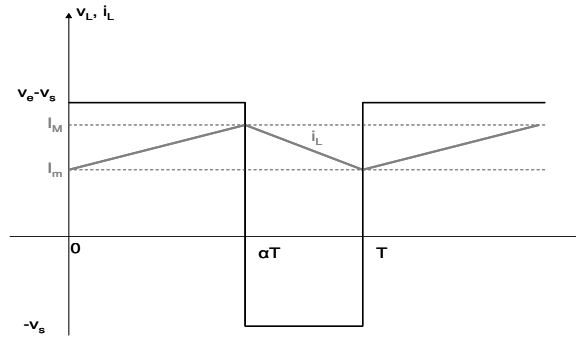


Figure III-39 : Signaux  $v_L$  et  $i_L$  de l'abaisseur de tension DC/DC

En régime permanent, la tension de sortie est proportionnelle à la tension d'entrée  $v_s = \alpha v_e$  (avec par définition  $0 \leq \alpha \leq 1$  pour l'abaisseur de tension). Ceci est dû au fait que l'ondulation de la tension de sortie  $\delta v_s$  est négligeable devant sa valeur moyenne  $V_s$ , et que la valeur moyenne de la tension de l'inductance est nulle. Nous obtenons, ainsi, l'expression de l'ondulation du courant :

$$\Delta I = \frac{v_e}{L} T(1-\alpha)\alpha
 \tag{III-45}$$

L'ondulation de la tension aux bornes de la capacité est égale à la surface du triangle formé par l'ondulation de  $i_L$  (voir la Figure III-40). Ainsi, à partir de l'expression du courant dans la capacité, la quantité de charge et l'ondulation de la tension peuvent être exprimé comme suit :

$$i_c = \delta I_L = C \frac{d(\delta v_s)}{dt}$$

$$Q_C = \frac{1}{2} \frac{\Delta I_L}{2} \frac{T}{2}$$

III-46

$$\Delta V_c = \frac{Q_C}{C} = \frac{\alpha(1-\alpha)V_e}{8LCf^2}$$

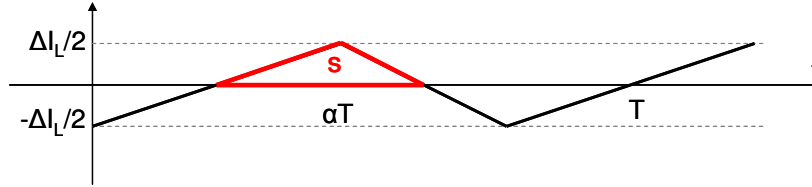


Figure III-40 : Forme de l'ondulation du courant dans l'inductance

### • Choix de L et de C

Les valeurs optimales de l'inductance et de la capacité peuvent être déterminées selon III-47 et III-48 pour  $\alpha=0,5$ .

$$L \geq \frac{T v_e}{4 \Delta I_{Lmax}} \alpha(1-\alpha) \quad \text{III-47}$$

$$C \geq \frac{v_e}{8 L f^2 \Delta V_s} \alpha(1-\alpha) \quad \text{III-48}$$

Le dimensionnement des composants du convertisseur permet l'obtention d'un bon rendement du convertisseur DC/DC en termes de transfert du maximum de puissance. Par ailleurs, les valeurs maximales de l'ondulation pour le courant et la tension sont liées aux conditions suivantes :

- Une faible valeur de l'ondulation conduit à de grandes valeurs de  $L$  et  $C$
- Une valeur trop élevée de l'ondulation augmente le courant dans les composants non linéaires (transistors et diodes).

### 3.2.2 Fonctionnement en mode discontinu

Le mode de fonctionnement discontinu est intéressant pour les applications qui nécessitent une certaine rapidité de réponse à la variation de la charge et/ou de la tension d'entrée. Par contre, ce mode entraîne un courant important dans l'inductance, et une capacité élevée pour le filtrage en sortie.

La fonction de transfert du convertisseur fonctionnant en mode discontinu est :

$$\frac{v_s}{v_e} = \frac{\alpha}{\alpha + \alpha'} \quad \text{III-49}$$

Le courant dans l'inductance est déterminé par les expressions suivantes :

$$\begin{aligned}
i_L(t) &= \frac{v_e - v_s}{L} t & 0 < t < \alpha T \\
i_L(t) &= I_M - \frac{v_s}{L} (t - \alpha T) & \alpha T < t < \alpha' T \\
i_L(t) &= 0 & \alpha' T < t < T \\
I_M &= \frac{v_e - v_s}{L} \alpha T = \frac{v_s}{L} \alpha' T
\end{aligned} \tag{III-50}$$

Le courant à l'entrée du convertisseur est donné par :

$$\langle i_e \rangle = \frac{\alpha}{2} I_M \tag{III-51}$$

Le courant en sortie du convertisseur est donné par :

$$\langle i_L \rangle = \frac{I_M}{2} (\alpha + \alpha') = \frac{\alpha^2}{2Lf_s} \frac{v_e}{v_s} (v_e - v_s) \tag{III-52}$$

Les formes d'onde des différents signaux sont représentées dans la figure ci-dessous.

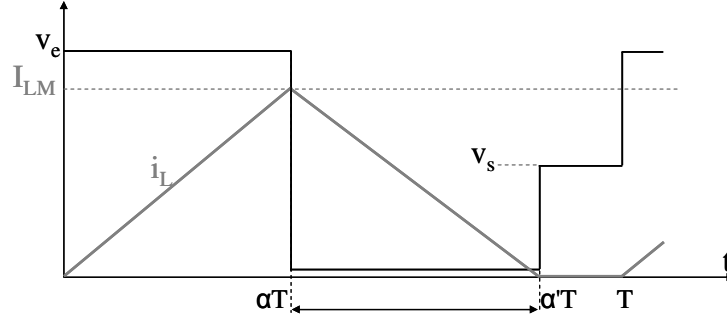


Figure III-41 : La forme d'onde des différents signaux buck DCM

### • Choix de L et de C

En mode discontinu, les valeurs de l'inductance et de la capacité sont données par les expressions suivantes :

$$I_M = \frac{v_i - v_o}{L} \alpha T \quad L = \frac{v_i - v_o}{I_M f_s} \alpha \tag{III-53}$$

$$\Delta v_o = \frac{\Delta Q}{C} = \frac{(I_M - i_o)^2 t_l}{2CI_M} \frac{v_i}{v_o} \quad C = \frac{(I_M - i_o)^2 \alpha}{2I_M f_s \Delta v_o} \cdot \frac{v_i}{v_o} \tag{III-54}$$

### • Exemple de calcul du rapport cyclique en mode discontinu

Pour transférer la puissance maximale vers la charge, des commutateurs sont intégrés dans le sous-système DC/DC. Ceci, s'explique par la charge induite lorsqu'un commutateur est ouvert ou fermé. Ainsi,

pour calculer un cycle d'ouverture et de fermeture d'un commutateur appelé le *rapport cyclique*, nous utilisons les expressions analytiques des sections 3.1 et 3.2.1. Pour ce calcul, nous supposons que le modèle de la charge est constitué d'une résistance d'une faible valeur  $R_b$  et une source de tension nominale  $V_b$  (voir la Figure III-42).

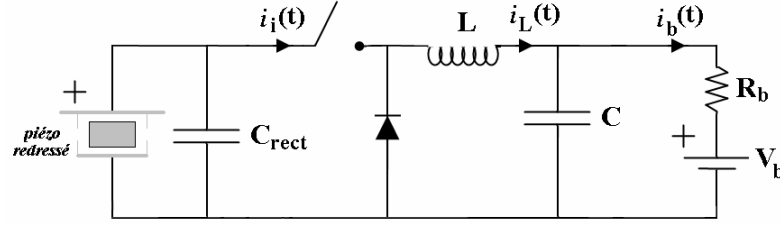


Figure III-42 : Modèle étudié pour calculer le rapport cyclique

En effet, la tension aux bornes de la batterie est proportionnelle à la tension redressée. Le courant dans la batterie est égal au courant moyen dans l'inductance  $L$  pendant l'intervalle de temps  $0 < t < \alpha T$  ( $I_b = \langle i_L \rangle = \langle \frac{V_{rect} - V_b}{L} t \rangle$ ). Donc, l'expression du courant peut être écrite comme suit :

$$I_b = \frac{\alpha^2}{2Lf_s} (V_{rect} - V_b) \quad \text{III-55}$$

Si on considère que l'énergie est conservée dans les circuits AC/DC et DC/DC; le courant dans la batterie et le courant moyen redressé ont les expressions ci-dessous:

$$I_b = \frac{V_{rect}}{V_b} I_{rect} \quad \text{III-56}$$

$$I_{rect} = \frac{2I_p}{\pi} - \frac{2V_{rect}\omega C_p}{\pi}$$

A partir des équations III-55 et III-56, nous pouvons déduire l'expression du rapport cyclique optimal pour le mode discontinu.

$$\alpha = \sqrt{\frac{2Lf_s \left( \frac{2I_p - 2V_{rect}\omega C_p}{\pi} \right)}{(V_{rect} - V_b)}} \Rightarrow \alpha = \sqrt{\frac{4Lf_s\omega C_p V_{rect}}{\pi(V_{rect} - V_b)}} \quad \text{III-57}$$

Les caractéristiques des composants linéaires  $L$  et  $C$  sont reportées sur la Figure III-43 et les valeurs optimales peuvent être prélevées. Ces courbes sont obtenues pour l'exemple de valeurs suivantes [Ott-02] :  $I_p = 3,1\text{mA}$ ,  $C_{piezo} = 0,184\mu\text{F}$ ,  $f_{piezo} = 53\text{Hz}$ ,  $f_{MOS} = 100\text{kHz}$ ,  $V_b = 3\text{V}$ ,  $R_b = 0,1\Omega$ . Pour cette configuration, les valeurs retenues sont  $L = 22\text{mH}$  et  $C = 3,4\text{mF}$ .

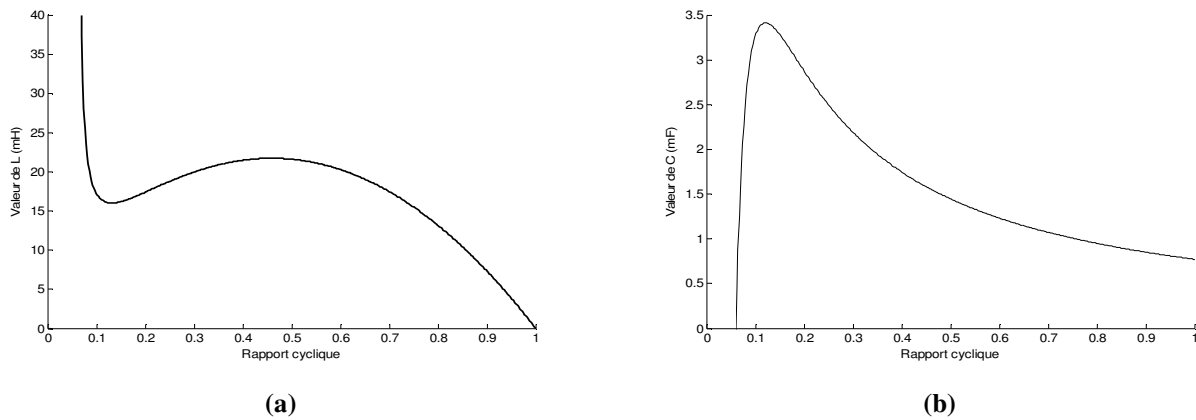


Figure III-43 : Caractérisation de l'inductance (a) et de la capacité (b) du DC/DC en mode continu

### 3.2.3 Modèle du convertisseur DC/DC

Le modèle du convertisseur DC/DC est réalisé avec la bibliothèque de SimPowerSystems.

En réalité, les propriétés de l'inductance dépendent des propriétés magnétiques, dont l'effet ne peut être négligé. Les principaux problèmes sont liés à des phénomènes d'hystérésis et de saturation de l'inductance. Ces propriétés ne sont pas prises en considération dans le modèle simple disponible dans la bibliothèque. L'inductance sera alors considérée comme idéale dans notre implémentation.

Le transistor MOS constitue une source de courant à deux états *on* et *off* dans le convertisseur DC/DC. Le modèle du transistor MOS est un macro modèle existant dans la bibliothèque SimPowerSystems. Ce modèle possède quatre ports : le drain, la source et la grille en plus d'un port destiné à la mesure du courant et de la tension de la diode entre le drain et la source.

Le signal du drain est lié à la tension de sortie de l'AC/DC. Le transistor MOS est contrôlé par un signal numérique appliqué sur sa grille. Ce signal de contrôle provient d'un modulateur de largeur d'impulsions (PWM<sup>36</sup>) qui lui-même est lié au signal du rapport cyclique<sup>37</sup> (cf. section 3.4)

Pour valider le premier prototype virtuel du convertisseur avec Simulink/SimPowerSystems, le circuit est connecté à une charge résistive même si dans la réalité, la charge doit avoir un modèle plus complexe (cf. section 4). La charge résistive permet une première étude du comportement du microsysteme. Ce premier prototype permet de définir les différents paramètres des composants et de résoudre les problèmes dus au non calibrage des composants dans le modèle.

#### • Evaluation du modèle

Afin de valider le modèle du convertisseur DC/DC, nous avons procédé à l'implémentation et à l'exécution des simulations sous Simulink. Les composants linéaires et non linéaires ainsi que les

<sup>36</sup> Pulse Width Modulation

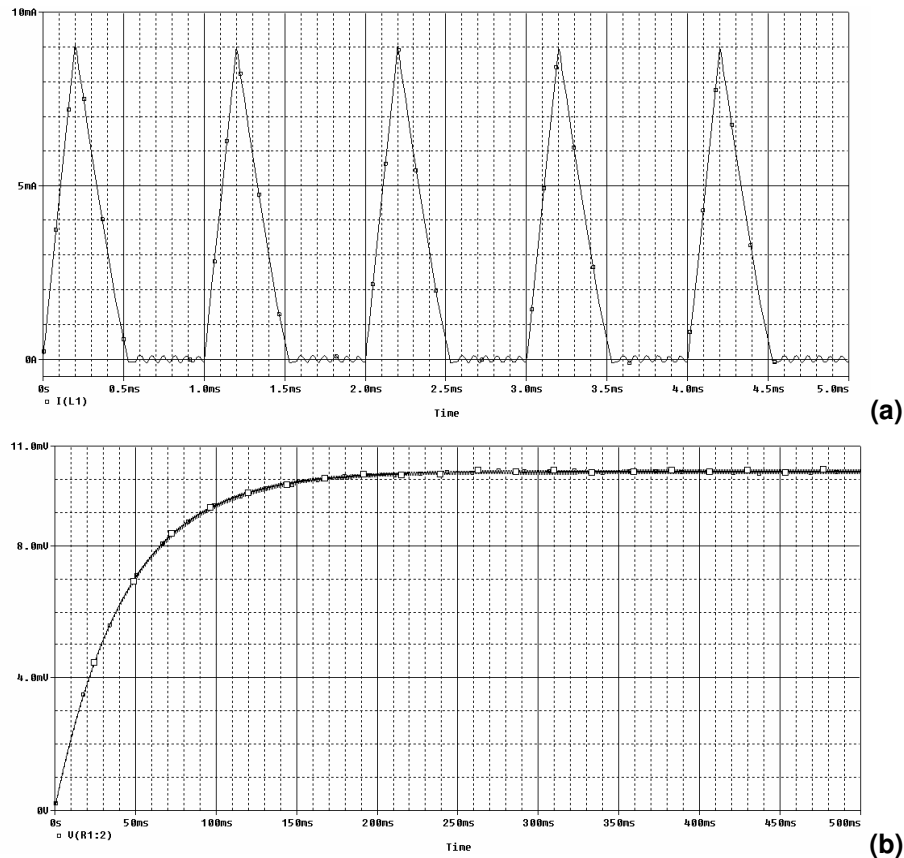
<sup>37</sup> Duty cycle en anglais

paramètres du DC/DC sont présentés dans le Tableau III-5. Cette configuration nous permet de comparer le modèle de l'abaisseur de tension construit sous Simulink avec le modèle sous SPICE.

Composant/paramètre	Valeur
Transistor	IRF150
Diode	D1N4002
$V_e$	1V
L	22mH
C	10m $\Omega$
$R_{load}$	4,5 $\Omega$

**Tableau III-5 : Composants linéaires utilisés**

La Figure III-44 illustre les courbes de la simulation sous SPICE qui sont la référence pour valider le modèle avec SimPowerSystems. Le courant maximal dans l'inductance  $i_L$  est égal à 9mA, et la tension aux bornes de la résistance de charge en régime permanent  $V_{out}$  est égale à 10,4mV.



**Figure III-44 : (a) courbes de  $i_L$ , (b)  $v_{out}$  sous SPICE**

Le modèle de la Figure III-45 est complètement construit avec les éléments de SimPowerSystems sous Simulink.



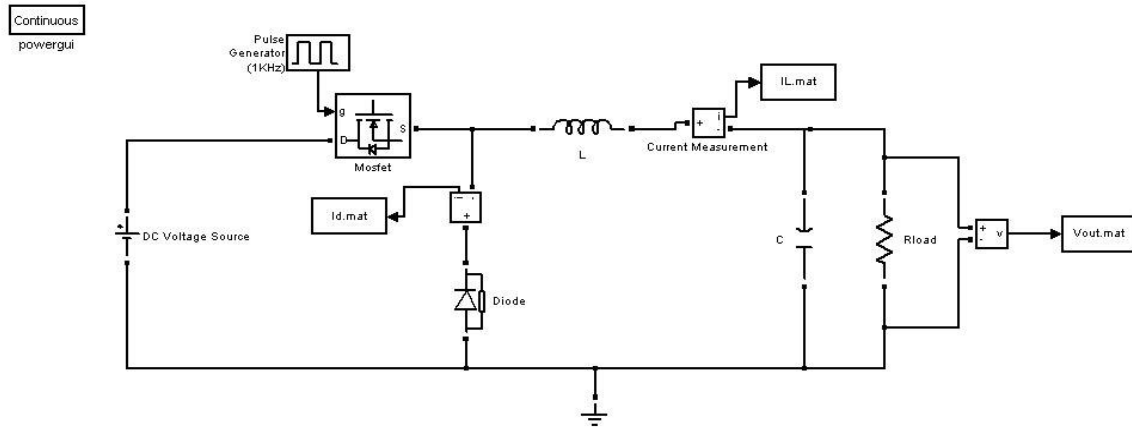
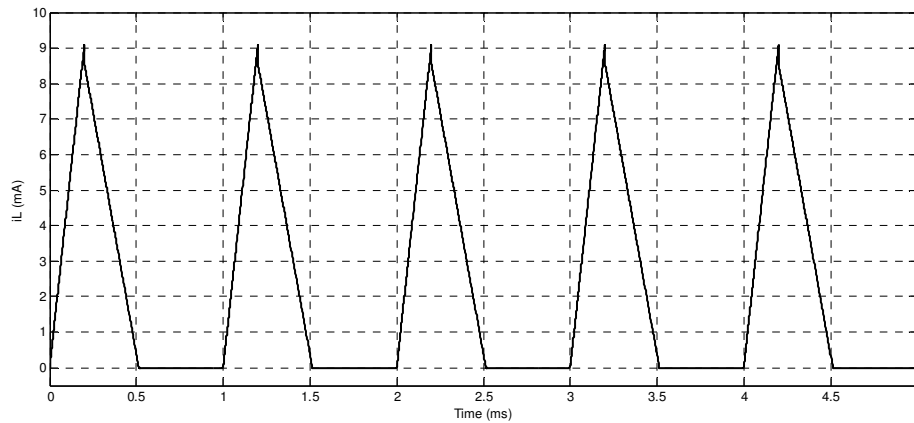


Figure III-45 : Schéma du DC/DC sous Simulink

La Figure III-46 illustre la courbe du courant  $i_L$  dans l'inductance obtenu par le modèle SimPowerSystems. Les paramètres des composants non linéaires sont ceux utilisés dans le modèle SPICE. Les modèles des blocs non linéaires de SimPowerSystems comportent des circuits de protection  $R_s C_s$ <sup>38</sup> appelé "snubber". Nous constatons qu'en faisant abstraction des effets de ces composants de protection ( $R_s = \text{inf}$  et  $C_s = 0$ ), les ondulations en hautes fréquences qui sont dues aux capacités de jonction et aux résistances internes du transistor MOS sont absentes. Toutefois, nous obtenons la même allure du courant à travers l'inductance que sous SPICE (se référer à la figure ci-dessous).

Figure III-46 : Courbe de  $i_L$  sans l'effet "snubber"

Cette courbe montre que le fonctionnement du DC/DC est en mode discontinu.

<sup>38</sup> En anglais : snubber circuit

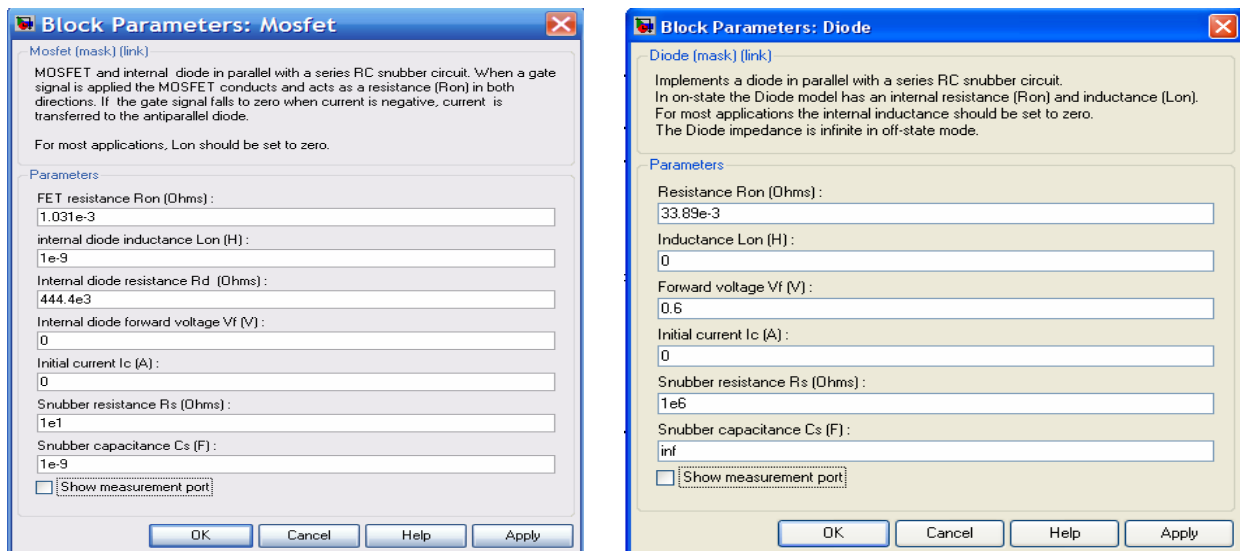
## • Introduction des effets "snubber"

L'effet "snubber" est modélisé par un circuit série  $R_s C_s$  connecté en parallèle dans les modèles des composants non linéaires. Ce circuit est utilisé dans le domaine de l'électronique de puissance pour amortir les oscillations parasites et est considéré comme un dispositif de protection. Ces circuits sont intégrés dans les modèles des composants non linéaires de SimPowerSystems, qui, à la base, sont destinés aux applications de l'électronique de puissance.

Les paramètres utilisés pour le transistor MOS et la diode de roue libre sont donnés dans les blocs des paramètres de la Figure III-47-a. Le circuit "snubber" absorbe les pics de tension dus aux transitions lors de la commutation ce qui pourrait constituer une solution pour les systèmes qui présentent des problèmes de convergences. Par ailleurs, l'introduction des effets "snubber" réduit la vitesse des simulations mais donne des résultats précis.

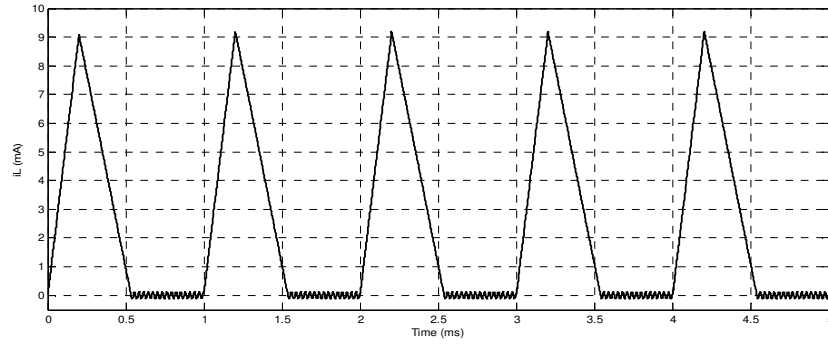
Dans le cas de l'abaisseur de tension, le transistor MOS est une source de courant qui alimente une inductance. Les variations rapides du courant traversant une inductance peuvent générer des pics. Ainsi, ce problème peut être résolu par l'introduction des effets "snubber" dans le modèle ou par la mise en parallèle de l'inductance une charge résistive suffisamment grande pour être négligée devant les autres composants du circuit DC/DC.

Nous avons introduit ces effets « snubber » dans le modèle avec SimPowerSystems en utilisant  $R_{snubber}=10\Omega$  et  $C_{snubber}=1nF$ . Sur la Figure III-47-b & c, nous reportons les courbes du courant  $i_L$  et la tension  $v_{out}$ . Ces courbes sont qualitativement proches des résultats obtenus avec le modèle sous SPICE<sup>39</sup>, où les ondulations ne sont pas amorties.

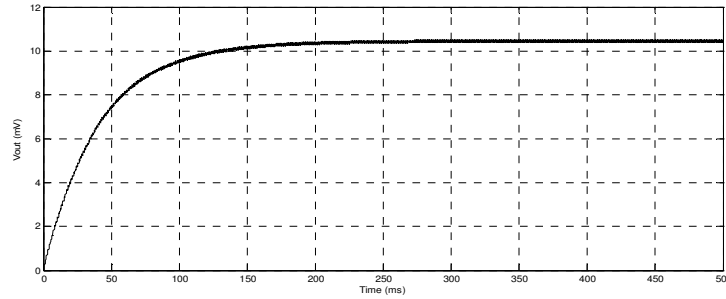


(a)

<sup>39</sup>  $R_{on}=R_{drain}$ ,  $R_d=R_{ds}$ .



(b)



(c)

**Figure III-47 : (a) Paramètres du transistor MOS et de la diode, (b)  $i_L$ , (c)  $v_{out}$  avec SimPowerSystems**

### 3.2.4 Modèles raffinés des composants non linéaires

Les éléments de SimPowerSystems sont destinés à la base pour la modélisation des circuits de l'électronique de puissance. Les modèles SPICE sont mieux adaptés aux composants microélectronique. Ainsi, le raffinement des composants non linéaires des étages AC/DC et DC/DC en utilisant la description des modèles sous SPICE est considéré comme une étape intégrante de la méthodologie de modélisation collaborative. En effet, les modèles du transistor MOS et de la diode ont été développés par un stagiaire au sein du groupe et sont implémentés sous Simulink.

La Figure III-48 illustre le modèle au niveau circuit de la diode sous la forme d'un schéma électrique. Pour ce modèle, une fonction MATLAB intégrée<sup>40</sup> est formulée avec les expressions analytiques de la diode. Cette fonction enveloppe le calcul du courant  $I_d$  traversant la diode qui dépend des tensions de l'anode  $V_a$  et de la cathode  $V_c$ , ainsi que des phénomènes physiques. Ces phénomènes sont liés aux capacités de jonction et de diffusion représentées sur la figure ci-dessous par la capacité équivalente  $C_{eq}$ , la résistance interne  $R_s$  et de la température (voir les détails en Annexe G).

<sup>40</sup> Embedded MATLAB Function

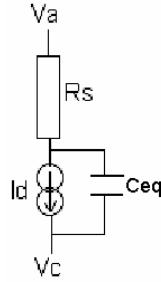


Figure III-48 : Schéma électrique équivalent d'une diode

De la même façon, la description du circuit du transistor MOS est réalisée selon le modèle de SPICE *niveau 1*. Le courant traversant le canal est calculé par les expressions analytiques en fonction des tensions appliquées à la grille, à la source et au drain. La Figure III-49 montre le schéma équivalent électrique du transistor MOS. Les phénomènes physiques (les pertes ohmiques, les effets capacitifs et les courants de fuites) sont aussi inclus dans le modèle. Ces phénomènes concernent les pertes ohmiques qui sont représentées par les résistances  $R_d$ ,  $R_g$ ,  $R_b$ ,  $R_s$ , les effets capacitifs entre source, drain, grille et substrat qui sont représentés par les paramètres  $C_{GS}$ ,  $C_{GD}$ ,  $C_{BD}$ ,  $C_{BS}$ , et enfin les courants  $I_{BD}$ ,  $I_{BS}$ .

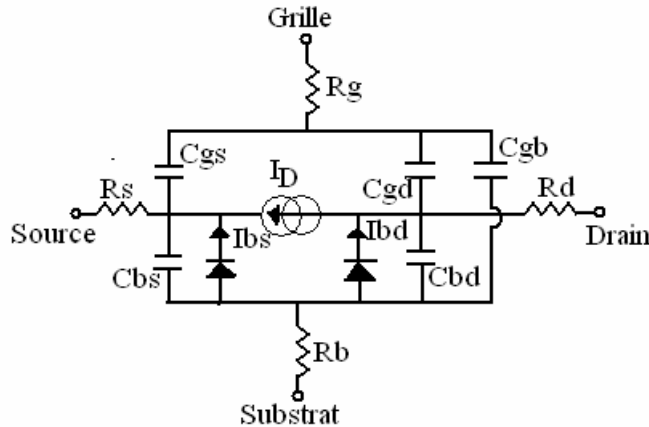


Figure III-49 : Schéma électrique équivalent du transistor MOS

Le modèle du transistor MOS est construit avec une fonction MATLAB intégrée qui calcule le courant du drain  $I_D$  à partir des lois de Kirchhoff (Figure III-49). L'implémentation détaillée des équations et le modèle sous Simulink sont donnés en Annexe H.

### 3.2.5 Evaluation du modèle raffiné

Le convertisseur DC/DC (voir la Figure III-50) au niveau circuit est réalisé avec les modèles raffinés en *BSIM niveau1* des blocs non linéaires. L'implémentation du DC/DC est réalisée avec SimPowerSystems pour les composants linéaires. Nous rappelons que pour les composants non linéaires, leurs fonctions sont intégrées sous la forme de Fonctions MATLAB Intégrées dont les masques sont aussi

réalisés pour permettre d'introduire les paramètres physiques dépendant de la technologie. Les interfaces entre les différents composants sont intégrées pour convertir les signaux Simulink en leurs équivalents électriques. Cette approche permet de construire des modèles utilisateurs qui peuvent être plus précis.

Cependant, l'exécution du modèle ci-dessous sous Simulink donne une simulation plus lente. Nous avons constaté que pour un temps de simulation de  $2ms$ , la durée est de  $9s$  alors que le modèle au niveau comportemental (composants linéaires et non linéaires en SimPowerSystems) est plus rapide.

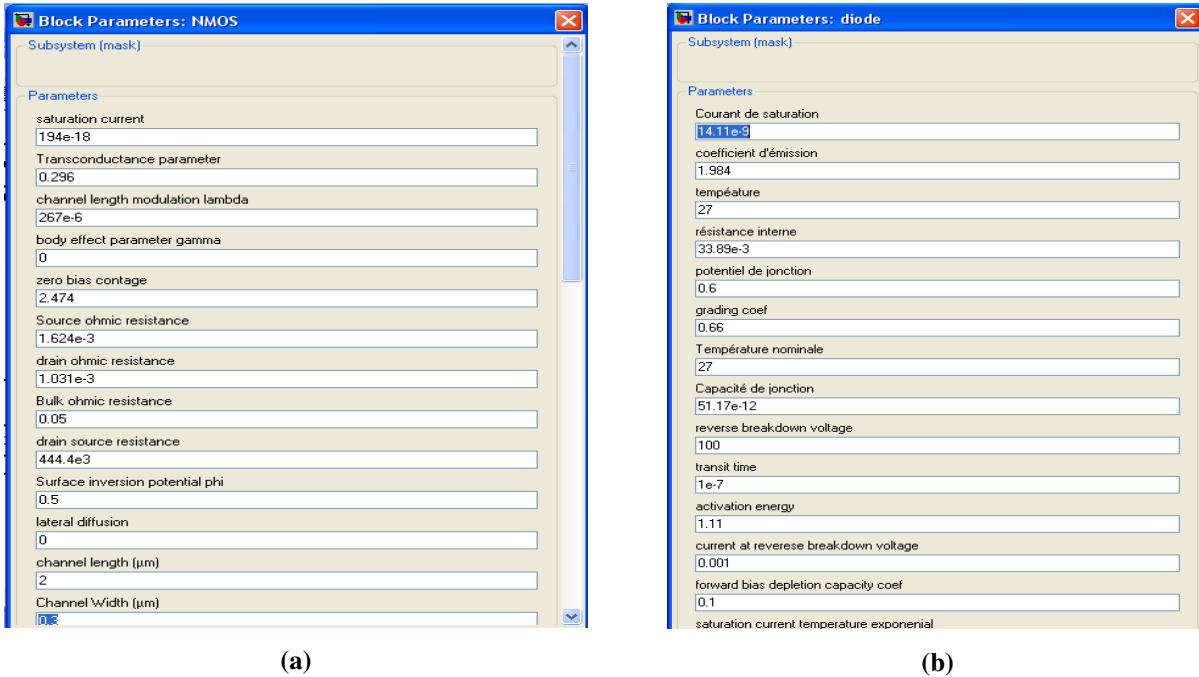


Figure III-50 : Paramètres du modèle raffiné (a) du MOSFET, (b) de la diode

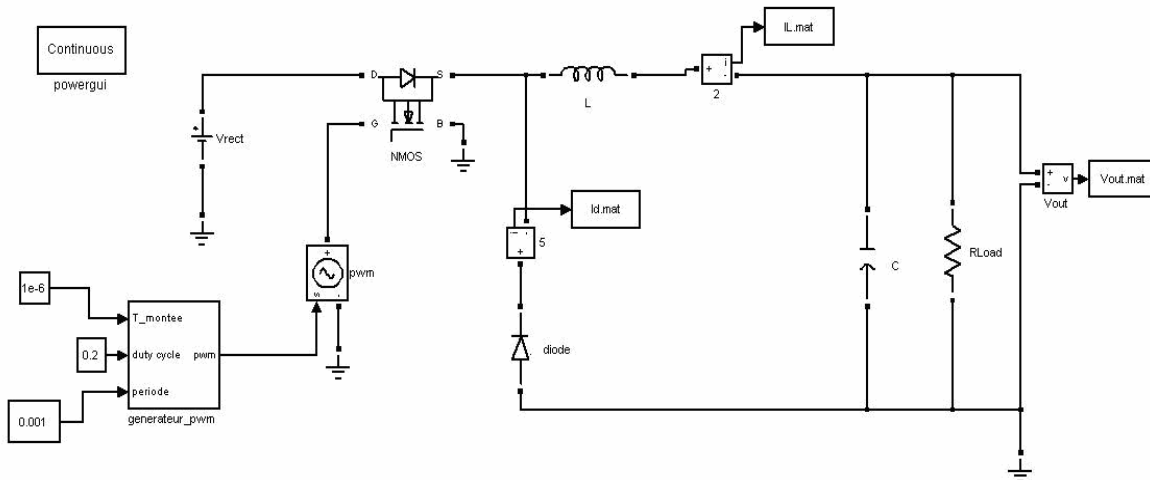


Figure III-51 : Implémentation du modèle raffiné du DC/DC avec MATLAB:Simulink/SimPowerSystems

La Figure III-52 représente une comparaison qualitative et aussi quantitative entre les deux modèles au niveau comportemental et circuit. Un décalage entre les deux courbes est à noter. Ceci est dû aux détails intégrés dans le modèle raffiné et qui n'existe pas dans le modèle comportemental.

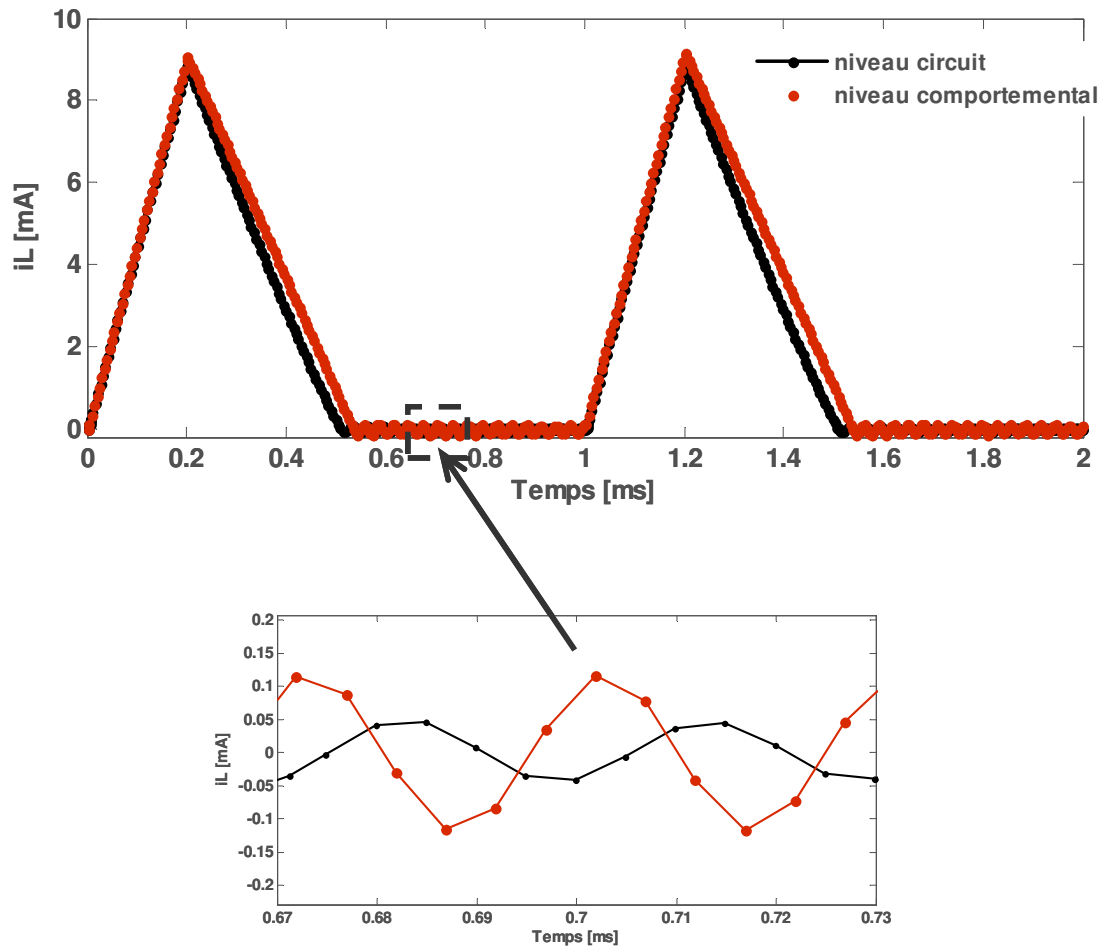


Figure III-52 : Comparaison des courbes de  $i_L$  obtenue avec le modèle du DC/DC au niveau circuit

### 3.3 Génération de la commande du commutateur

Nous rappelons que le transfert de la puissance à la charge est maximal si le générateur piézoélectrique a à ses bornes une charge optimale. Le signal de commande du transistor MOS est un train d'impulsions de largeurs variables. Ainsi, comme son nom l'indique, le modulateur de largeur d'impulsions (PWM)<sup>41</sup> est un circuit qui génère un signal carré avec un rapport cyclique variable. Le signal de sortie de ce circuit est l'intersection d'un signal triangulaire ou en dents de scie avec un signal de référence. Lorsque le signal de référence est supérieur au signal de modulation, la largeur du MLI est petite, et grande dans le cas inverse (voir la Figure III-53-a). Ce signal est utilisé pour commander le transistor

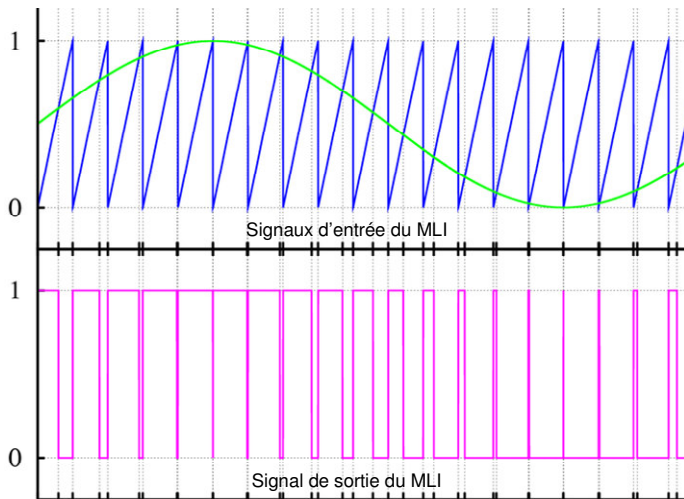
<sup>41</sup> PWM : -de l'anglais Pulse Width Modulator

MOS dans le convertisseur DC/DC et permet de réguler la puissance délivrée par le circuit AC/DC en amont de la charge.

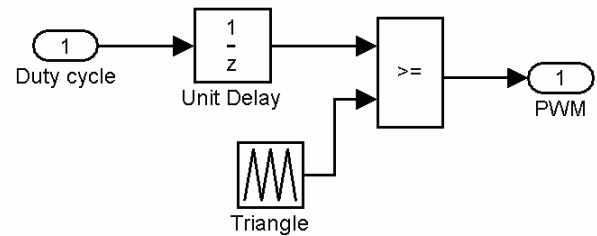
L'implémentation de ce sous-système est réalisée selon deux modèles. Le premier modèle de base consiste à comparer le signal du rapport cyclique issu de la partie numérique (cf. section 3.4) au signal triangulaire (voir la Figure III-53-b). L'implémentation du modèle est effectuée avec les éléments de base existant dans la bibliothèque de Simulink.

Le deuxième modèle est une description détaillée des états du signal MLI avec un temps de montée donné. Ce modèle est implémenté sous la forme d'une fonction incorporée, qui est intégrée dans le circuit DC/DC de la section 3.2.4 (voir la Figure III-53-c). Cette réalisation permet de définir un modèle utilisateur du MLI qui peut être modifié pour un temps de montée et un rapport cyclique donnés.

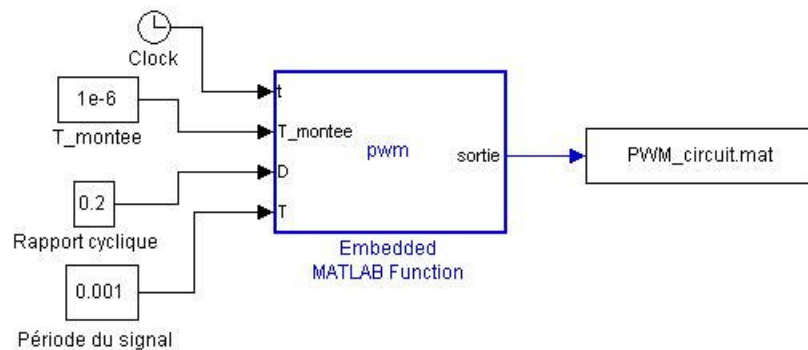
Notons que le rapport cyclique utilisé pour générer le signal est issu de la partie numérique que nous développons dans la section suivante.



(a)



(b)



(c)

**Figure III-53 : (a) Génération du signal MLI, (b) Modèle de base, (c) Modèle analytique**

### 3.4 Implémentation du bloc numérique

Nous rappelons que le microsysteme autonome contient une partie numérique destinée au contrôle et au maintien de la puissance maximale dans la charge [Ott-02]. Ce circuit est destiné à réguler la tension rectifiée issue du transducteur piézoélectrique qui alimente la batterie. Ainsi, le transfert du maximum de puissance correspond à maximiser le courant dans la batterie  $I_{bat}$ , puisque la tension à ses bornes change lentement. Cette opération est soutenue par l'utilisation d'un circuit qui contrôle le rapport cyclique en fonction du courant selon la Figure III-55. Il s'agit de trouver la valeur optimale du rapport cyclique qui dépend du courant du transducteur piézoélectrique  $I_p$ . D'autre part, l'implémentation de la fonction de calcul du rapport cyclique optimal passe par un algorithme qui s'adapte aux variations instantanées du courant dans la charge. Dans la réalité, ce circuit est de type : DSP, FPGA, ou ASIC et donc traite avec des signaux numériques. Ce circuit est connecté au modulateur des impulsions afin de générer le signal de commande du transistor MOS.

Les formules du rapport cyclique pour les modes continu et discontinu sont données par III-58 et III-59 respectivement.

$$\alpha = \sqrt{\frac{4Lf_s \omega C_p V_{rect}^2}{\pi V_b (V_{rect} - V_b)}} \quad III-58$$

$$\alpha = \sqrt{\frac{4Lf_s \omega C_p V_{rect}}{\pi (V_{rect} - V_b)}} \quad III-59$$

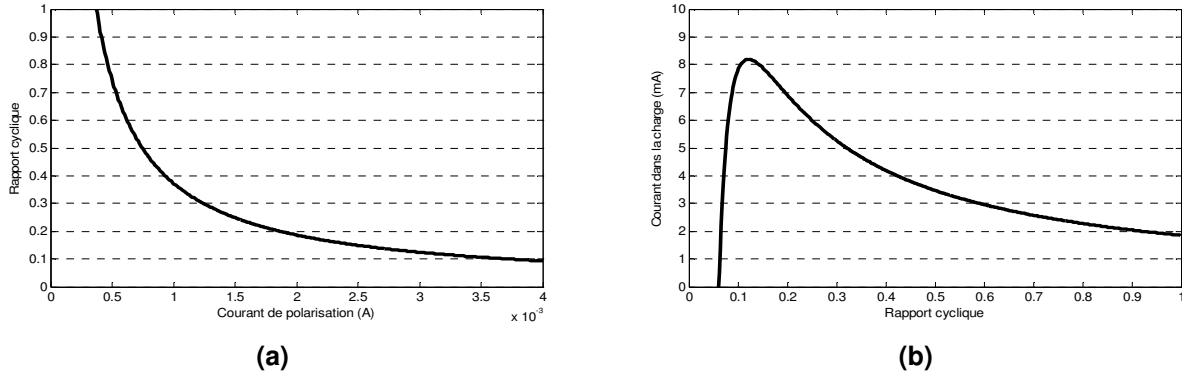
Nous avons déterminé analytiquement la valeur optimale du rapport cyclique pour les paramètres du tableau ci-dessous.

$I_p$ (mA)	$C_{piezo}$ ( $\mu F$ )	$f_p$ (Hz)	$f_s$ (Hz)	$V_{rect}$ (V)	$V_{bat}$ (V)	$L$ (mH)
3.1	0.184	53	1000	20	3	22

**Tableau III-6 : Valeurs utilisées pour le calcul du rapport cyclique**

Nous reportons sur la Figure III-54-a l'exemple en mode continu de la courbe du rapport cyclique en fonction du courant du transducteur piézoélectrique. Nous pouvons prélever la valeur du rapport cyclique optimal pour  $I_p=3.1mA$  à 12%. A cette valeur du rapport cyclique est associée la valeur maximale du courant dans la batterie comme le montre la Figure III-54-b.



Figure III-54 : (a)  $\alpha=f(I_p)$ , (b)  $I_b=f(\alpha)$ 

### • En diagramme blocs

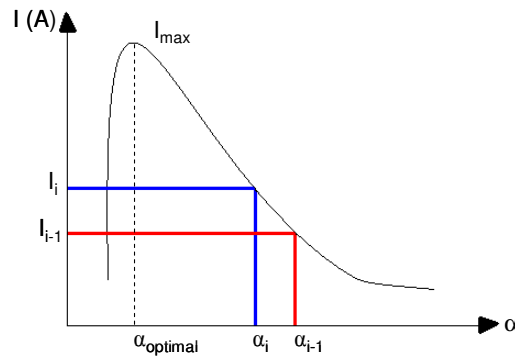
Dans les travaux d'Ottman & al. [Ott-02], un algorithme a été proposé afin de déterminer le rapport cyclique optimal. Cet algorithme compare deux valeurs successives du courant échantillonné et additionne ou soustrait la valeur du pas de changement  $k$  à l'ancienne valeur du rapport cyclique.

$$\frac{\partial I_b}{\partial \alpha} \neq 0 \Rightarrow \alpha_{i+1} = \alpha_i + k \operatorname{sgn}\left(\frac{\partial I_b}{\partial \alpha}\right)$$

$$\frac{\partial I_b}{\partial \alpha} = 0 \Rightarrow \alpha_{i+1} = \alpha_i$$

III-60

Le changement du signe du rapport de  $I_b$  à  $\alpha$  permet d'augmenter ou de diminuer la nouvelle valeur de  $\alpha$ .

Figure III-55 : Courbe théorique du courant dans la batterie  $I_{bat} = f(\alpha)$ 

Le système ci-dessus est un modèle continu de type algorithme qui peut être implanté dans un circuit cible. La cible est un circuit numérique d'où la nécessité de discrétiser les opérations et les signaux. L'implantation réelle se fait par la génération de code C très générique en utilisant le Real-Time Workshop (RTW) et un banc de test spécifique aux DSP.

L'implémentation de l'algorithme de contrôle est effectuée avec les éléments de la librairie Simulink comme le montre la Figure III-56. La valeur du rapport cyclique est évaluée en fonction du sens du

courant dans la batterie (s'il est croissant ou décroissant) et l'ancienne valeur du rapport. Le bloc de retard sert à discrétiser le signal d'entrée. Un nombre d'échantillons de ce dernier sont accumulés et comparés avec les prochains échantillons. Les dérivées numériques du signal d'entrée et de  $\alpha$  sont calculées puis leur rapport permet de déterminer le signe. Si le signe du rapport est positif ou nul, la nouvelle valeur de  $\alpha$  va augmenter de  $k$  sinon ça va diminuer de  $-k$ . La valeur du taux de changement  $k$  est déterminée de façon à produire des changements du courant dans la batterie suffisamment mesurables afin de calculer la nouvelle valeur du rapport cyclique. Le rapport initial donné au départ facilite la convergence de l'algorithme vers la valeur optimale.

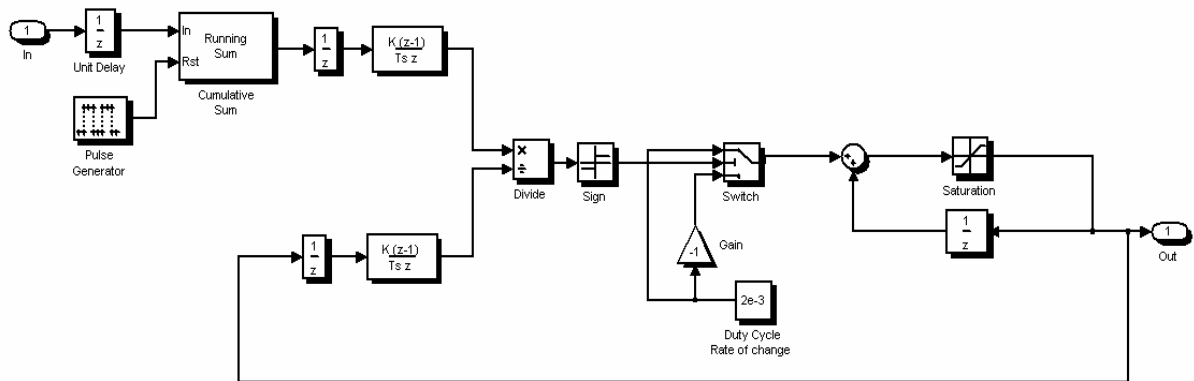


Figure III-56 : Diagramme bloc de l'algorithme de contrôle

Nous avons testé le modèle ci-dessus pour un signal d'entrée sinusoïdal (dans notre application le courant dans la batterie est positif). Nous constatons comme le montre la Figure III-57 que le rapport cyclique augmente avec le signal d'entrée et ensuite garde la même valeur pendant la diminution de ce dernier. Nous pouvons conclure que l'algorithme implémenté en diagramme blocs d'adapte à la variation du signal d'entrée, et donc fonctionne correctement.

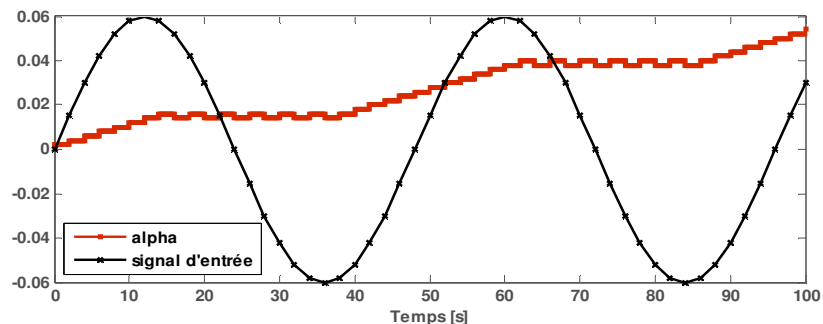


Figure III-57: Réponse de l'algorithme de contrôle à un signal sinusoïdal

## • En code VHDL

Dans le domaine de la microélectronique, les circuits numériques sont développés en langage matériel afin d'être vérifiés puis synthétisés. Alors que nous préconisons une approche de modélisation

collaborative avec des experts dans les différents domaines, l'introduction d'un modèle décrit en code VHDL dans notre flot de modélisation est une phase indispensable. La boîte à outil *Link for ModelSim* disponible sous Simulink nous a intéressé pour modéliser la partie numérique du système. Nous montrerons dans la section B.2 qu'il est possible de réaliser la vérification du comportement du microsystème global en utilisant la cosimulation multi langages MATLAB/Simulink/ModelSim.

Le code VHDL de l'algorithme implémenté est donné en Annexe I. le résultat de simulation de cet algorithme sous ModelSim est donné par la Figure III-58. Le signal  $v_{in}$  est le signal d'entrée qui est une suite de stimuli augmentant, puis diminuant. Le signal de sortie *PWM* est le rapport cyclique  $\alpha$ .

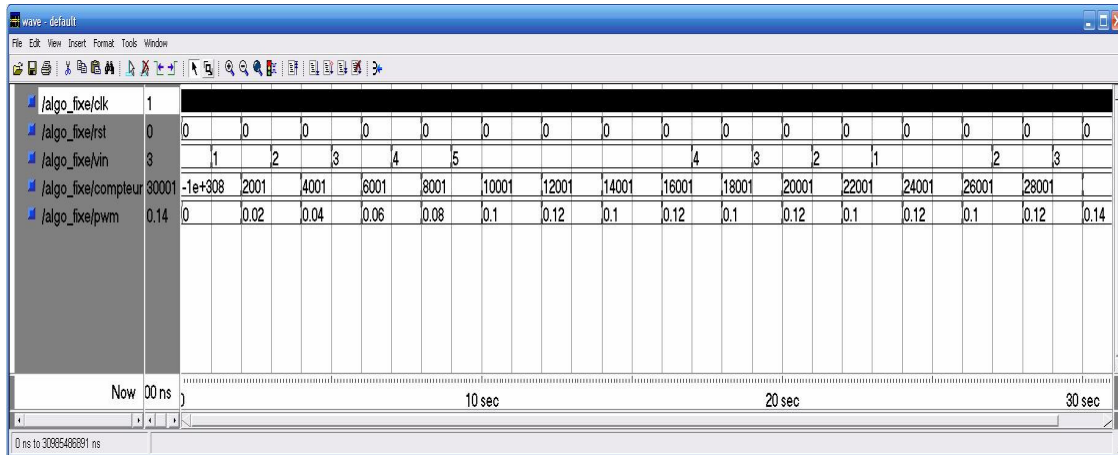


Figure III-58 : Exemple d'exécution de l'algorithme à pas fixe en VHDL

### 3.5 Bilan

Cette partie a porté sur la modélisation des sous-systèmes constituant le circuit de récolte et de gestion de l'énergie dans le microsystème autonome. Nous avons utilisé la toolbox SimPowerSystems pour modéliser les parties électriques et les implémenter sous Simulink. Nous avons vérifié les modèles individuellement. Les composants non linéaires ont été raffinés en utilisant les expressions mathématiques et en les intégrant sous des fonctions MATLAB embarquées.

L'algorithme de contrôle du circuit DC/DC est implémenté de deux façons différentes. La première est sous la forme de diagramme blocs en profitant des éléments de la bibliothèque de Simulink directement. Et la deuxième consiste à vérifier le code VHDL de cette partie numérique sous ModelSim afin de l'intégrer dans le prototype virtuel présenté plus loin dans ce document (cf. B.2).

Le tableau ci-dessous récapitule les différents modèles établis ainsi que les avantages et les inconvénients de cette approche de modélisation des systèmes électriques sous Simulink.

Sous-systèmes		
AC/DC	DC/DC	Algorithme de contrôle
•Description en SimPowerSystems	•Description en SimPowerSystems •Raffinement des modèles des composants non linéaires	•Description en diagramme blocs sous Simulink •Description en code VHDL sous ModelSim
<b>Avantages:</b> ➤ Implémentation intuitive avec les éléments de SimPowerSystems : simulateur électrique ➤ Construction de modèles utilisateurs ➤ Ajustement des composants électriques génériques indépendamment de la technologie et du logiciel  <b>Limitations:</b> ➤ Simulation plus lente en comparaison à un simulateur électrique (SPICE) ➤ Manque de composants prédéfinis ➤ Pas d'étude paramétrable dans le même modèle :développement de routines supplémentaires		<b>Avantages:</b> Implémentation et vérification dans Simulink Possibilité de cosimulation multi langages avec Simulink

Tableau III-7 : Récapitulatif sur la modélisation du circuit de récolte d'énergie

## 4. Elément de stockage : la batterie

Les batteries constituent des éléments essentiels dans le microsysteme sans fil, et ce pour leurs grandes densités énergétiques et leur long cycle de vie. Il existe différentes approches de modélisation des batteries. Ces approches classifient : les modèles analytiques, les modèles électriques, les modèles stochastiques, et les modèles électrochimiques. De plus, ces modèles peuvent être caractérisés par d'autres facteurs liés à la technologie supportée (Alkaline, Lithium-ion, Ni-Cd, Ni-MH...etc.), de la nature du courant, des effets internes (la température, l'état de charge et de décharge capacitif) et de la prédiction du comportement de la batterie durant son cycle de vie.

Dans notre étude, nous nous sommes intéressés à des modèles dynamiques afin d'observer le comportement de la batterie. Nous avons classifié les modèles de la batterie selon deux niveaux d'abstraction : les modèles comportementaux où les phénomènes électrochimique et électrothermiques sont approximés et inclus dans les caractéristiques générales, et les modèles raffinés qui tiennent compte de ces phénomènes. De l'état de l'art sur les batteries, il en sort deux catégories : les modèles équivalents de Thévenin [Gla-96] [Gao-02], et les modèles dynamiques<sup>42</sup> [Hag-93] [Gol-97] [Ben-00] [Che-06]. Dans notre étude nous proposons :

- a. des modèles comportementaux basés sur un circuit électrique équivalent ou sur une description analytique
- b. un modèle raffiné basé sur un circuit électrique où les dépendances sur le débit de la batterie, les effets de la température et les variations non linéaires du potentiel d'équilibre sont modélisés.

### 4.1 Modèle comportemental dynamique

Nous rappelons que dans cette partie, l'étude et le développement des modèles sont au niveau comportemental. De plus, les modèles sont implémentés sous l'environnement MATLAB/Simulink.

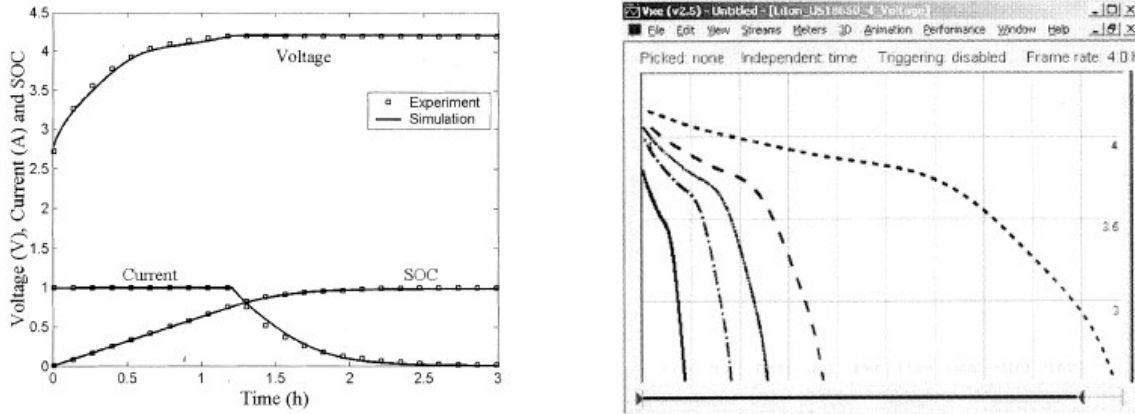
Nous nous sommes intéressés à modéliser une batterie de type *Li-ion*. La modélisation du fonctionnement électrochimique est plus difficile à manipuler et ne nous intéresse pas pour notre application où nous étudions l'évolution de la charge de la batterie dans le temps.

Par ailleurs, nous proposons plusieurs méthodes de modélisations pour une batterie **Sony US18650** avec une capacité nominale de *1,5Ah*, une tension maximale de *4,2V* et une tension minimale de *2,5V*. Cette batterie figure dans plusieurs travaux sur lesquels nous nous sommes basés afin de développer nos modèles.

---

<sup>42</sup> Run-time based model

Dans la littérature, la charge stockée dans une batterie correspond à la tension aux bornes de la capacité  $C=3600 \times C_{nominal}$ .  $C_{nominal}$  est la capacité nominale en  $Ahr$ , dont la valeur est fournie dans les fiches techniques des fabricants. La batterie est aussi caractérisée par la tension à circuit ouvert  $V_{oc}$  en charge ou en décharge. La Figure III-59 est une comparaison entre les résultats expérimentaux et la simulation d'un modèle analytique développé par Gao [Gao-02]. Ces caractéristiques vont constituer une référence pour valider nos modèles de batterie.



**Figure III-59 : Caractéristique de charge et de décharge en fonction du temps d'une batterie Sony 18650 (4,2 V; 1,5 Ah)**

#### 4.1.1 Circuit équivalent électrique

Le comportement de la batterie est équivalent à la tension  $v_{bat}$  mesurée pour le courant  $i_{bat}$  traversant une résistance interne  $R_{bat}$  englobant les dissipations, à partir du potentiel d'équilibre  $V_{oc}$ .

Le premier modèle que nous avons implémenté est le circuit équivalent de Thévenin donné par la Figure III-60-a. Le total de  $R_l$  et  $R_{bat}$  représente la résistance effective de la batterie qui peut être estimée dans les fiches des fabricants ou obtenue par mesures expérimentale. Cette résistance effective est divisée pour pouvoir introduire dans le modèle la capacité effective  $C_l$ . Le circuit  $R_l C_l$  parallèle caractérise la dépendance de la réponse de la batterie aux changements pendant la charge/décharge [Gao-02]. Nous reportons dans le Tableau III-8 les valeurs des paramètres utilisés dans le modèle. Le courant  $i_{bat}$  est un signal continu égal à  $1A$ .

Paramètres	$I_{bat}$ [A]	$V_{oc}$ [V]	$T$ [°C]	$C_{nominal}$ [Ah]	$R_{bat}$ [Ω]	$C_l$ [F]	$R_l$ [Ω]
Valeurs	1	2,5	23	1,5	0,11	4	0,04

**Tableau III-8 : Paramètres utilisés pour le modèle de Thevenin**

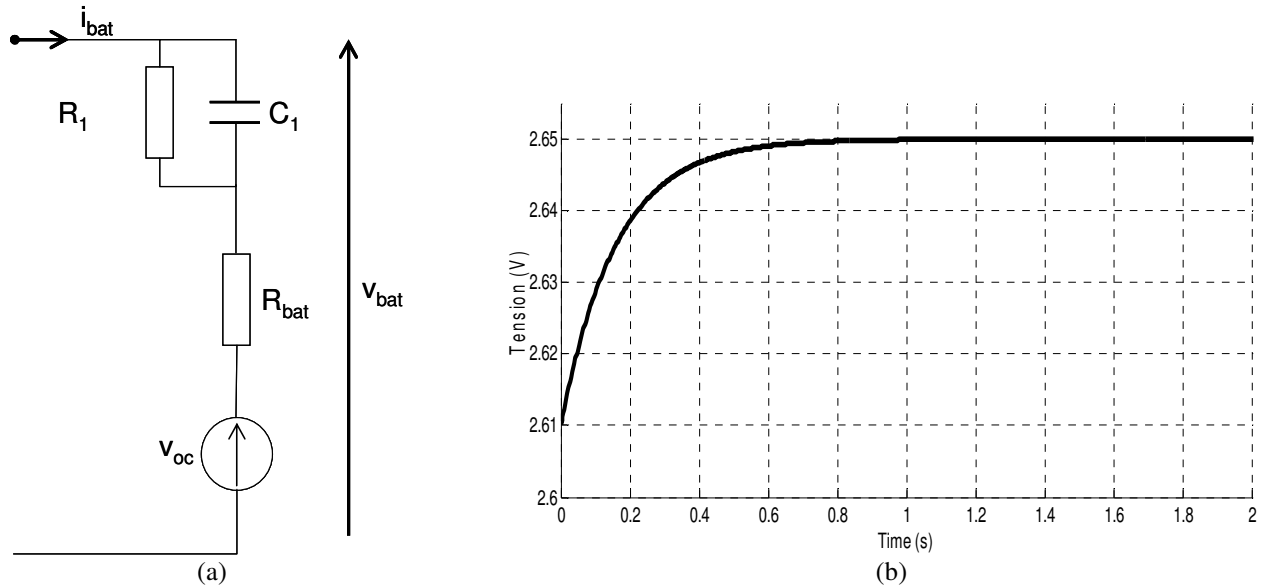


Figure III-60 : (a) Modèle de Thevenin (b) Tension en fonction du temps

La courbe sur la Figure III-60-b représente la réponse de la batterie au courant appliqué. Le temps de réponse à 5% est de 0,48s. La tension maximale aux bornes de la batterie est 2,65V.

Ce modèle est insuffisant et ne décrit qu'une partie du comportement pour la réponse de la batterie en régime transitoire aux changements rapides. D'autres phénomènes doivent être pris en considération [Gao-02].

L'introduction dans le modèle du débit de la batterie conduit à décrire le comportement en fonction du courant injecté. La tension aux bornes de la batterie dans ce cas, dépend de la tension en circuit ouvert  $V_{oc}$ , de l'état de charge initial et du courant appliqué pendant le temps de chargement de la batterie. Le débit est modélisé par un courant proportionnel à une capacité, et est défini par  $C_{rate} = C_{nominal} / t$ . Cette caractéristique de la batterie est définie par le temps que prend une batterie de 1Ah par exemple pour se charger (ex : 10 heures). Dans ce cas, le débit est  $C_{rate} = 1Ah / 10h$ .

La réponse temporelle d'une batterie est aussi sensible aux capacités créées par la polarisation électrique et la diffusion des charges accumulées. Donc, ces effets capacitifs sont modélisés avec des réseaux ( $R_1C_1$  et  $R_2C_2$ ) et sont intégrés dans le modèle afin de prédire les variations transitoires. Le circuit de la Figure III-62 est implémenté sous Simulink en utilisant SimPowerSystems.

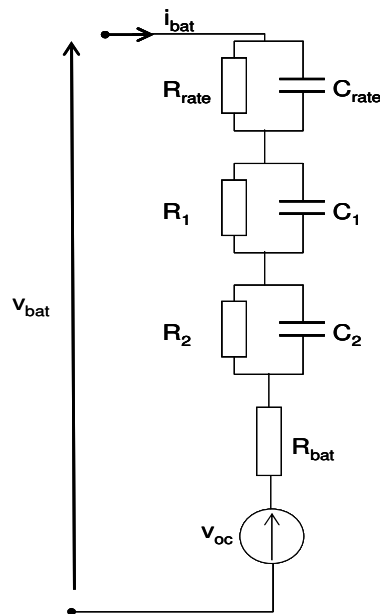


Figure III-61 : Circuit électrique équivalent d'une batterie

Pour vérifier le modèle au niveau comportemental ci-dessus, nous utilisons les paramètres donnés par le Tableau III-9.

Paramètres	$I_{bat}$ [A]	$V_{oc}$ [V]	$T$ [°C]	$C_{nominal}$ [Ah]	$R_{bat}$ [Ω]	$C_{rate}$ [Asec]	$R_{rate}$ [Ω]	$C_1$ [F]	$R_1$ [Ω]	$C_2$ [F]	$R_2$ [Ω]
Valeurs	1	2,5	23	1,5	0,175	0,2C	1,4	400	0,125	4	0,025

Tableau III-9 : Paramètres de la batterie

Cette configuration est celle de la batterie citée précédemment et dont les simulations sont réalisées pour  $C_{rate}=0,2 \times C_{nominal}$ . Nous reportons sur la Figure III-62 les tensions de charge et de décharge.

La courbe de charge est obtenue pour un courant constant  $i_{bat}=IA$ , et la tension initiale de  $C_{rate}$  est mise à zéro. Cette tension est équivalente à l'état de charge (SOC<sup>43</sup>) lorsque la batterie est déchargée (la tension de départ est donc  $V_{oc}$ ). Nous pouvons constater que l'évolution de la tension est cohérente en comparaison à la Figure III-59 (la tension à partir de laquelle la batterie se charge est 2,675V et la tension de charge est 4,14V obtenue pour 1,26h). **Ce modèle est donc satisfaisant pour modéliser le phénomène de charge.**

Nous avons vérifié ce modèle pour la décharge de la batterie pour la même configuration ci-dessus. La tension équivalente à l'état de décharge (SOD<sup>44</sup>) initial (batterie chargée) est 1,7V (4,2-2,5). L'allure de

<sup>43</sup> State Of Charge

<sup>44</sup> State Of Discharge



la tension de décharge pour deux valeurs négatives du courant : (a)  $i_{bat} = -1A$ , (b) un courant dont la forme est donnée par le Figure III-63.

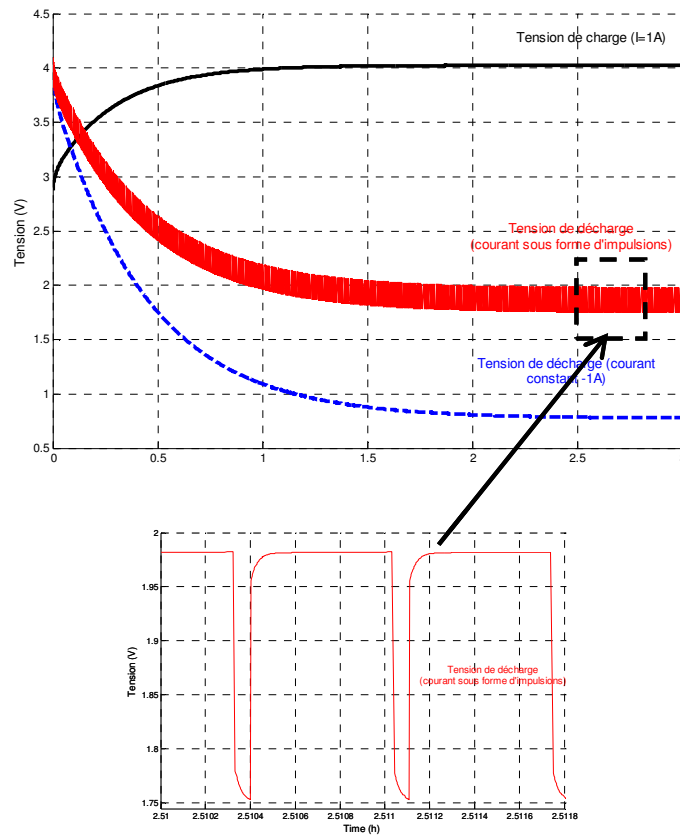


Figure III-62 : Caractéristiques de la batterie Sony 18650 (4,2 V; 1,5 Ah)

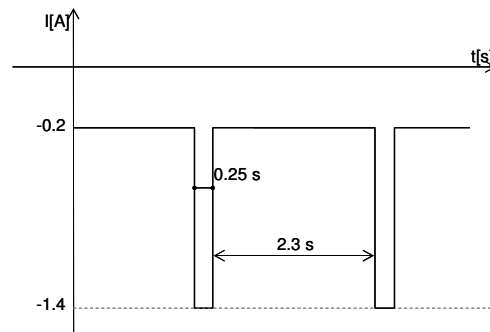


Figure III-63 : Train d'impulsions appliqué pour la décharge

Les résultats obtenus pour la tension reflètent le comportement de décharge d'une simple capacité pour les deux courants appliqués. En comparaison avec la courbe de la Figure III-59 pour différents courant de décharge, ce modèle équivalent électrique ne convient pas à la description du comportement de décharge et nécessite l'implication d'équations décrivant la non linéarité de la tension en fonction de l'état de décharge. Cette non linéarité de l'état de décharge (SOD) est en général déterminée selon des tables de

conversion expérimentales (LUT)<sup>45</sup> qui ne peuvent pas être facilement implémentées dans un circuit électrique.

#### 4.1.2 Modèle basé sur les LUT

Dans ce paragraphe, nous nous sommes intéressés à développer un modèle basé sur les tables de conversion fournies par les fabricants. Ainsi, nous avons implémenté les deux modèles de charge et de décharge sous MATLAB/Simulink. Le principe de cette méthode est de réaliser le modèle par la capture des caractéristiques réelles de la batterie afin de prédire son comportement sous différentes conditions. Ces caractéristiques sont extraites de courbes expérimentales et implémentées sous la forme de tables de conversion (LUT).

Nous avons développé les deux modèles pour décrire la charge et la décharge de la batterie. Chaque modèle se compose de deux parties. Une partie analytique pour calculer l'état de charge/décharge (SOC/SOD) en fonction du courant injecté dans la capacité nominale, le temps de fonctionnement, et l'état de charge initial (équation III-61). La deuxième partie concerne l'extraction de la valeur de la tension  $v_{bat}$  qui correspond à la valeur de l'état de charge  $SOC$  à l'instant  $t$  (équation III-62).

$$SOC = SOC(0) + \frac{I_{bat}}{C} t \quad III-61$$

$$v_{bat} = LUT(SOC_t) \quad III-62$$

Pour ce faire, nous utilisons le bloc LUT disponible dans la librairie de Simulink et nous introduisons les valeurs de l'état de charge ainsi que les valeurs de la tension. La table utilisée pour la charge contient dix points de SOC et donc chaque valeur est indexée par interpolation lors des simulations (la tension pour la batterie **Sony US18650** varie de 2,5V à 4,2V ce qui correspond à un état de charge de 0% à 100%).

La Figure III-64 est le modèle comportemental de la charge sous Simulink. La table de conversion est construite à partir de la courbe du fabricant [SON].

---

<sup>45</sup> Look Up Table

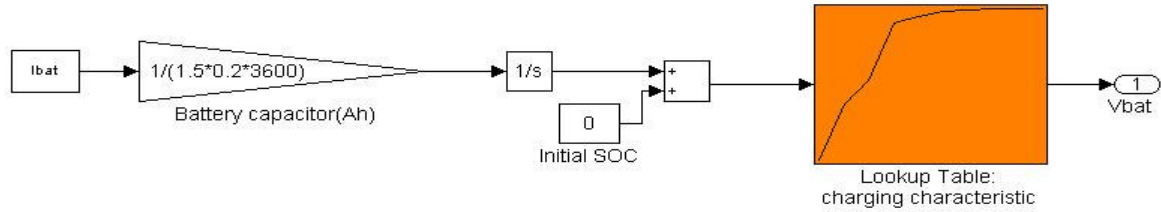


Figure III-64 : Implémentation du modèle de la batterie en charge avec un LUT

L'exécution du modèle pour différentes valeurs du courant (2,8A; 1A et 0,28A) illustre le comportement temporel de la batterie en charge sur la Figure III-65. Cependant, la précision du modèle dépend du nombre de valeurs dans la table de conversion.

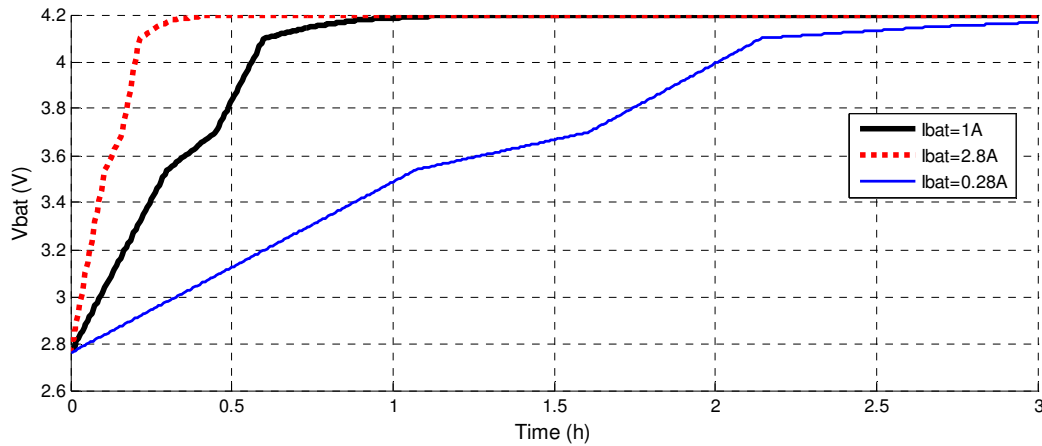


Figure III-65 : Tension de charge pour la batterie Sony 18650 (4,2 V; 1,5 Ah)

Dans le cas de la décharge, une deuxième table de conversion est utilisée pour construire le modèle de la batterie en décharge (voir la Figure III-66). La tension aux bornes de la batterie est indexée à partir des valeurs de l'état de décharge (SOD) (équation III-63). Nous posons l'état de décharge initial à zéro ce qui correspond à la première tension dans le LUT (tension maximale de la batterie est 4,2V).

$$V_{bat} = LUT(SOD)$$

III-63

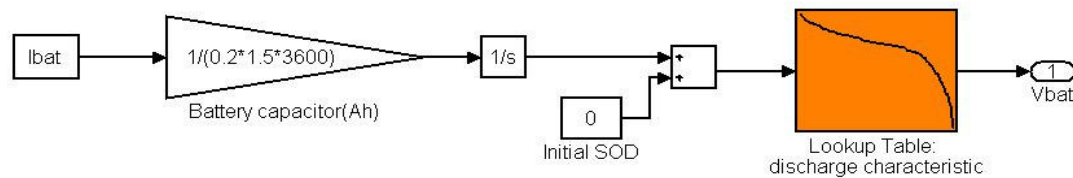


Figure III-66 : Implémentation du modèle de la batterie en décharge avec un LUT

La relation entre la tension aux bornes de la batterie en fonction de l'état de décharge pour  $C_{nominal}=1,5Ah$  et un courant de  $1A$  est représentée sur la Figure III-67. La courbe correspond aux résultats publiés dans la référence [Gao-02].

Les modèles basés sur les tables de conversion conviennent à l'étude du comportement de la batterie dans le domaine temporel. Sur la Figure III-67, la courbe de la tension est donnée en fonction de l'état de décharge.

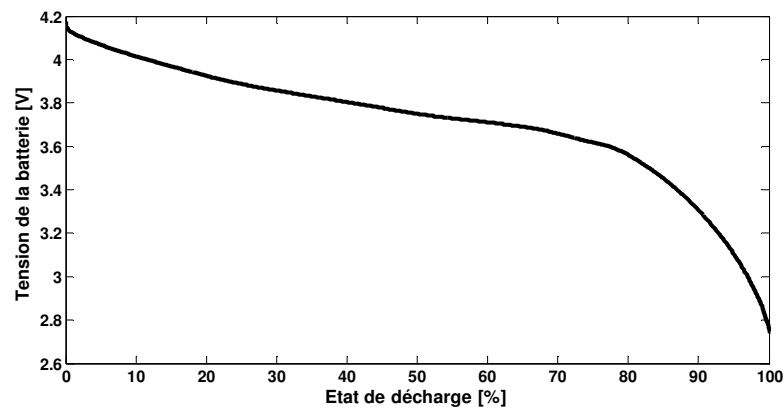


Figure III-67 : Caractéristique de  $v_{bat}$  en fonction de l'état de décharge

Sur la Figure III-68, nous donnons les courbes de décharge pour plusieurs valeurs de courant. Les courbes représentent la réponse du modèle en fonction du temps nécessaire pour décharger la batterie pour un courant donné.

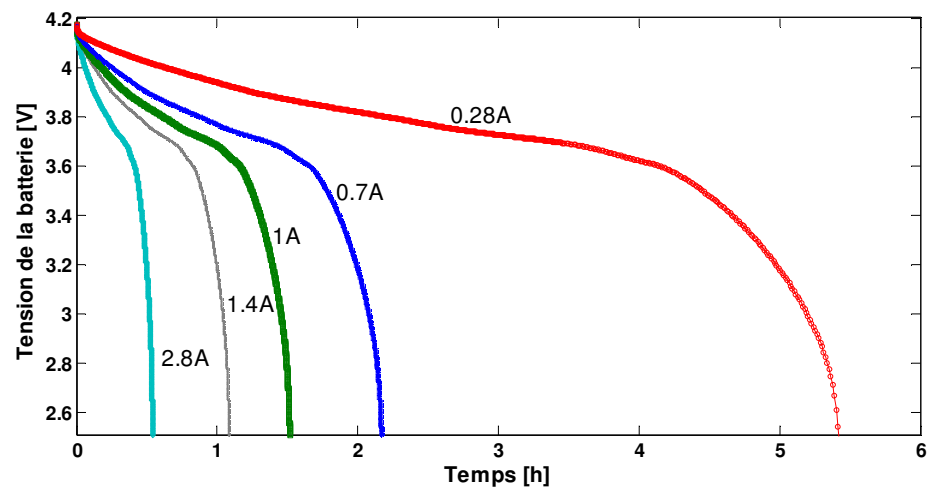


Figure III-68 : Caractéristiques de la batterie pour différents courants de décharge

Les modèles de charge et décharge basés sur les tables de conversion décrivent mieux le comportement de la batterie pour les deux modes de fonctionnement. Cependant les modèles peuvent être

plus précis en utilisant plusieurs caractéristiques d'état de charge et de décharge mesurées pour plusieurs capacités nominales. Ces caractéristiques peuvent être introduites dans un LUT à  $n$  dimensions ce qui permet d'interpoler efficacement les tensions en fonction des courants injectés.

### 4.1.3 Modèle de décharge par interpolation de courbes

Une autre méthode de modélisation du comportement de la batterie consiste à construire le modèle à partir d'un polynôme dont les paramètres sont extraits des caractéristiques expérimentales [Gao-02] [Amm-04]. Les équations décrivant la tension de la batterie en fonction de l'état de décharge, de la variation de la température, et des décalages du courant par rapport au courant nominal et à la température nominale sont implémentées sous Simulink sous forme de fonctions systèmes "*S-FUNCTION*" (Figure III-69). Ces S-function sont données en Annexe J.

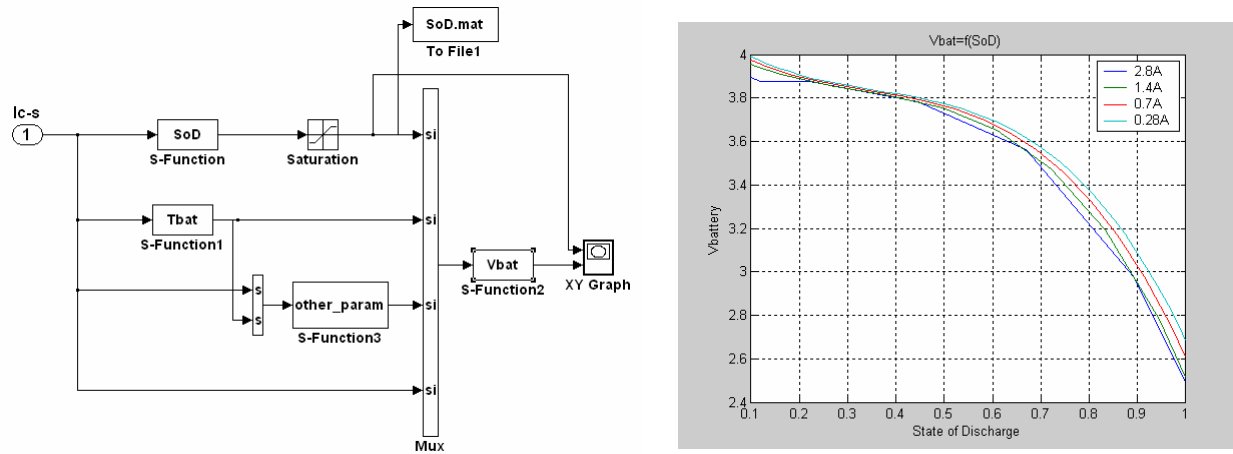


Figure III-69 : Modèle de la micro batterie en utilisant les S-FUNCTION

Ce modèle est vrai pour un seul de type de batterie. Il faut recalculer toutes les fonctions en utilisant les mêmes méthodes d'extraction pour d'autres ensembles de caractéristique.

### 4.1.4 Bilan

Les deux approches de modélisation au niveau comportemental permettent d'obtenir des modèles simples qui peuvent être utilisés lors de la concaténation de tous les sous-systèmes afin de valider le microsysteme. Le modèle avec les LUT peut être considéré fiable du fait que les résultats sont obtenus à partir des mesures expérimentales.

Ces modèles peuvent faire partie d'une bibliothèque de batteries et être utilisés selon le besoin et aussi les paramètres disponibles pour la modélisation. Cependant, d'autres modèles plus raffinés peuvent être réalisés pour représenter le comportement de ces dipôles pour tenir compte des phénomènes physiques. La section suivante traite de cet aspect.

## 4.2 Modèle raffiné de la batterie

Dans la littérature, plusieurs travaux ont exploré l'impact du courant appliqué, la température et l'usure d'une batterie. Pour ce faire, des modèles de charge et de décharge ont été développés. Ces modèles sont basés sur des circuits électriques équivalents [Gla-96] [Gol-97] [Ben-00]. Par définition, le processus de charge d'une batterie décrit les variations de la tension aux bornes d'une capacité effective. Alors que le processus de décharge représente la correction continue de la charge stockée dans la capacité et donc la valeur de la tension de sortie. En réalité la charge et la décharge d'une batterie sont des phénomènes non linéaires et dépendent du cycle de vie, de la résistance interne et des caractéristiques thermiques.

Dans la section 4.1, nous avons construit le modèle équivalent électrique permettant de déterminer l'état de charge en considérant la batterie comme un ensemble de circuits RC en série avec une source de tension. Nous avons vu que le modèle reflète le comportement de la charge mais présente des problèmes pour la décharge d'où la nécessité d'introduire d'autres phénomènes physiques. Ainsi pour que le modèle soit précis, il doit supporter la description des phénomènes complexes observés lors des cycles de charge et de décharge. L'introduction des détails de modélisation constitue un autre niveau d'abstraction. Pour cela, on définit le modèle au niveau circuit qui tient compte de l'état de charge, du débit, et de la fréquence de charge et de décharge de la batterie ainsi que l'effet de la température.

### 4.2.1 Implémentation du modèle

Le raffinement du modèle, nous conduit à modéliser d'autres phénomènes qui interagissent sur le comportement de la batterie. Par ailleurs, le modèle que nous étudions dans cette section sous MATLAB/Simulink/SimPowerSystems correspond aux travaux publiés dans [Gol-97].

- **Détermination du SOC**

L'état de charge SOC est déterminé par la quantité de charge maximale transférée à la capacité. Cette tension n'ayant pas de dimension est donnée par l'équation ci-dessous :

$$SOC = \frac{q}{C} = \frac{\int I_{bat} dt}{C} \quad III-64$$

L'état de charge ou de décharge dans le modèle est représenté sur la Figure III-70-b. Cet état dépend du débit du courant (courant de charge ou courant de décharge) dans la capacité et de tension  $V_{Lost}$ . Cette tension est en fonction de  $V_{rate}$  qui caractérise le changement du débit et déterminée par interpolation dans la table de conversion  $LUT(V_{rate}, V_{Lost})$ . L'état de charge ou de décharge est donc la différence de potentiel aux bornes de  $C$  à laquelle on ajoute  $V_{Lost}$ .

- **Sens du courant**

Le sens du courant de la batterie détermine son état de fonctionnement (charge ou décharge). D'un point de vue implémentation, le sens du courant est modélisé par une source de tension commandée par un courant et ce dernier est appliqué à la capacité avec une source de courant commandée par un courant.

- **Usure de la batterie**

L'usure de la batterie en fonction du temps est considérée comme la multiplication du facteur de correction  $[1-(4,4 \times 10^{-4} \times CYCLE)]$  par la capacité  $C_{bat}$ . La variable *CYCLE* représente le nombre de cycles de charge et de décharge.

- **Débit de la batterie**

Le rapport du courant à la capacité est la base du changement de l'état de charge. Ce rapport permet d'indexer une table qui contient le débit au niveau de la capacité. La constante de temps RC représente le délai nécessaire au changement de la tension dans la batterie (voir la Figure III-70-a).

- **Normalisation du SOC**

La tension de la batterie est obtenue à partir de la conversion de l'état de charge normalisé en utilisant une autre table. Par ailleurs, les outils de modélisation supportent les LUT avec des valeurs croissantes. La tension de la batterie est définie en tant qu'une fonction de perte de la capacité. Un circuit complémentaire est toutefois utilisé pour indexer correctement l'état de charge normalisé. Ce sous-système est représenté par la Figure III-70-d.

- **Impédance de la batterie**

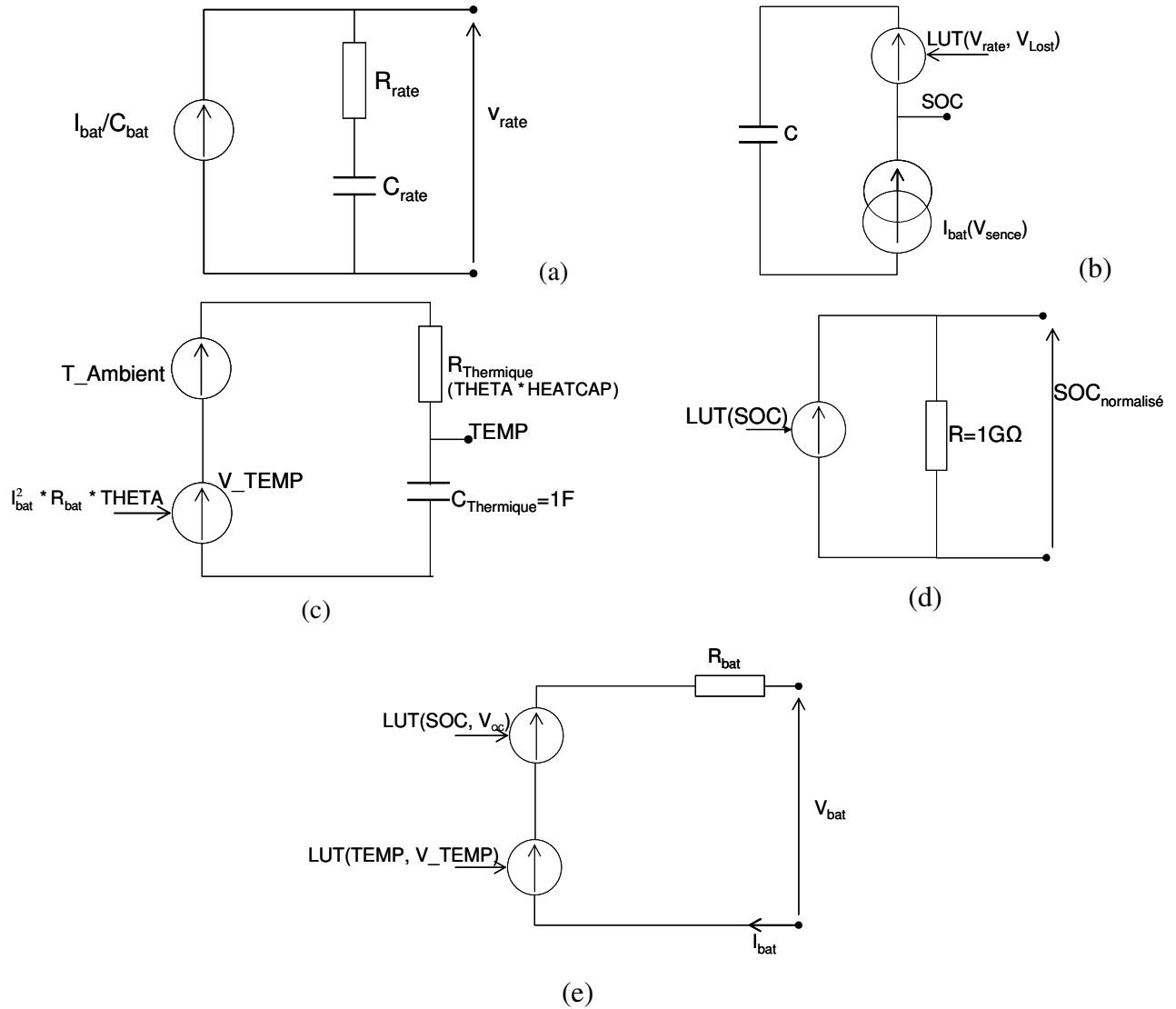
L'impédance d'une batterie est complexe et dépend des propriétés physico-chimiques. La modélisation de l'impédance est réalisée par la courbe de la tension en fonction de la capacité. Cependant, une résistance en série est nécessaire pour absorber les pics de courant et caractériser les dissipations.

- **Propriétés thermiques**

La dépendance de la batterie à la température est modélisée par une autre table de conversion. Le circuit électrique (Figure III-70-c). La tension calculée introduit une tension de seuil.

- **Tension aux bornes de la batterie**

La tension  $V_{bat}$  est modélisée par le circuit de la Figure III-70-e. la tension est la somme des potentiels aux bornes de la résistance, de la tension en circuit ouvert qui est extraite à partir d'une autre table en fonction de l'état de charge et du seuil introduit par la température.



**Figure III-70 : Sous-systèmes du modèle au niveau circuit de la batterie**

Pour implémenter ce modèle qui est basé sur des circuits équivalents électriques, sur des blocs Simulink et sur des tables de conversion, nous utilisons les techniques des sections précédentes (hiérarchisation et interfaçage entre les domaines). La Figure III-71 montre les différents sous blocs du modèle décrit ci-dessus implémentés sous Simulink en utilisant SimPowerSystems pour les parties électriques, les LUT et les éléments prédéfinis de Simulink.



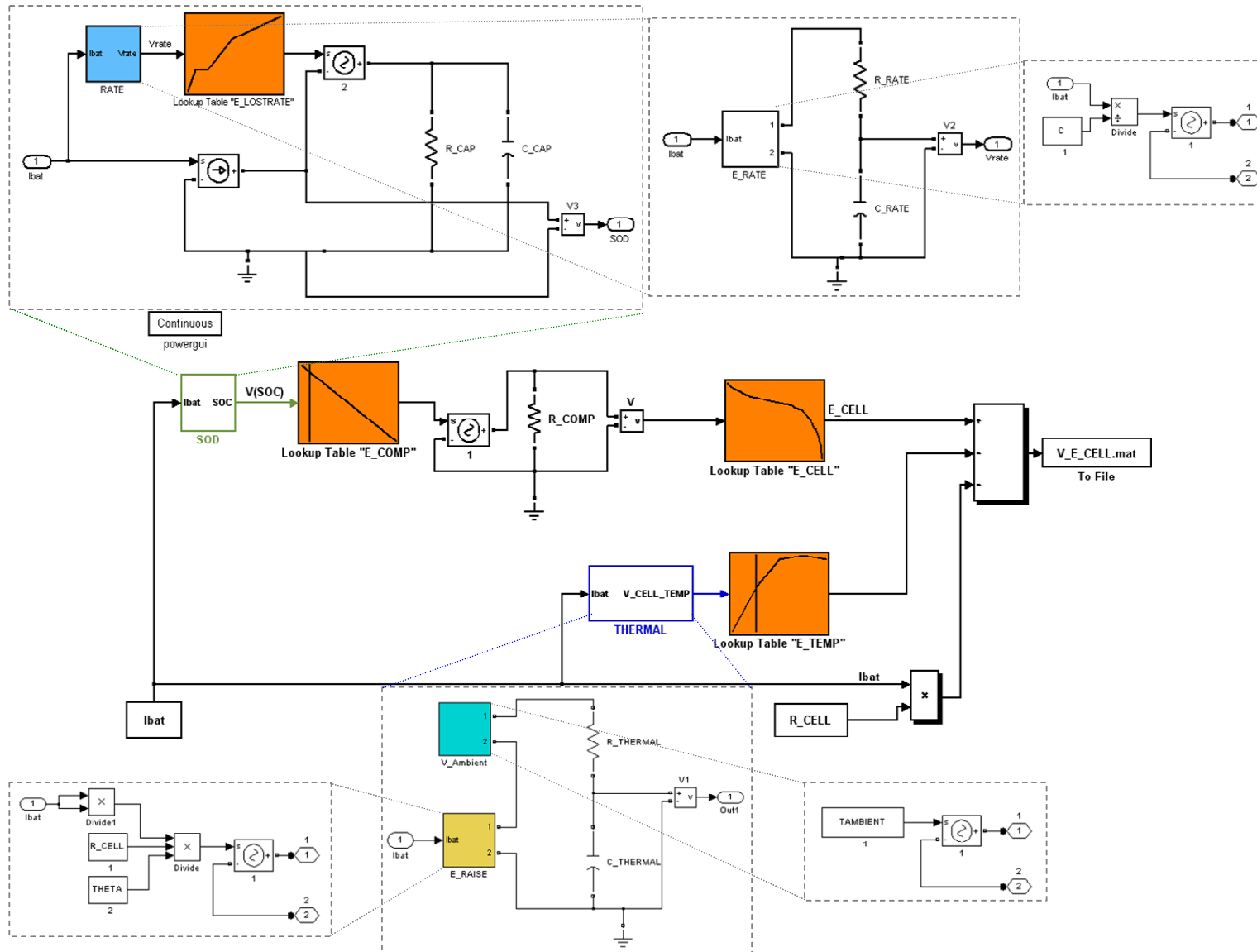


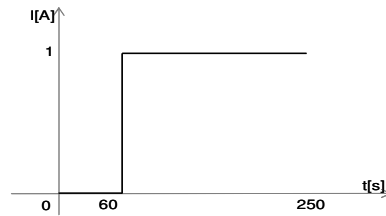
Figure III-71: Implémentation du modèle au niveau circuit de la batterie sous Simulink/SimPowerSystems

Pour vérifier ce modèle, nous utilisons d'une batterie Lithium-Ion **PolyStor ICR-18650**. Les tables de conversion utilisées sont pour  $0,2C$  de la capacité nominale qui est à  $C_{bat}=1,25Ah$ . Les différents paramètres du modèle sont donnés par le Tableau III-10.

$R_{bat}$ ( $\Omega$ )	$C_{bat}$ (Ah)	$R_{rate}$ ( $\Omega$ )	$C_{rate}$ (F)	$R_{cap}$ ( $\Omega$ )	$C_{cap}$ <sup>46</sup> (F)	$R_{thermique}$ <sup>47</sup> ( $\Omega$ )	$C_{thermique}$ (F)	T ( $^{\circ}C$ )
0.08	1,25	1	3	1G	$3600 \cdot C_{bat}$	Theta·HeatCap	1	20

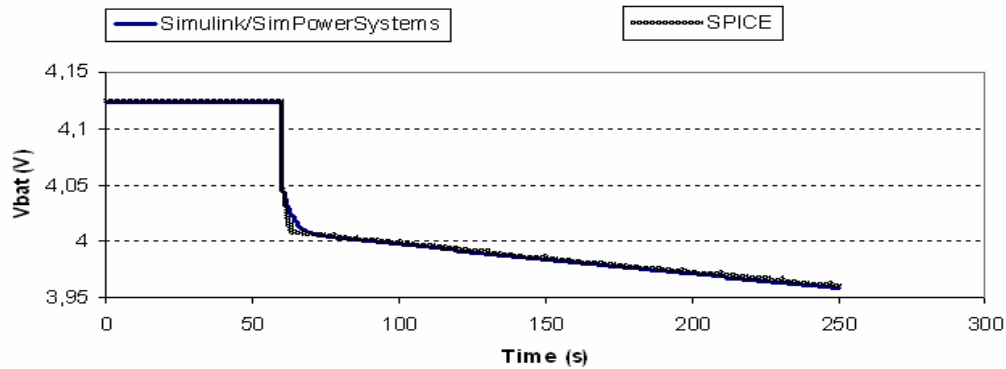
**Tableau III-10 : Paramètres du modèle au niveau circuit de la batterie [Gol-97]**

Le courant appliqué pour tester ce modèle est donné par la ci-dessous.



**Figure III-72 : Allure du courant appliqué pour la décharge**

Les résultats de la Figure III-73 montrent la concordance pour la décharge entre le modèle sous Simulink/SimPowerSystems que nous avons implémenté et les résultats obtenus sous SPICE par l'auteur [Gol-97]. L'erreur entre les deux courbes est de l'ordre de 0,2%. Cette erreur peut être attribuée aux méthodes d'interpolation des tables de conversion. Sous Simulink, plusieurs façons d'interpoler sont possibles et peuvent être choisies. A notre connaissance, sous SPICE l'interpolation est liée aux méthodes d'approximation internes et sont définies par les solveurs.



**Figure III-73 : Comparaison entre le modèle SPICE et Simulink/SimPowerSystems**

<sup>46</sup>  $C_{cap}$  est aussi en fonction du nombre de cycle de charge et de décharge CYCLE :  
 $C\_CAP=3600 \cdot C \cdot (1 \cdot CYCLE \cdot (4,4 \cdot 10^{-4}))$

<sup>47</sup> Theta=19,6 et HeatCap=31,5

Nous avons utilisé le modèle ci-dessus pour vérifier le comportement de charge de la batterie. Nous avons obtenu la courbe de la tension  $v_{bat}$  de la Figure III-74. L'état de charge initial dans ce cas, est égal à zéro. La tension de charge finale est  $4,09V$  obtenue au bout de  $1,25h$  de temps de charge.

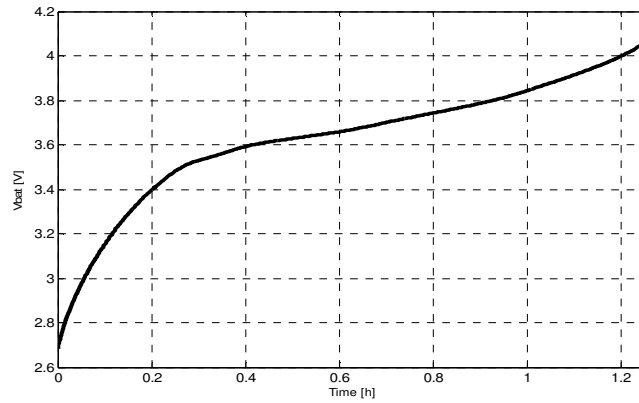


Figure III-74 : Tension de charge aux bornes de la batterie

### 4.3 Bilan

Nous reportons dans le tableau ci-dessous une synthèse des différents modèles développés pour la batterie. Les descriptions en circuit électrique équivalent et le modèle basé sur la table de conversion sont de haut niveau. Le deuxième modèle constitue une description plus précise du comportement de la batterie.

Le troisième modèle est plus précis mais l'implémentation d'autres types de batteries nous paraît plus difficile à réaliser. Cette difficulté est due à la disponibilité de tous les paramètres pour une batterie. Donc, d'autres modèles doivent être développés.

Type de Modèle de la batterie	Description	Degré d'abstraction	Facilité d'implémentation ou complexité	Précision
Modèle 1	Circuit électrique équivalent	++	+	+
Modèle 2	Analytique $V_{bat}=LUT(SOC)$ ou $V_{bat}=LUT(SOD)$	++	+	++
Modèle 3	Analytique+électrique $LUT(SOC+\text{débit}+\text{température}+\text{usure})$	+	++	+++

Tableau III-11 : Synthèse de la modélisation d'une batterie

## 5. Conclusion

Dans la première partie du chapitre III, après avoir défini et partitionné le microsysteme sans fil en sous-systèmes, nous avons appliqué les techniques présentées dans le Chapitre II-A pour modéliser les différents sous-systèmes. L'approche que nous proposons (cf. Chapitre II-A.7.3) suppose que chaque modèle des sous-systèmes peut être développé individuellement puis raffiné selon plusieurs niveaux de détails. Ainsi, nous avons proposé plusieurs descriptions pour le générateur piézoélectrique selon plusieurs niveaux d'abstraction. Nous avons élaboré différentes techniques pour modéliser ce type de dispositif en prenant en compte le problème du fort couplage avec la charge à ses bornes.

Nous avons aussi implémenté deux modèles pour le convertisseur AC/DC. Ce sous-système étant purement électrique, nous avons utilisé pour le premier modèle les composants prédéfinis de SimPowerSystems. Pour le deuxième modèle, nous nous sommes basés sur des travaux réalisés au sein de l'équipe pour développer le modèle équivalent SPICE de niveau 1 en utilisant les fonctions mathématiques des composants non linéaires.

Ensuite, nous avons étudié et développé les modèles du sous-système d'adaptation de puissance DC/DC. Ce sous-système se compose d'éléments électriques linéaires et non linéaires ainsi qu'une partie purement numérique. L'objectif de cette partie était de rappeler le fonctionnement de ces systèmes, de développer leurs modèles en utilisant les ressources de MATLAB/Simulink.

Finalement, nous avons proposé plusieurs méthodes de modélisation pour les batteries. Ces modèles permettent de représenter le comportement de la batterie en mode de charge ou de décharge. Le choix du modèle dépend de la précision des études à faire sur le microsysteme global et aussi des caractéristiques disponibles pour chaque batterie.

Tous les sous-systèmes développés dans cette partie du chapitre ont été vérifiés puis validés individuellement. Ainsi, dans la deuxième partie du chapitre III, nous allons intégrer les modèles des sous-systèmes dans un premier prototype virtuel complet. Ceci permet d'observer le comportement global du microsysteme et vérifier le fort couplage entre les sous-systèmes. Nous avons adopté deux stratégies que nous développons et dont nous discutons les résultats de simulation.

## **B. Stratégie de la simulation**

Dans cette partie du manuscrit, nous concaténons les modèles développés dans la première partie de ce chapitre afin de construire le prototype virtuel du microsysteme autonome. Dans la suite nous devons vérifier l'interaction entre les sous-systèmes afin de valider le prototype virtuel. Pour ce faire, nous définissons deux stratégies. La première est basée sur l'implémentation du prototype avec tous les sous-systèmes sous le même environnement. La seconde stratégie se base sur la cosimulation multi langages, où l'environnement principal de modélisation (MATLAB/Simulink) fait appel à un autre environnement supportant le langage de description matériel (VHDL).

### **1. Simulation globale**

Nous définissons la simulation du prototype du microsysteme composé de tous ses sous systèmes en utilisant les capacités offertes dans l'environnement MATLAB/Simulink comme une stratégie de simulation globale.

Le prototype virtuel du microsysteme autonome est construit par concaténation des modèles des sous-systèmes développés dans la section A. Il contient des modèles continus et la partie de contrôle numérique. Ce type de modélisation est supporté par Simulink et la résolution d'un tel système se fait par les solveurs disponibles dans cet outil. Le solveur doit déterminer le pas de simulation pour lequel les contraintes de précision sur les états continus et le temps d'échantillonnage pour les états discrets soient satisfaites. Dans ce cas, le temps d'échantillonnage constitue une contrainte supplémentaire pour le solveur continu. Le pas choisi par le solveur continu est toujours plus petit que le temps d'échantillonnage. Le choix du solveur et les paramètres de simulation n'ont pas d'influence sur la partie numérique du système.

Le Tableau III-12 illustre les différentes simulations globales multi niveaux effectuées pour le microsysteme autonome.

La première simulation du premier niveau est caractérisée par l'utilisation des modèles équivalents électriques ce qui permet la validation fonctionnelle du prototype. Après cette validation, nous pouvons utiliser les modèles raffinés du générateur piézoélectrique ou/et de la batterie avec les modèles de l'AC/DC et du DC/DC du premier niveau. Pour le troisième niveau, nous pouvons utiliser les modèles raffinés de tous les sous-systèmes.

Niveau de description	Modèles			
	Générateur	Batterie	AC/DC	DC/DC
Niveau 1	Circuit électrique équivalent	Circuit électrique équivalent	Circuit électrique	Circuit électrique
Niveau 2	Modèle analytique raffiné	Circuit électrique		
Niveau 3			Modèle analytique raffiné	Modèle analytique raffiné

Tableau III-12 : Différentes simulations multi niveaux

## 1.1 Premier prototype virtuel

Le prototype réalisé est basé sur les modèles développés avec les éléments prédéfinis de Simulink/SimPowerSystems comme le montre la Figure III-75. Ce premier prototype est construit en concaténant les modèles équivalents électriques des sous-systèmes en utilisant SimPowerSystems. Pour ce faire, les modèles utilisés sont :

- ⇒ Le circuit électrique équivalent simple du générateur (cf. section A.2.2.1)
- ⇒ Le redresseur AC/DC (cf. section A.3.1)
- ⇒ Le convertisseur DC/DC (cf. section A.3.2.3)
- ⇒ Le circuit électrique équivalent de la batterie (cf. section A.4.1.1)
- ⇒ L'algorithme de contrôle décrit en diagramme bloc (cf. section A.3.4)

L'avantage de ce modèle est que les sous-systèmes sont connectés directement comme dans un environnement dédié à la modélisation des circuits électriques. De plus, la boucle de contrôle numérique est insérée dans ce prototype. Le calcul du rapport cyclique est réalisé par la mesure en temps réel d'une tension  $v_{sc}$  aux bornes d'une petite résistance  $R_{sc}$ . Dans la réalité cette tension marque le sens du courant dans la batterie et constitue le signal d'entrée d'un convertisseur analogique-numérique.

Nous rappelons dans le tableau ci-dessous les caractéristiques de chaque sous-système pour lesquels les simulations sont exécutées.

Composant/paramètre	Valeur
$i_p$	$3,1 \times 10^{-3} \sin(2\pi \times 53.8t)$
$C_{\text{piezo}}$	$184 \mu\text{F}$
$C_{\text{rect}}$	$50 \mu\text{F}$
$L$	$22 \text{mH}$
$C$	$10 \text{mF}$
$R_{\text{sc}}$	$5 \Omega$
$V_{\text{bat}}$	$4,2 \text{V}$

Tableau III-13: Paramètres utilisés dans le premier prototype virtuel

Les simulations sont effectuées sur une station de travail Dell Inspiron 9400 à base de processeur CPU INTEL (R) Core (TM) T7200 cadencé à 2GHz et associé à une mémoire de 2.0 Go.

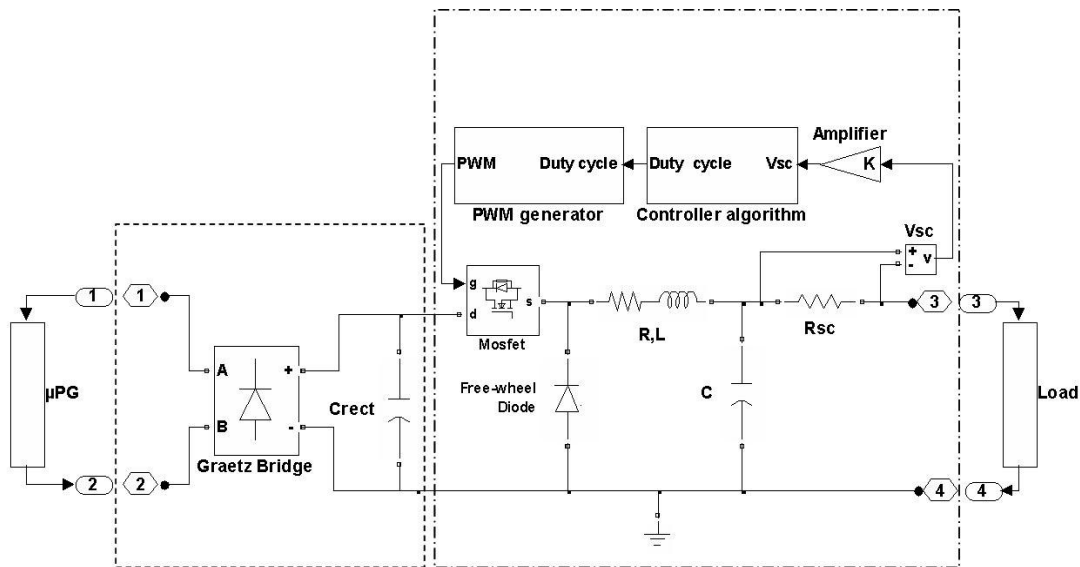


Figure III-75 : Premier prototype virtuel du microsystème

### 1.1.1 Simulation en mode continu

Les discontinuités correspondent généralement à un changement significatif dans le système dynamique. Dans nos modèles nous utilisons des blocs qui génèrent ces discontinuités (ex : sign, saturation, les LUT et les opérateurs logiques). Pour résoudre ces problèmes, les solveurs doivent détecter le passage par zéro avec précision à chaque pas de simulation.

Les simulations à pas fixe se produisent à des multiples du pas de simulation. Ceci ne garantit pas la détection précise des points de discontinuités. Cependant, la réduction du pas de simulation à pas fixe peut

rendre la simulation très lente. Nous choisissons les simulations à pas variables, car les solveurs ajustent le pas de simulation de façon dynamique. Les solveurs à pas variables procèdent en augmentant le pas lorsque les changements sont lents et en diminuant le pas lorsque les changements sont rapides.

La simulation du système ci-dessus, nous a amené à trouver un compromis entre la précision et la rapidité des simulations. Pour ce faire, nous avons utilisé le solveur à pas variable *ode15s*, avec un pas initial de  $5 \times 10^{-5}$ , et la valeur de la tolérance absolue à  $10^{-4}$ . Ces paramètres ont été choisis après plusieurs essais.

Nous avons exécuté une simulation en mode normal et une autre avec l'accélérateur de Simulink. Les résultats que nous obtenons sont superposés comme le montre la Figure III-76. en régime permanent, le rapport cyclique varie entre 16% et 20% et le courant est maximal pour la première valeur (à noter 16%). Ceci correspond à la valeur théorique calculée en mode continu pour la même configuration.

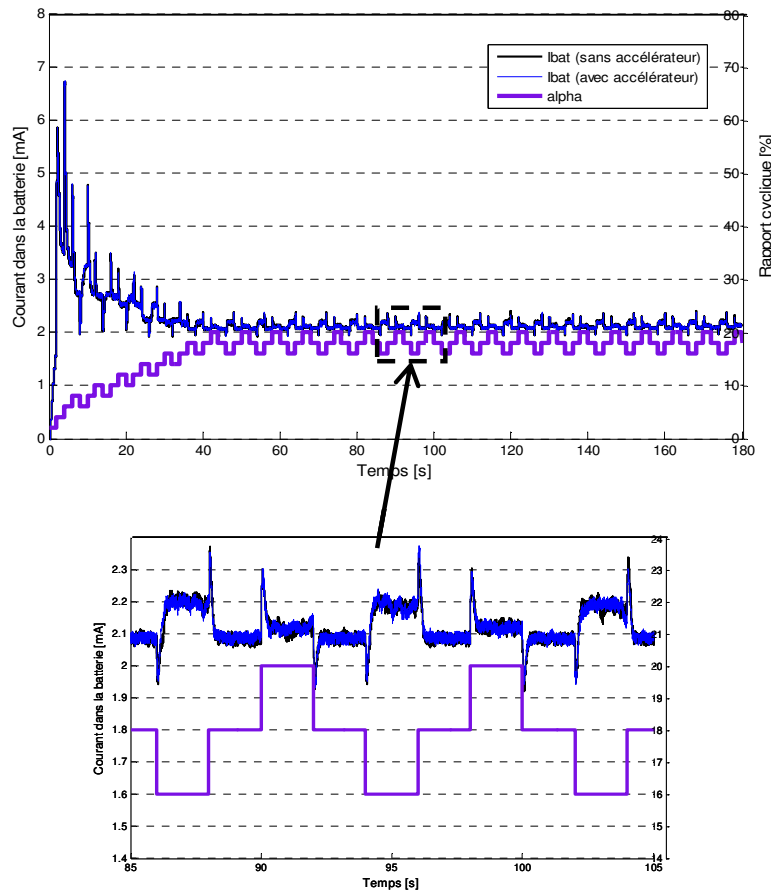


Figure III-76 : Résultats de simulation avec/sans l'accélérateur de Simulink

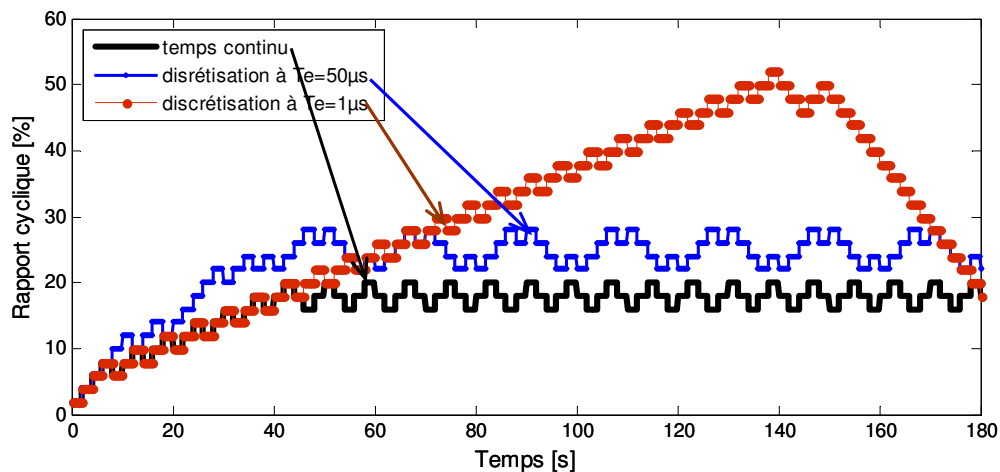
### 1.1.2 Discretisation du prototype

SimPowerSystems offre la possibilité de réaliser des simulations continues avec des algorithmes d'intégration à pas variable ou des solveurs discrets. Pour des petits systèmes, les algorithmes à pas

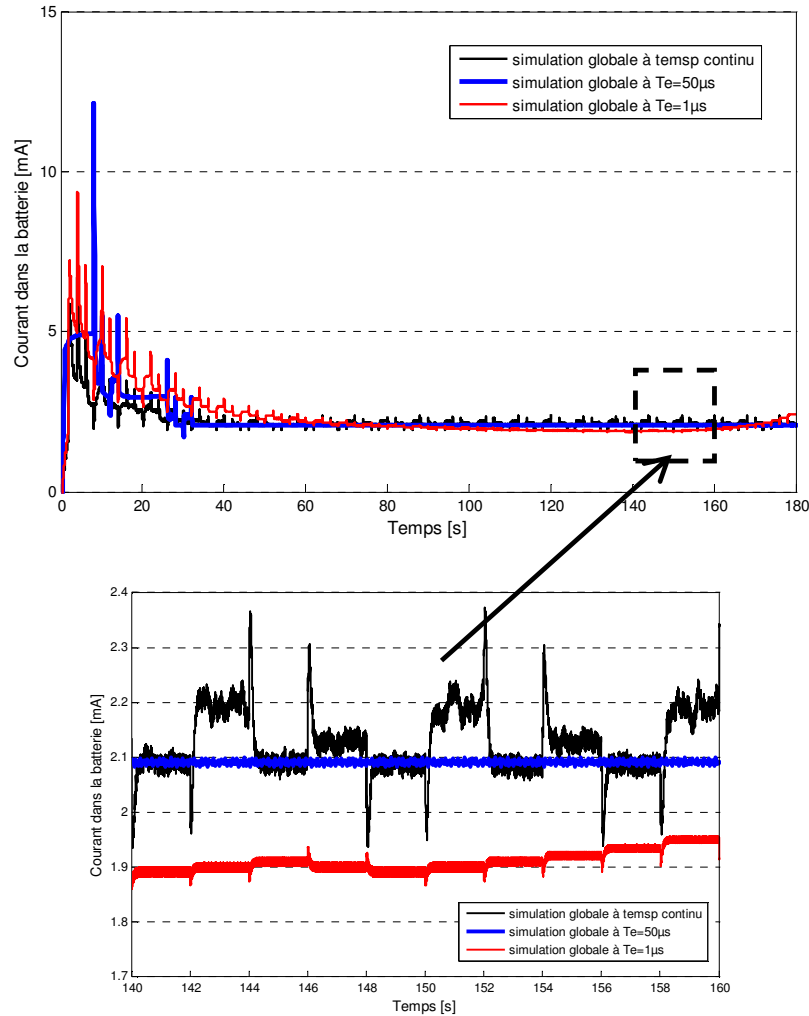


variable sont plus rapides que les méthodes à pas fixes. Ceci est dû au petit nombre de pas d'intégration. Pour les grands systèmes qui contiennent plusieurs états et des blocs non linéaires, il est plus avantageux de discrétiser le système électrique. La précision des simulations d'un système discrétisé est contrôlée par le pas de simulation. La seule manière d'avoir une bonne précision est de refaire des simulations avec différents pas et de trouver le bon compromis entre la précision et un temps de simulation acceptable. Le pas de simulation doit être réduit pour les systèmes comportant des dispositifs de commutations (transistor MOS, thyristors à gâchette et les commutateurs à hautes fréquences). La discrétisation du système est réalisée suivant la méthode d'approximation trapézoïdale *Tustin*. Cette méthode est équivalente à l'intégration trapézoïdale à pas fixe (*fixed-step*).

Pour discrétiser les modèles électriques dans le prototype, nous utilisons le bloc *Powergui*. Nous avons exécuté les simulations en mode continu et en mode discret pour obtenir les résultats de la Figure III-77 montrant l'évolution du rapport cyclique. La Figure III-78 représente les courbes du courant aux bornes de la batterie pour la simulation en mode continu et en mode discret. Dans ce cas nous avons choisi deux temps d'échantillonnage différents.



**Figure III-77 : Evolution du rapport cyclique pour différents modes de simulation globale**



**Figure III-78 : Comparaison des résultats de la simulation globale en temps continu et discret**

Il s'avère que le choix du temps de discrétisation est très important comme nous le remarquons sur la figure ci-dessus. En choisissant  $T_e=1\mu s$ , le calcul du rapport cyclique diverge de la valeur optimale, et le courant fourni à la batterie n'est plus maximal. Le résultat obtenu avec un temps d'échantillonnage de  $T_e=50\mu s$  est plus satisfaisant, d'autant plus que la durée de la simulation est très rapide comme nous le montrons dans le Tableau III-14 en comparaison avec d'autres modes de simulation. Nous avons obtenu ce résultat en **1mn22s en mode accélérateur avec discrétisation du prototype contre 31mn avec seulement l'accélérateur.**

## 1.2 Exemple du prototype virtuel avec plusieurs niveaux d'abstraction

Le prototype multi niveaux est représenté par le niveau2 dans le Tableau III-12 ce prototype est construit avec les modèles suivants:

- ⇒ Le modèle raffiné en diagramme blocs du générateur de puissance (cf. section de la partie A.2.2.2 de ce chapitre). Pour ce modèle nous utilisons les paramètres du Tableau III-1.
- ⇒ Le modèle du redresseur AC/DC au niveau comportemental (cf. section A.3.1).
- ⇒ Le modèle du convertisseur DC/DC au niveau comportemental (cf. section A.3.2.3).
- ⇒ Le modèle raffiné en circuit électrique équivalent de la batterie (cf. section A.4.2.1). Pour ce modèle nous utilisons les paramètres du Tableau III-10.
- ⇒ Le modèle de l'algorithme de contrôle décrit en diagramme bloc (cf. section A.3.4).

Nous avons utilisé le solveur à pas variable *ode15s*, avec un pas initial qu'on choisit après plusieurs essais  $5 \times 10^{-5}$ . Nous avons pris la valeur de tolérance absolue à  $10^{-4}$ .

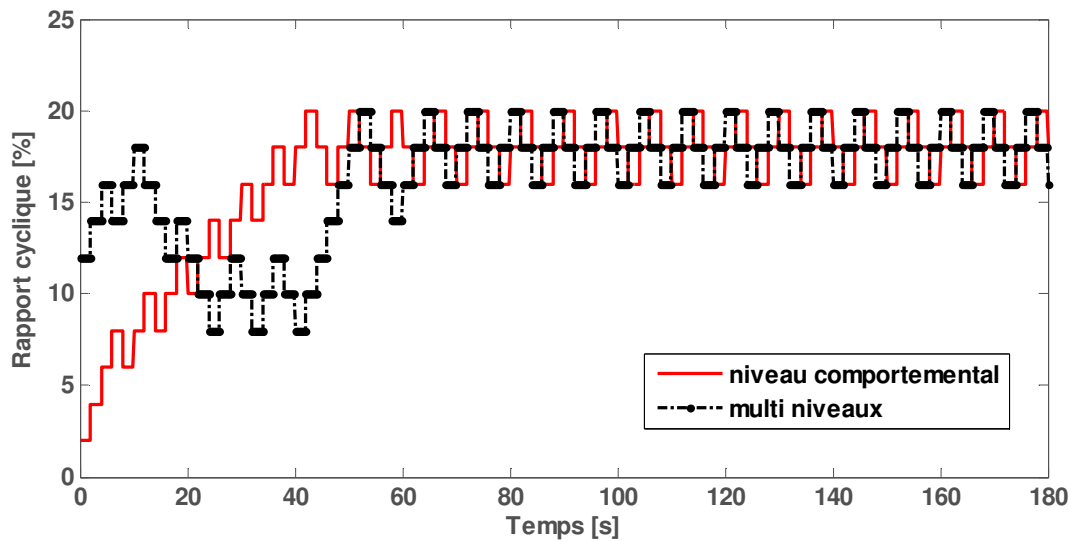


Figure III-79 : Evolution du rapport cyclique pour le prototype multi niveaux

La Figure III-79 représente une comparaison entre l'évolution du rapport cyclique du *niveau2* et du *niveau1* (en référence au Tableau III-12). Nous rappelons que le *niveau2* est construit avec le modèle raffiné du générateur piézoélectrique en diagramme blocs et le modèle raffiné de la batterie.

Nous avons initialisé le rapport cyclique pour le prototype multi niveaux (*niveau2*) à 12% et celui du prototype du *niveau1* à 2%. Nous pouvons observer que l'algorithme du *niveau2* s'adapte et converge vers la même valeur du *niveau1*.

Le courant mesuré aux bornes de la batterie dans ce cas est donné par la Figure III-80. Ce courant est très faible et ne permet pas de charger la batterie que nous utilisons dans notre application.

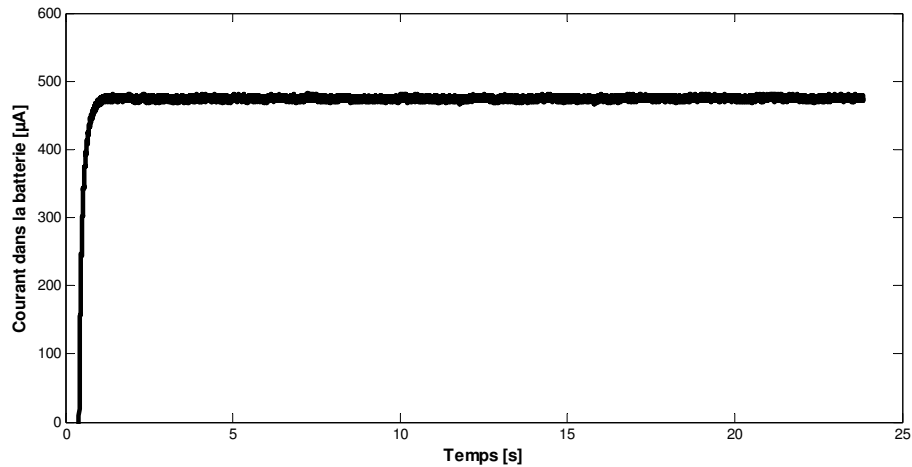


Figure III-80 : Courant dans la batterie pour le prototype multi niveaux d'abstraction

## 2. Cosimulation

Nous utilisons la cosimulation entre l'environnement Simulink/ModelSim en utilisant le lien direct appelé *Link for ModelSim*. Ce lien est un bloc qui permet de faire communiquer un composant en VHDL avec un modèle exécutable sous ModelSim<sup>48</sup>. Le bloc *HDL Cosimulation* est l'interface qui permet d'appliquer les signaux (vecteurs de test) de Simulink et de lire les résultats issus de ModelSim dans Simulink aussi. Ceci est réalisé par configuration des ports d'entrées/sorties, de la correspondance entre les échelles du temps entre Simulink et ModelSim. L'horloge du composant en VHDL peut être configurée dans le bloc *HDL Cosimulation* ou dans le code VHDL. En ce qui concerne la connexion entre les deux environnements, deux modes peuvent être utilisés soit par "*mémoire partagée*" ou par "*socket*".

Sur la Figure III-81 nous montrons le prototype virtuel du microsystème en utilisant la stratégie de cosimulation. L'introduction du bloc *HDL Cosimulation* pour remplacer l'algorithme de contrôle (de la section A.3.4) nécessite quelques configurations. Ainsi, nous avons utilisé la cosimulation multi langages par mémoire partagée du fait que nous travaillons en local. La correspondance pour *1ms* sous Simulink est de *1ms* sous ModelSim. Le temps d'échantillonnage est à *1ms*. Les deux ports d'entrée sont la nouvelle valeur correspondante au courant ( $v_{sens}$ ) et de l'ancienne valeur du rapport cyclique ( $\alpha_{in}$ ). Les signaux de sorties sont la nouvelle valeur du rapport cyclique ( $\alpha_{out}$ ) et le signal de remise à zéro (*rst*) du bloc accumulateur (running sum) servant à calculer la somme des échantillons du signal d'entrée qui est activé chaque 2000 échantillons (ce qui correspond à 2s). Nous introduisons des retards de *1ms* à travers le bloc "delay". Ce bloc a pour rôle l'échantillonnage et l'insertion d'un retard au niveau du signal d'entrée.

<sup>48</sup> La version utilisée : ModelSim SE Plus 5.7f de Mentor Graphics Corporation.

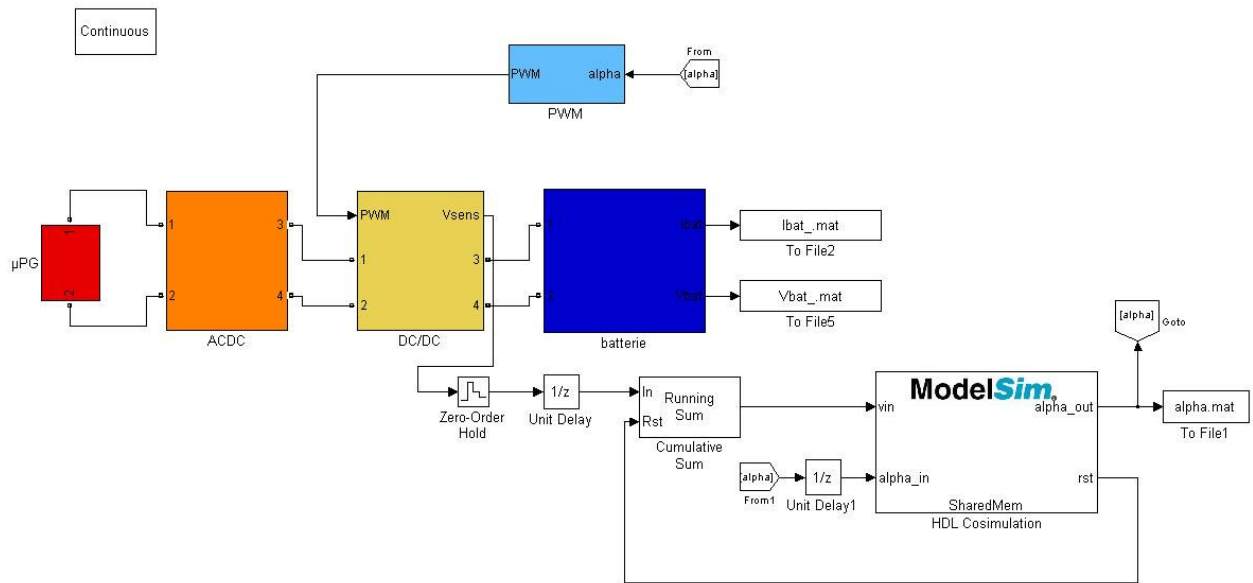


Figure III-81 : Prototype virtuel avec la cosimulation Simulink/ModelSim

Nous avons utilisé le solveur *ode15s*, avec un pas initial qu'on choisit à  $5 \times 10^{-7} s$ . Nous avons pris la valeur de tolérance absolue à  $10^{-4}$ . Au niveau de ModelSim, le fichier en code VHDL doit être compilé, et il faut configurer le même temps de simulation, pour que les résultats dans les deux environnements soient cohérents. L'horloge du sous-système est déclarée dans le code VHDL est la période est de *1ms*.

A ce stade, toutes les configurations sont réalisées et la simulation peut être lancée à partir de Simulink. Les résultats sous ModelSim sont obtenus à la fin de la simulation, alors que sous Simulink, ils peuvent être observés en temps réel. Cependant, pour accélérer la durée de la simulation, il est souhaitable d'utiliser les blocs de sortie "*to file*".

La Figure III-82 présente la fenêtre des signaux sous ModelSim. Nous pouvons constater que l'évolution du rapport cyclique (pwm sur la figure) chaque *2s* est de 2% (qui est le taux de changement du rapport cyclique *k*). Ce résultat correspond à celui observé sous Simulink à la sortie du bloc *HDL Cosimulation* comme le montre la Figure III-83 (pour la courbe en mode sans accélérateur). Le rapport cyclique varie entre 16% et 18%.

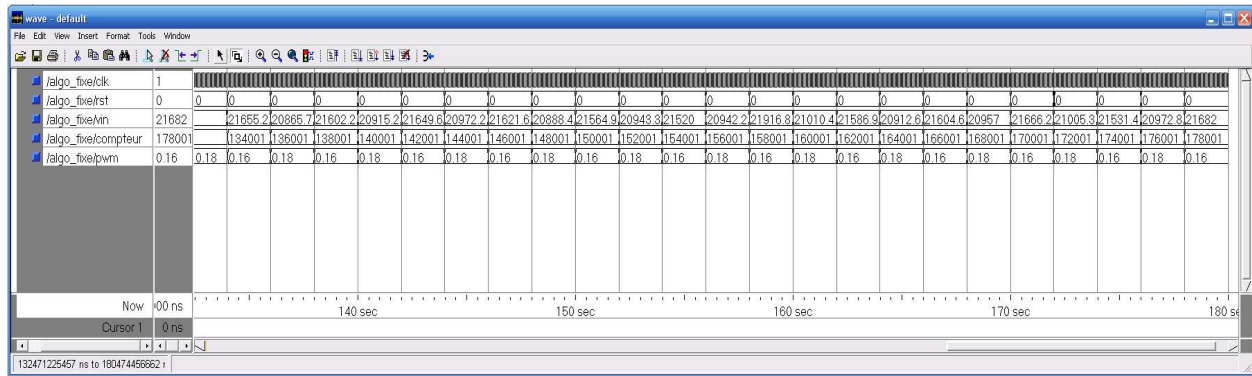


Figure III-82 : Fenêtre des signaux sous ModelSim

Nous avons comparé les deux modes avec et sans accélérateur et nous avons constaté une erreur de  $2k$  c'est-à-dire  $4\%$ . Et une erreur de  $\pm k$  ( $2\%$ ) en comparaison avec le résultat de la simulation globale. Cette erreur est due à la façon avec laquelle l'algorithme est implémenté. Nous rappelons que nous introduisons un retard au niveau du signal d'entrée du rapport cyclique ( $\alpha_{in}$ ) et que le signal dans le code VHDL est pris lors du coup d'horloge suivant. De cette manière, le signal de sortie du rapport cyclique ( $\alpha_{out}$ ) ne change pas de valeur pendant  $4s$  au lieu de  $2s$  pour le modèle de l'algorithme de contrôle sous Simulink.

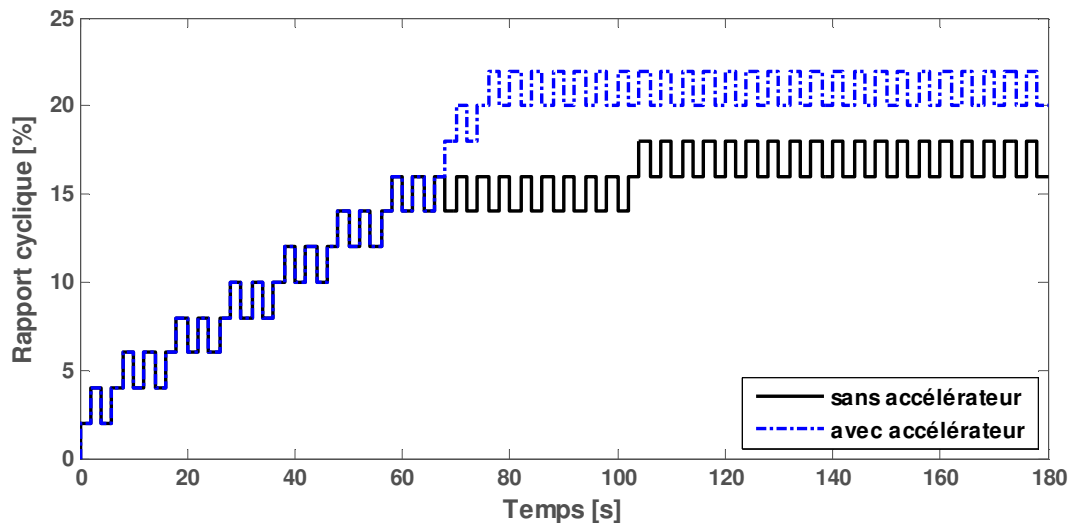
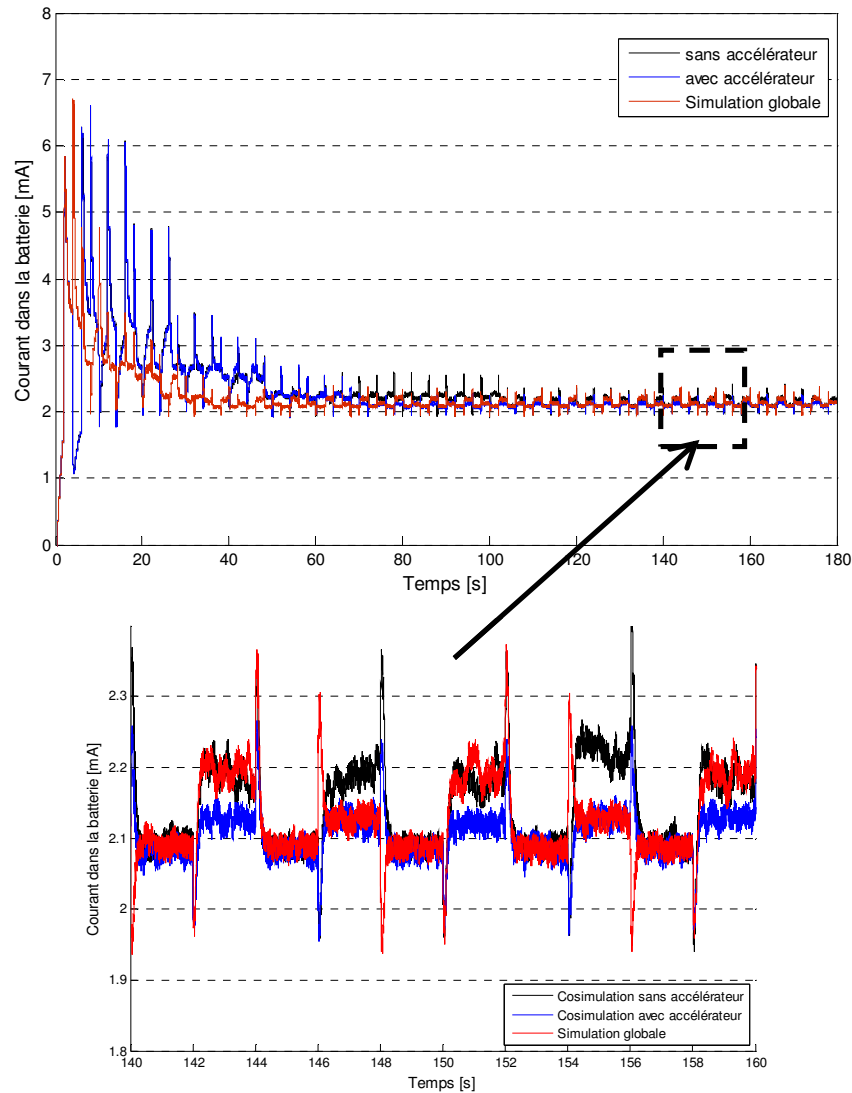


Figure III-83 : Evolution du rapport cyclique en cosimulation

Sur la Figure III-84 est montrée l'allure du courant dans la batterie pour la cosimulation Simulink/ModelSim pour les deux modes de simulation avec et sans accélérateur en comparaison avec le résultat de la simulation globale. Nous pouvons remarquer qu'il y a un léger décalage entre toutes les courbes en régime transitoire et qui se réduit de façon notable pour le régime permanent. Ce décalage peut être dû aux techniques de résolutions lors des simulations pour les modes sans et avec l'accélérateur de Simulink.



**Figure III-84 : Comparaison des résultats de cosimulation avec/sans l'accélérateur de Simulink et simulation globale**

Les résultats de la cosimulation Simulink/ModelSim sont encourageants. Nous pouvons espérer pouvoir intégrer un modèle en VHDL synthétisable dans le cadre de la modélisation collaborative.

### 3. Comparaison des temps de simulation

Dans le Tableau III-14 nous exposons des exemples de durées de simulations du prototype virtuel. Les simulations sont réalisées pour différents niveaux d'abstraction comme nous l'avons indiqué précédemment dans le Tableau III-12. Nous rappelons que trois niveaux de simulation ont été définis. Le niveau1 est exécuté pour tous les sous-systèmes qui sont modélisés sous la forme de circuits électriques équivalents en utilisant les éléments de SimPowerSystems (SPS sans le tableau ci-dessous) avec la partie

numérique en diagramme blocs ou en code VHDL. Le niveau2 est exécuté pour les modèles raffinés du générateur et de la batterie. Finalement, le niveau3 qui est réalisé pour les modèles analytiques raffinés des sous-systèmes.

Afin de réaliser la comparaison entre ces différents niveaux, les simulations sont exécutées pour un taux de changement de  $k=2\%$  chaque 2s pour un temps de 180s. Nous avons comparé les temps d'exécution entre différentes méthodes de simulation et nous avons obtenus les résultats ci-dessous.

Type/mode	Modèle	Temps	Durée de simulation
<b>Simulation globale</b>			
sans accélérateur	Prototype_niveau1 (SPS)	180	~1h15
accélérateur	Prototype_niveau1 (SPS)	180	~31mn
Virgule fixe	Prototype_niveau1 (SPS)	180	~30mn
accélérateur/discret	Prototype_niveau1 (SPS)	180	~ <b>1mn22s</b> avec $T_e=50\mu s$
accélérateur	Prototype_niveau2 Raffinement du $\mu PG \& Batterie$	180	~32mn
sans accélérateur	Prototype_niveau3 Raffinement des modèles	50	Plus de 7h
<b>Cosimulation Simulink/ModelSim</b>			
sans accélérateur	Link for Modelsim	180	~1h11mn30s
accélérateur	Link for Modelsim	180	~29mn

**Tableau III-14 : Comparaison des temps de simulation entre les différents modes de simulation**

Les résultats que nous avons montrés dans les paragraphes précédents démontrent que le prototype virtuel au niveau comportemental permet d'approximer le fonctionnement du microsystème. De plus, nous pouvons constater que les temps de simulation sont équivalents en mode normal pour les niveaux 1 et 2. En mode accélération, les simulations sont environ **2.5 fois** plus rapides que le mode normal. La discrétisation du prototype en utilisant le mode accélérateur, permet de rendre les simulations **75 fois** plus rapides.

Le niveau3 de simulation, le prototype contient des modèles raffinés (pour l'AD/DC et le DC/DC) dont l'implémentation a été réalisée avec des fonctions MATLAB ce qui a rendu les simulations très lentes.

Nous faisons la remarque qu'il est aussi possible d'utiliser la simulation à virgule fixe (*fixed point*) qui peut être sélectionnée dans l'onglet des outils. Nous avons constaté que les temps de simulation avec cette approche et en mode accélérateur sont équivalents dans notre cas.



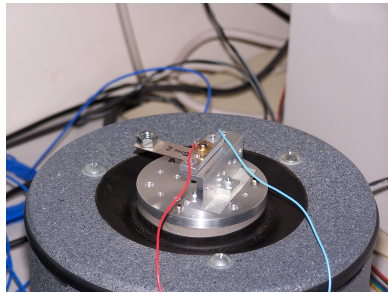
## 4. Validation expérimentale

La validation expérimentale du prototype virtuel du microsystème autonome est réalisée pour les paramètres donnés dans le Tableau III-15. Nous avons utilisé le système de l'image Figure III-85, nous avons utilisé une batterie NiCd du commerce. Cette batterie rechargeable dont la capacité nominale est  $940mAh$ , une tension de  $1,2V$  et un courant nominal de  $950mA$ . Sa résistance interne est de  $R_{bat}=0,012\Omega$ .

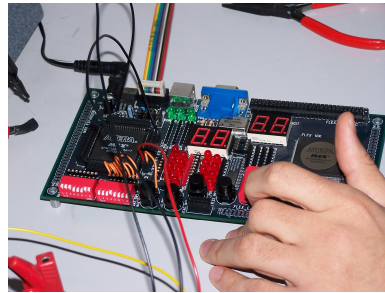
Nous donnons dans le tableau ci-dessous les paramètres physiques et géométriques du générateur piézoélectrique monté sur le pont vibrant de la Figure III-85-a. La Figure III-85-b et la Figure III-85-c représentent la carte d'un circuit FPGA de type ALTERA Flex 10k pour l'implémentation de l'algorithme de contrôle, et le banc du circuit de récolte d'énergie respectivement.

Désignation	Valeur	Description	Unité
$f_{piezo}$	87	fréquence de résonance	[Hz]
$C_{piezo}$	$35 \times 10^{-9}$	capacité du bimorphe	[F]
$t_c$	$179,5 \times 10^{-6}$	épaisseur des couches PZT	[m]
$t_{sh}$	$92 \times 10^{-6}$	épaisseur de la cale de laiton	[m]
$w_b$	$12,5 \times 10^{-3}$	largeur de la poutre	[m]
$l_b$	$28,14 \times 10^{-3}$	Longueur de la poutre	[m]
$l_e$	$28,14 \times 10^{-3}$	longueur de l'électrode	[m]
$l_m$	$9,86 \times 10^{-3}$	longueur de la masse	[m]
$w_m$	$11,3 \times 10^{-3}$	largeur de la masse	[m]
$m$	2,3	masse du bimorphe	[g]
$Y_c$	$80 \times 10^9$	module d'Young du PZT	[N/m <sup>2</sup> ]
$Y_{sh}$	$110 \times 10^9$	module d'Young de la cale	[N/m <sup>2</sup> ]
$D$	$120 \times 10^{-12}$	coefficient de couplage	[C/N]
$b_m$	7%	coefficient d'amortissement	
$\epsilon/\epsilon_0$	1326,2	constante diélectrique	
$A$	2	mode de fonctionnement 31	
$A_{in}$	6,9	accélération des vibrations	[m/s <sup>2</sup> ]

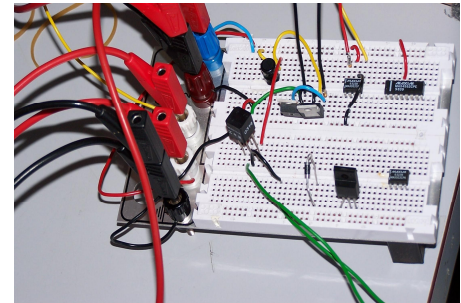
Tableau III-15 : Paramètres de la structure piézoélectrique du générateur



(a) Structure bimorphe sur un pont vibrant



(b) Carte FPGA de type ALTERA



(c) Circuit de récolte de l'énergie

Figure III-85 : Expérimentation d'un macro modèle à base d'une structure bimorphe

La Figure III-85-c représente les résultats obtenus pour le bimorphe ci-dessus. Le générateur est excité avec des vibrations continues d'une amplitude de  $6.9 \text{ m/s}^2$  à une fréquence de  $87\text{Hz}$ . Le MOSFET du convertisseur DC/DC fonctionne à  $1\text{kHz}$ . Ces résultats sont établis pour un rapport cyclique initial de  $1\%$ , et un taux de changement de  $0.2\%$  chaque  $2\text{s}$ .

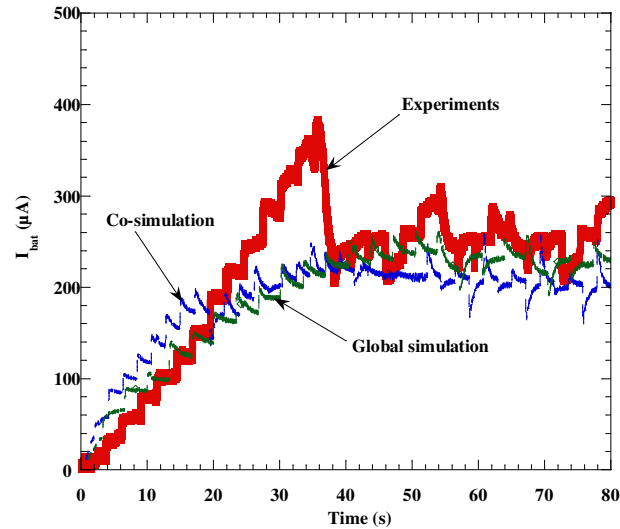


Figure III-86 : Comparaison entre le résultat expérimental, simulation globale et cosimulation

La Figure III-86 montre l'évolution du courant dans la batterie pour les deux stratégies de simulation présentées dans ce chapitre comparée au courant mesuré aux bornes de la batterie de l'expérience ci-dessus.

## 5. Conclusion

Dans cette partie, nous avons mis en œuvre le prototype virtuel du microsystème autonome à partir des modèles de ses sous-systèmes. Nous avons défini deux stratégies pour observer le comportement du microsystème. Ces stratégies peuvent être associées à la modélisation multi niveaux afin d'étudier l'influence du raffinement des modèles.

Lors de la simulation globale, nous avons utilisé plusieurs modes de simulation : avec et sans l'accélérateur, la discrétisation du prototype et la simulation en virgule fixe. Ces deux derniers modes sont équivalents. Le but de l'exécution des différentes simulations a été de vérifier la précision des résultats par comparaison des différents modes et de déterminer les temps de simulations. Nous avons pu vérifier que la discrétisation du prototype virtuel en mode accélérateur permet de réaliser des simulations **75 fois** plus rapide qu'en mode normal avec un solveur continu dans notre cas.

Nous avons pu vérifier aussi la cohérence entre les résultats expérimentaux et ceux de la simulation globale et de la cosimulation avec ModelSim de MentorGraphics en utilisant la passerelle LinkForModelSim sous MATLAB/Simulink.

# Chapitre IV- Conclusion et perspectives

---

L'objectif de ce travail a été de proposer des méthodes pour simuler le fonctionnement d'un microsystème multi domaines à signaux mixtes complet, alors que classiquement un tel système complexe est validé lors du prototypage physique. Le dispositif MSMD étudié fait partie d'une nouvelle génération de microsystèmes dédiée à des applications sans fils. Ces microsystèmes peuvent être complètement indépendants de l'environnement externe.

Nous avons ainsi présenté une méthode de modélisation et de simulation d'un prototype virtuel qui permet à différents collaborateurs dans différents domaines d'interagir avec le concepteur système afin de vérifier et ensuite valider le comportement du système complet. Cette approche permet de raffiner le prototype virtuel du microsystème autonome. Le raffinement s'est fait par des modèles à différents niveaux d'abstraction des sous-systèmes constituant le microsystème. Nous cherchons par cette approche à rendre l'abstraction des descriptions plus flexible avec des modèles plus précis et un prototype simulable. L'approche de modélisation collaborative a pour but de réduire les coûts de conception et les temps de mise sur le marché.

Afin de mettre en œuvre cette approche, nous avons opté pour l'environnement MATLAB/Simulink pour modéliser et simuler le microsystème. Ce choix a été motivé par l'évolution permanente de cet environnement, les possibilités de modélisation offertes et ses avantages en termes de diversité d'outils d'analyse et de convivialité graphique. Nous nous sommes basés sur la toolbox SimPowerSystems pour les modèles électriques, et les diagrammes blocs pour la description mathématique.

Nous avons ensuite présenté le prototype du microsystème autonome permettant l'analyse de la charge de la batterie à partir du générateur piézoélectrique. Dans une première étape, nous avons procédé par la décomposition du microsystème en sous-systèmes. Ces sous-systèmes sont: le générateur piézoélectrique, les circuits AC/DC et DC/DC, l'algorithme de contrôle numérique et l'élément de stockage. Les modèles de chacun de ces sous-systèmes ont été réalisés selon différents niveaux d'abstraction.

Nous avons exploré différentes descriptions pour le générateur piézoélectrique et nous avons retenu deux niveaux de modélisation afin d'observer l'impact du raffinement dans le prototypage virtuel. Les circuits analogiques ont été aussi validés selon deux niveaux d'abstraction. Nous avons aussi développé le modèle en diagramme bloc et en code VHDL de l'algorithme de contrôle qui est utilisé pour commander le commutateur du DC/DC pour un maximum de puissance dans la batterie. Le dernier sous-

système modélisé est la batterie. La description de ce type de composant a été réalisée selon deux niveaux de détails. Les modèles basés sur les données des fabricants sont les plus fiables. Il est à noter que tous les sous-systèmes, ont été validés individuellement à cette étape.

Dans une seconde étape, nous avons simulé le prototype virtuel en concaténant les modèles des sous-systèmes. La vérification fonctionnelle du prototype est réalisée en proposant deux stratégies de simulations. La première stratégie a consisté à implémenter le prototype dans l'environnement Simulink entièrement. Pour cette technique, nous pouvons réaliser des simulations avec des modèles décrits à différents niveaux d'abstraction. Nous pouvons ainsi vérifier en comparaison avec les modèles relativement simples le résultat du comportement du microsystème par simulation. La deuxième stratégie est basée sur la cosimulation multi langages entre Simulink et ModelSim. Nous avons pu vérifier que la cosimulation dans notre cas est équivalente à la simulation globale en terme de résultats, de complexité de mise en œuvre et en temps de simulation.

Finalement nous avons comparé nos résultats de simulation à un prototype physique macroscopique où nous avons pu vérifier la validité de notre approche de modélisation et de simulation MSMD.

L'approche du prototype virtuel proposée présente une solution pour la modélisation et la simulation dynamique du fonctionnement des systèmes MSMD.

### **Perspectives :**

Concernant la modélisation du microsystème autonome, il serait intéressant à court terme d'introduire des modèles raffinés au niveau microscopique. De façon complémentaire, un autre objectif serait de développer une bibliothèque de modèles en se basant sur les travaux de Y. Ammar [Amm-07] et de M. Marzencki [Mar-07] par exemple, mais aussi de d'autres projets qui traitent essentiellement du domaine de la génération et de la récupération d'énergie. Il est important de noter, l'efficacité d'une approche de modélisation automatique des composants d'un microsystème MSMD. Dans cette thèse, certains modèles peuvent être rendu génériques (tel que le générateur de puissance et les circuits analogiques et la partie numérique) en réalisant leurs fonctions paramétrables par la génération des masques.

De plus, au cours de cette thèse, nous avons pu voir l'intérêt de la modélisation fonctionnelle des systèmes MSMD et surtout l'intérêt de la modélisation sous Simulink. Toutefois, cet environnement présente des insuffisances de modélisation d'un système MSMD complet de la spécification à la réalisation finale (le flot de conception complet pour la réalisation physique n'existe pas). Ceci est dû à l'absence d'outils de modélisation et de conception capable de couvrir tous les aspects nécessaires à la conception d'un tel système. Ainsi, à long terme, nous pouvons envisager la définition d'un flot global de la spécification système à l'implémentation réelle d'un microsystème MSMD ayant la capacité de couvrir tous les niveaux d'abstraction et les environnements multi domaines.

# Références

## A

- [Amm-04] Y. Ammar, S. Basrour, "*Behavioural Modelling of Micro-Batteries for Self Powered Micro Systems*", DTIP of MEMS and MOEMS, Montreux, Switzerland, pp. 101-105, 12-14 May, 2004.
- [Amm-07] Y. Ammar, " *Conception de systèmes de gestion d'énergie pour microsystèmes autonomes*", thèse en micro et nano électronique, Université Joseph Fourier, Grenoble, 2007.
- [Ana-96] G. K. Ananthasuresh, R. K. Gupta, S. D. Senturia, "*An approach to macromodeling of MEMS for nonlinear dynamic simulation*", in Proc. Dynamics Systems and Controls, ASME Int. Mechanical Engineering Congr. Exposition, vol. 59, Atlanta, GA, pp. 401-407, November 1996.
- [ANS] <http://www.ansoft.com/products/em/simplorer/>
- [Asa-98] G. Asada, T. Dong, F. Lin, G. Pottie, W. Kaiser, H. Marcy, "*Wireless integrated network sensors: Low power systems on a chip*", European Solid State Circuits Conference, The Hague, Netherlands, October 1998.
- [Asc-98] Ascher, U. M. Petzold, L. R., "Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations", Siam, Philadelphia, Pennsylvania, 1998.
- [Asa-98] G. Asada, T. Dong, F. Lin, G. Pottie, W. Kaiser, H. Marcy, "*Wireless integrated network sensors : Low power systems on a chip*", European Solid State Circuits Conference, The Hague, Netherlands, October 1998.
- [Asc-98] Ascher, U. M., and Petzold, L. R., "Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations", Siam, Philadelphia, Pennsylvania, 1998.
- [Asc-98] Ascher, U. M., and Petzold, L. R., "Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations", Siam, Philadelphia, Pennsylvania, 1998.
- [Asc-97] U. Ascher, P. Lin, "*Sequential regularization methods for nonlinear higher-index DAEs*", SIAM, Journal of Science and Computation, vol. 18, n°. 1, pp. 160-181, Jan 1997.

## B

- [Bac-97] G. Baciù, H. K. Kesavan, "*From Particle-Mass to Multibody Systems : Graph-Theoretic Modeling*", IEEE Trans. Syst. Man Cybern. Vol. 27, pp. 244-250, 1997.
- [Bal-97] F. Balarin, M. Chiodo, P. Giusto, H. Hsieh, A. Jurecska, L. Lavagno, C. Passerone, A. Sangiovanni-Vincentelli, E. Sentovich, K. Suzuki, and B. Tabbara, "*Hardware-Software Co-Design of Embedded Systems : The Polis Approach*", Kluwer Academic Press, 1997.
- [Bee-06] S. P. Beeby, M.J. Tudor, R.N. Torah, E. Koukharenko, S. Roberts, T. O'Donnell, S. Roy, " *Macro and micro scale electromagnetic kinetic energy harvesting generators*", Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS DTIP'06, Stresa, Lago Maggiore, Italy, pp.286-291, 26-28 April 2006.
- [Ben-00] Luca Benini, Giuliano Castelli, Alberto Macii, Enrico Macii, Massimo Poncino, Riccardo Scarsi, "*A Discrete-Time Battery Model for High-Level Power Estimation*", DATE 2000.

- [Boe-88] Ba. W. Boehm, "A Spiral Model of Software Development and Enhancement", IEEE Computer, vol. 21, pp. 61-72, May 1988.
- [Bou-06] A. Bouchhima, "Modélisation du logiciel embarqué à différents niveaux d'abstraction en vue de la validation et la synthèse des systèmes monopuces", thèse en microélectronique, INP de Grenoble, 2006.
- [Bra-66] Branin, F. H., "*The Algebraic-Topological Basis for Network Analogies and the Vector Calculus*", Symposium on Generalized Networks, Brooklyn, New York, pp. 453–491, 1966.
- [Bre-97] Breunese, A. P. J., and Broenink, J. F., "*Modeling Mechatronic Systems Using the SIDOPS1 Language*", ICBGM '97, Phoenix, AZ, pp. 301–306, 1997.
- [Bri-04] M. Brissaud, "*Modelling of non-symmetric piezoelectric bimorphs*", Journal of Micromechanics and Microengineering, vol. 14, pp. 150-1518, 2004.
- [Buc-94] J. T. Buck, S. Ha, E. A. Lee, and D. G. Messerschmitt, "*Ptolemy : A framework for simulating and prototyping heterogeneous systems*," International Journal of Computer Simulation, special issue on Simulation Software Development, vol. 4, pp. 155–182, April 1994.
- [But-81] E. M. Buturla, P. E. Cottrell, B. M. Grossman, and K. A. Salsburg, "*Finite element analysis of semiconductor devices : The FIELDAY program*", IBM J. Res. Develop., vol. 25, pp. 218–231, 1981.

## C

- [Cel-91] Cellier, F. E., "*Continuous System Modeling*", Springer-Verlag, 1991.
- [Cel-96] Cellier, F. E. "*Object-Oriented Modeling : Means for Dealing with System Complexity*", 15th Benelux Meeting on Systems and Control, Mierlo, Netherlands, 1996.
- [Cha-79] P. C. H. Chan, C. T. Sah, "Exact equivalent circuit model for steadystate characterization of semiconductor devices with multiple energy level recombination centers", IEEE Trans. Electron Devices, vol. ED-26, pp. 924–936, 1979.
- [Che-06] M. Chen, G.A. Rincon-Mora, "*Accurate electrical battery model capable of predicting runtime and I-V performance*", IEEE TRANSACTIONS ON ENERGY CONVERSION, vol. 21, n°. 2, June 2006.
- [Chi-06] Y. Chiu, C.-T. Kuo, Y.-S. Chu, "*Design and fabrication of a micro electrostatic vibration-to-electricity energy converter*", In Proc. of DTIP 2006, pp. 298-303, Stresa, Italy, 26-28 April, 2006.
- [Chr-99] E. Christen, K. Bakalar, "*VHDL-AMS : A hardware description language for analog and mixed-signal applications*", IEEE Trans. Circuits Syst. II, vol. 46, pp. 1263–1272, Oct. 1999.
- [COV] <http://www.coventor.com/coventorware/>
- [CRO] Crossbow (2005), "*MICA2/DOT Professional Kit*", Crossbow Technology, Inc. (Online Catalog), available at <http://www.xbow.com/Products/productsdetails.aspx?sid=69>

## D

- [Dav-01] J. Davis, II, C. Hylands, B. Kienhuis, E. A. Lee, J. Liu, X. Liu, L. Muliadi, S. Neuendorffer, J. Tsay, B. Vogel, and Y. Xiong, "*Ptolemy II : Heterogeneous concurrent modeling and design in Java*", EECS, University of California, Berkeley, Technical Memorandum UCB/ERL M01/12, March 2001.

- [Des-05] G. Despesse, "*Etude des phénomènes physiques utilisables pour alimenter en énergie électrique des microsystèmes communicants*", thèse en micro et nano électronique, Laboratoire TIMA, Grenoble, 2005.
- [Dia-00] Diaz-Calderon, A., Paredis, C. J. J., and Khosla, P. K., 2000, "*Reconfigurable Models : A Modeling Paradigm to Support Simulation-Based Design*", Summer Computer Simulation Conference, Vancouver, Canada, 2000.

[Dynasim] <http://www.dynasim.se/>

### E

- [Eds-99] Edström, K., 1999, "*Simulation of Newton's Pendulum Using Switched Bond Graphs*", Western MultiConference, San Francisco, California, 1999.
- [ELDO] ELDO. Mentor Graphics Inc. : <http://www.mentor.com/ams/>
- [Elm-98] Elmqvist, H., Mattsson, S. E., and Otter, M., "*Modelica : The New Object-Oriented Modeling Language*", The 12th European Simulation Multiconference, Manchester, UK, 1998.
- [Ess-01] R. Esser and J. Janneck, "*Moses — A tool suite for visual modeling of discrete-event systems*", in Proceedings of Symposium on Visual/Multimedia Approaches to Programming and Software Engineering, Stresa, Italy, Sep 2001.
- [EET-05] EETimes EDA Survey – Chip Survey, 2005 : [http://i.cmpnet.com/eetimes/eda/edasurvey\\_chip.pdf](http://i.cmpnet.com/eetimes/eda/edasurvey_chip.pdf)
- [ELDO] ELDO. Mentor Graphics Inc., <http://www.mentor.com/ams/>

### F

- [Fer-94] J.-P. Ferrieux, F. Forest, "*Alimentations à découpage - Convertisseurs à résonance*", DUNOD, 2e édition revue et augmentée, 1994.
- [Fer-95] J. B. Ferris, J. L. Stein, "*Development of Proper Models of Hybrid Systems : A Bond Graph Formulation*," International Conference on Bond Graph Modeling and Simulation, Las Vegas, Nevada, 1995.
- [Fis-97] P. A. Fishwick, "*Integrating Continuous and Discrete Models with Object Oriented Physical Modeling*", Western Simulation Multi Conference, Phoenix, Arizona, 1997.

### G

- [Gao-02] L. Gao, S. Liu, R. A. Dougal, " *Dynamic Lithium-Ion Battery Model for System Simulation*", IEEE TRANSACTIONS ON COMPONENTS AND PACKAGING TECHNOLOGIES, vol. 25, n°. 3, pp. 495-505, September 2002.
- [Gla-96] M. Glass, "Battery electrochemical non-linear dynamic S P I C E model", in Energy Conversion Engineering Conference, pp. 292–297, 1996.
- [Gol-97] S. Gold, "A P S P I C E macromodel for lithium-ion batteries", in Proc. Annual Battery Conference on Applications and Advances, pp. 9–15, 1997.
- [Gri-04] C. Grimm, K. Einwich, A. Vachoux, "*Analog and Mixed-Signal System Design with SystemC*", FDL'04 Tutorial, Lille, September 2004.
- [Gly-01] P. Glynn-Jones, S. Booby, and N. White, "*Towards a piezoelectric vibration-powered microgenerator*," IEE Proceedings Sci. Meas. Technol. vol. 148, pp. 68 -72, 2001.
- [Gly-92] P. W. Glynn, "A GSMP Formalism for Discrete Event Systems, in Discrete Event Dynamic Systems : Analyzing Complexity and Performance in the Modern World", Y.-C. Ho, ed., IEEE Press, pp. 11–20, 1992.



- [Gly-04] P. Glynn-Jones et al., "An electromagnetic, vibration-powered generator for intelligent sensor systems", *Sensors and Actuators A*, vol. 110, pp. 344-349, 2004
- [Guo-05] Y. Guo, K. Kakimoto, H.Ohsato, "*(Na<sub>0.5</sub>K<sub>0.5</sub>) NbO<sub>3</sub>-LiTaO<sub>3</sub> lead-free piezoelectric ceramics*", *Materials Letters*, vol. 59, pp. 241-244, 2005

## H

- [Hil-02] J. Hill, D. Culler, "*Mica : a wireless platform for deeply embedded networks*", *IEEE Micro*, n° 22, pp. 12-24, November-December 2002.
- [Her-02] Y. Hervé, "*VHDL-AMS : Applications et enjeux industriels*". Dunod-Université - collection : Sciences-sup - préface d'Alain Vachoux. ISBN : 2-10-005888-6 - mars 2002.
- [Hag-93] S. Hageman, "*Simple P S P I C E models let you simulate common battery types*", *Electronic Design News*, vol. 38, pp. 117-132, October 1993.

## J

- [Jac-99] G. Jacquemod, K. Vuorinen, F. Gaffiot, A. Spisser, C. Seassal, J.-L. Leclercq, P. Rojo-Romero, and P. Viktorovitch, "*MOEMS modelling for opto-electro-mechanical co-simulation*", *J. Modeling Simulation Microsyst.*, vol. 1, n° 1, pp. 39-48, 1999.
- [Jun-05] H. Al-Junaid, T. Kazmierski, "*Analogue and mixed-signal extension to SystemC*", *Institution of Electrical Engineers proceedings : Circuits, devices and systems*, vol. 152, n°6, pp. 682-690, 2005.
- [Jeo-05] Y. B. Jeon, R. Sood, J.-h. Jeong, S.-G. Kim, "*MEMS power generator with transverse mode thin film PZT*", *Sensors and Actuators*, pp.16-22, 2005.

## K

- [Kah-00] J. M. Kahn, R. H. Katz, K. S. J. Pister, "*Emerging Challenges : Mobile Networking for Smart Dust*", *Journal of Communications and Networks*, vol. 02, n° 3, pp. 188-196, September 2000.
- [Kar-75] D. L. Karnoop and R. C. Rosenberg, "*System Dynamics : A Unified Approach*", New York : Wiley, 1975.
- [Koe-67] H. E. Koenig, Y. Tokad, H. K. Kesavan, H. G. Hedges, "*Analysis of Discrete Physical Systems*", McGraw-Hill, New York, 1967.
- [Kun-00] K. Kundert, "*A Formal Top-Down Design Process for Mixed-Signal Circuits*", *Advances in Analog Circuit Design*, April 2000.
- [Kym-98] J. Kymissis, C. Kendall, J. Paradiso, N. Gershenfeld, "*Parasitic power harvesting in shoes*", *Proceedings of the Second IEEE International Conference on Wearable Computing ISWC*, Pittsburgh, PA, USA, 1998.

## L

- [Lan-02] R. Lanz, P. Carazzetti, P. Muralt, "*Surface Micromachined BAW Resonators Based on AlN*", *In Proc. Of 2002 IEEE Ultrasonics Symposium*, pp. 981-983, 2002.
- [Lee-03] J.M.H. Lee et al., Development of an AA Size Energy Transducer with Micro Resonators, *In Proc. of IEEE International Symposium on Circuits and Systems*, pp. , Bangkok, Thailand, May 25-28, 2003

## M

- [Mar-00] J. A. Martinez, "*Piecewise linear simulation of optoelectronic devices with application to MEMS*", M.S. thesis, Dept. Elect. Eng., Univ. Pittsburgh, Pittsburgh, PA, 2000.

- [Mar-07] M. MARZENCKI, "*Conception de microgénérateurs intégrés pour systèmes sur puce autonomes*", thèse en micro et nano électronique, Université Joseph Fourier, Grenoble, 2007.
- [Men-01] S. Meninger, J.O. Mur-Miranda, R. Amirtharajah, A.P. Chandrakasan, "*Vibration to electric energy conversion*", IEEE Transact. VLSI Syst. vol. 9, pp. 64–76, 2001.
- [Mor-97] J. Morikuni and S. Kang, "*Computer-Aided Design of Optoelectronic Integrated Circuits and Systems*", Englewood Cliffs, NJ : Prentice-Hall, chapt. 6, 1997.
- [Modelica] Modelica. <http://www.modelica.org/>
- [Muk-00] T. Mukherjee, "*CAD for MEMS design*", Proc. Design, Test, Integration, and Packaging of MEMS/MOEMS, pp. 3-14, Paris, 2000.
- [Mor-99] J. Morikuni, P. V. Mena, A. V. Harton, and K. W. Wyatt, "*The mixed technology modeling and simulation of optoelectronic microsystems*", Journal of Modeling and Simulation of Microsystems, vol. 1, n° 1, pp. 9-18, 1999.

## O

- [O'Do-06] O'Donnell T., Saha C. Beeby S.-P. and Tudor, M.-J., "*Scaling effects for electromagnetic vibrational power generators*", Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS - DTIP'06, Stresa, Lago Maggiore, Italy, 26-28 April 2006.
- [Ott-02] G. K. Ottman, H. F. Hofmann, A. C. Bhatt, G. A. Lesieutre, "*Adaptive piezoelectric energy harvesting circuit for wireless remote power supply*", IEEE Transactions on power electronics, vol. 17, n° 5, pp. 669-676, September 2002.
- [Oud-04] J. Oudinot, S. Guessab, "*Can MATLAB/SIMULINK Co-Exist with VHDL-AMS?*" EuroDesignCon' 04. Munich, Allemagne. 11-14 October 2004.
- [Ott-99] M. Otter, H. Elmqvist, S. Mattsson, "*Hybrid Modeling in Modelica based on the Synchronous Data Flow Principle*", CACSD'99, Hawaii, USA. August 22-26 1999.

## P

- [Par-05] J. A. Paradiso, T. Starner, "*Energy Scavenging for Mobile and Wireless Electronics*", IEEE Pervasive Computing, vol. 04, n° 1, pp. 18-27, Jan-Mar. 2005.
- [Pay-61] Paynter, H. M., "*Analysis and Design of Engineering Systems*", MIT Press, Cambridge, MA, 1961.
- [Pea-05] F. Peano, T. Tambosso, "*Design and Optimization of a MEMS Electret-Based Capacitive Energy Scavenger*", Journal of Microelectromechanical Systems, vol. 3, pp. 429-435, June 2005.
- [Pêc-05] F. Pêcheux, Ch. Lallement, and A. Vachoux, "*VHDL-AMS and Verilog-AMS as Alternative Hardware Description Languages for Efficient Modeling of Multidiscipline Systems*", IEEE Transactions on computer-aided design of integrated circuits and systems, vol. 24, n° 2, pp. 204-225, February 2005.
- [Pin-84] M. R. Pinto, C. S. Rafferty, and R. W. Dutton, "*PISCES II—Poisson and continuity equation solver*", Stanford Univ., Stanford, CA, Stanford Electronics Lab. Tech. Rep., September 1984.
- [Pou-04] G. Poulain, "*Contribution au développement d'un générateur piézoélectrique pour applications nomades*", Thèse présentée dans l'université paris XI, 2004.

## R

- [Rab-02] J. Rabaey , J. Ammer , T. Karalar , S. Li , B. Otis , M. Sheets, T. Tuan "*Picoradios for wireless sensor networks : the next challenge in ultra-low-power design*", Proc. Int. Conf. on Solid-State Circuits, San Francisco, CA, February 2002.
- [Ros-83] Rosenberg, R. C., and Karnopp, D. C., "*Introduction to Physical System Dynamics*", McGraw-Hill, New York, 1983.
- [Raj-01] S. Rajarishi, C. J. J. Paredis, L. Vei-Chung, K. K. Pradeep, "*Modeling and simulation methods for design of engineering systems*", Journal of Computing and Information Science in Engineering, vol. 1 , n° 1, pp. 84-91, March 2001, ISSN:1530-9827.
- [Rom-98] B. Romanowicz, "Methodology for the Modeling and Simulation of Microsystems", Norwell, MA : Kluwer, pp. 24-39, 1998.
- [Rou-03] S. Roundy , P. K. Wright, J. M. Rabaey, "*A study of low level vibrations as a power source for wireless sensors nodes*", Commun. Computer, pp. 1131-1144, 2003.
- [Rou-04] S. Roundy, P. K. Wright, J. M. Rabaey, "*Energy Scavenging for Wireless Sensor Networks*", Norwell, Massachusetts: Kluwer-Academic Publishers, 2004.
- 
- S
- [SAB] Saber. Synopsys Inc.:  
<http://www.synopsys.com/products/mixedsignal/saber/saber.html>
- [Sch-00] P. Schneider, E. Huck, S. Reitz, S. Parodat, A. Schneider, P. Schwarz, "*A modular approach for simulation-based optimization of MEMS*", Proc. ISMA2000, pp. 71-82, Singapore, December 2000.
- [Schw-00] P. Schwarz, "*Physically oriented modeling of heterogeneous systems*", Mathematics and Computers in Simulation 53, pp. 333-344, 2000
- [Schw-01] P. Schwarz, P. Schneider, "*Model Library and Tool Support for MEMS Simulation*", Conference on MICROELECTRONIC AND MEMS TECHNOLOGY, SPIE Proceedings Series, vol. 4407, Edinburgh, Scotland. 30.5.-1.6, 2001.
- [Sel-94] B. Selic, G. Gullekson, and P. Ward, *Real-Time Object-Oriented Modeling*. New York, NY : John Wiley & Sons, 1994.
- [Sen-98] S. D. Senturia, "*CAD Challenges for Microsensors, Microactuators, and Microsystems*", Proc. IEEE, vol. 86, n° 8, pp. 1611-1626, 1998.
- [Ser-06] C. Serre et al., "*Vibrational Energy Scavenging With Si Technology electromagnetic inertial microgenerators*", In Proc. Of... , pp. 292-297, Stresa, Italie, 26-28 April, 2006
- [Ses-61] S. Seshu, and M. B. Reed, "*Linear Graphs and Electrical Networks*", Addison-Wesley, Reading, Massachusetts, 1961.
- [She-97] C. Shearwood, R.B. Yates, "*Development of an electromagnetic microgenerator*", Electronics Letters, vol.33(22), pp. 1883-1884, October 1997
- [Sher-99] S. Sherit, S. P. Leary, B. P. Dolgin, Y.Bar-Cohen, "*Comparison of the Mason and KLM Equivalent Circuits for Piezoelectric Resonators in the Thickness Mode*", IEEE Ultrasonics Symposium 1999.
- [Sin-01] Rajarishi Sinha, Christiaan J. J. Paredis, Vei-Chung Liang, Pradeep K. Khosla, "*Modeling and Simulation Methods for Design of Engineering Systems*", Journal of Computing and Information Science in Engineering, vol. 1, pp. 84-91, March 2001.
- [Smi-94] J. G. Smits and A. Ballato, "*Dynamic admittance matrix of piezoelectric cantilever bimorphs*," Journal of Microelectromech. Syst., vol. 3, pp. 105–112, September 1994.

- [Ste-03] T. Sterken et al., "An Electret-Based Electrostatic  $\mu$ -Generator", In Proc. of Transducers'03, Boston, Etat-Unis, June 8-12, 2003.
- [Str-67] J. C. Strauss et al., "The SCI Continuous System Simulation Language (CSSL)", Simulation, vol. 9, pp. 281–303, 1967.
- [SAB] Saber. Synopsys Inc. : <http://www.synopsys.com/products/mixedsignal/saber/saber.html>
- [Sch-02] P. Schneider, A. Schneider, P. Schwarz, "A modular approach for simulation-based optimization of MEMS", Microelectron. J., no. 33, pp. 29–38, 2002.
- [Sem-01] F. Semon et al., "Epitaxy of AlN and GaN thin films on silicon or sapphire for the development of high frequency SAW devices", Ann. Chim. Sci. Mat., vol. 26(177), 2001.
- [Shi-03] A.W.M. (Jos) van Schijndel, "Integrated building physics with FEMLAB SIMULINK/MATLAB", Eighth International IBPSA Conference, Eindhoven, Netherlands, pp. 1177-1184, August 11-14, 2003.
- [Sim-05] SimPowerSystems 4.0 Reference, "Improving simulation performance : How SimPowerSystems works".
- [Sna-04] S. Snaidero, "Modélisation multidisciplinaire VHDL-AMS de système complexes : vers le prototypage virtuel", thèse en microélectronique, Université Louis Pasteur de Strasbourg, 2004.
- [SOF] [http://www.softmems.com/mems\\_pro.html](http://www.softmems.com/mems_pro.html)
- [SON] Catalogue Sony des batteries Lithium-ion batteries.  
<http://products.sel.sony.com/semi/PDF/Catalog-e.pdf>
- [SPE] Spectre reference manual 5.0.  
[http://lsmwww.epfl.ch/EDA/eda\\_tools/Cadence/doc/spectreref\\_5.0.pdf](http://lsmwww.epfl.ch/EDA/eda_tools/Cadence/doc/spectreref_5.0.pdf), June 2003.
- [SPI] SPICE from the [EECS Department of the University of California at Berkeley](http://bwrc.eecs.berkeley.edu/Classes/IcBook/SPICE/),  
<http://bwrc.eecs.berkeley.edu/Classes/IcBook/SPICE/>
- [Spr-06] D. Spreemann et al., "Tunable transducer for low frequency vibrational energy scavenging", In Proc. of Eurosensors XX, pp. 132-133, Goteborg, Suede, 17-20 September, 2006.

## T

- [Ton-76] E. Tonti, "The reason for analogies between physical theories", Appl. Math. Modeling 1, pp. 37-50, 1976.
- [Tis-04] F. Tissafi Drissi, "Méthode et outils de synthèse pour systèmes multi-domaine", thèse en dispositifs de l'électronique intégrée, Ecole Centrale de Lyon, 2004.
- [Tri-85] Triengo, M. J. L., and Bos, A. M., "Modeling the Dynamics and Kinematics of Mechanical Systems with Multibond Graphs", J. Franklin Inst., vol. 319, pp. 37–50, 1985.

## V

- [Van-97] J. E. Vandemeer, "Nodal design of actuators and sensors (NODAS)", M.S. thesis, Dept. Elect. Comput. Eng., Carnegie Mellon Univ., Pittsburgh, PA, 1997.
- [Verilog] Verilog, IEEE Standard 1364-2001.
- [Voi-97] P. Voigt, G. Wachutka, "Electro-fluidic microsystem modeling based on Kirchhoffian network theory", TRANSDUCERS '97, International Conference on Solid-State Sensors and Actuators, pp. 1019-1022, Chicago, June 1997.

- [Voi-98] P. Voigt, G. Schrag, and G. Wachutka, "*Microfluidic system modelling using VHDL-AMS and circuit simulation*", *Microelectron. J.*, vol. 29, n° 11, pp. 791–797, 1998.

## W

- [War-01] B. Warneke, B. Atwood, K. S. J. Pister, "*Smart dust mote forerunners*", MEMS 2001 : 14th Annual Int. Conf. on MicroElectroMechanical Systems, Interlaken- Switzerland, January 2001.
- [Wil-95] C.B. Williams, R.B. Yates, "*Analysis of a micro-electric generator for Microsystems*", *Transducers 95/Eurosensors IX*, pp. 369-372, 1995.

## X

- [XBO] <http://www.xbow.com/Products/productsdetails.aspx?sid=62>

## Y

- [Yan-89] A. T. Yang and S. M. Kang, "*iSMILE : A novel circuit simulation program with emphasis on new device model development*", in *Proc. 26<sup>th</sup> ACM/IEEE Design Automation Conf.*, pp. 630–633, 1989.

## Z

- [Zei-00] Zeigler, B. P., Praehofer, H., and Kim, "*Theory of Modeling and Simulation : Integrating Discrete Event and Continuous Complex Dynamic Systems*", 2nd ed. Academic Press, T. G., 2000.
- [Zha-01] X. F. Zha, H. Du, "*Mechanical systems and assemblies modeling using knowledge-intensive Petri nets formalisms*", *J. Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, pp. 145-171, April 2001.
- [Zha-04] T. Zhang, K. Chakrabarty, Richard B. Fair, "*Behavioral Modeling and Performance Evaluation of Microelectrofluidics-Based PCR Systems Using SystemC*", *IEEE Transactions on computer-aided design of integrated circuits and systems*, vol. 23, n° 6, pp. 843-858, June 2004
- [Zen-04] A. Zenati, S. Basrour, B. Courtois, "*Modélisation et simulation au niveau système de Micro Système auto alimenté (SPMS)*", JNRDM, Marseille, 2004.
- [Zen-05] A. Zenati, Y. Ammar, K. Matou, S. Basrour, "*Global simulation and cosimulation of self powered microsystems*", *Proceedings symposium on Design, Test, Integration and Packaging of MEMS/MOEMS*, Montreux/Switzerland, 1-3 June 2005.

# Annexe A : Comparaison entre VHDL-AMS et Verilog AMS

---

Le tableau ci-dessous est un récapitulatif des différentes fonctionnalités supportées par les deux langages de description comportementale :

Fonctions supportées	VHDL-AMS	Verilog-AMS
Langage standard	+	-
Systèmes analogiques et à signaux mixtes et multi domaines	+	+
Modélisation comportementale, structurelle	+	+
Temps continu	+	+
Équations mathématiques : implicite & explicite	+	+
discontinuités	+	+
Lois de conservation d'énergie	+	+
Description en flot de signaux	+	+
Interface automatique entre les signaux mixtes	-	+
Définition formelle pour l'initialisation à signaux mixtes	+	-
Analyse temporel : continue, petits signaux, bruit	+	+
Modèles prêts établis (bibliothèque)	-	+
Compatibilité avec SPICE (netlist)	-	+
Modularité (sous-systèmes et interfaces compilés séparément)	+	-

**Tableau IV-1 : VHDL-AMS vs. Verilog-AMS**

# Annexe B : Analogie entre les différents domaines

Dans le Tableau IV-2, on donne les analogies entre les différents domaines :

Domaine énergétique	Effort	Flux
Électrique	Courant $i[A]$	Tension $u [V]$
Magnétique	Taux du flux $\dot{\phi} [Wb.s^{-1}]$	Voltage $\dot{\mathcal{E}}_w [A]$
Thermique	Flot $\dot{\mathcal{E}} [J.K^{-1}.s^{-1}]$	température $v [K]$
Fluidique ou Acoustique	Flot $Q [m^3.s^{-1}]$	Pression $p [N.m^{-2}]$
Mécanique translatore	Force $F [N]$	Vitesse $v [m.s^{-1}]$
Mécanique rotative	moment de force $\tau [N.m]$	Vitesse Angulaire $\omega [rad.s^{-1}]$

**Tableau IV-2 : Analogies entre les différents domaines**

Élément	Variables de l'effort		Variables du flux	
	Électrique	Mécanique	Électrique	Mécanique
Capacité	$i = C \frac{du}{dt}$	$f = m \frac{dv}{dt}$	$u = \frac{1}{C} \int i dt$	$f = k \int v dt$
Inductance	$i = \frac{1}{L} \int u dt$	$f = k \int v dt$	$u = L \frac{di}{dt}$	$f = m \frac{dv}{dt}$
Résistance	$i = \frac{1}{R} u$		$u = Ri$	$f = Bv$
Conductance	$i = Gu$	$f = Bv$	$u = \frac{1}{G} i$	

\* K : constante de raideur, m : masse, B : coefficient d'amortissement

**Tableau IV-3 : Analogies des équations entre le domaine électrique et mécanique**

# Annexe C : Définition de la communication sans fil

---

Les systèmes sans fil profitent des évolutions technologiques et des possibilités émergentes de l'intégration des modules de communications à faible consommation et des capteurs, du stockage d'énergie et du traitement de données. Une nouvelle direction dans la conception touche l'extension des connexions sans fil à des microsystèmes embarqués et bon marché, qui sont destinés à de larges applications. Une telle configuration de communication sans fil élimine le besoin des réseaux interconnectés, qui sont coûteux et permettent le déploiement des microsystèmes dans l'environnement [Hil-02] [Rab-02] [War-01]. Ces microsystèmes utiliseront les techniques d'interconnexion de réseau d'égal à égal<sup>49</sup>. La Figure IV-1 montre le déploiement d'un réseau de nœuds sans fils. On peut imaginer qu'un tel réseau soit muni de microsystèmes qui mesurent les vibrations dans différents champs d'application et que les informations peuvent être transmises et traitées dans des centres spécialisés.

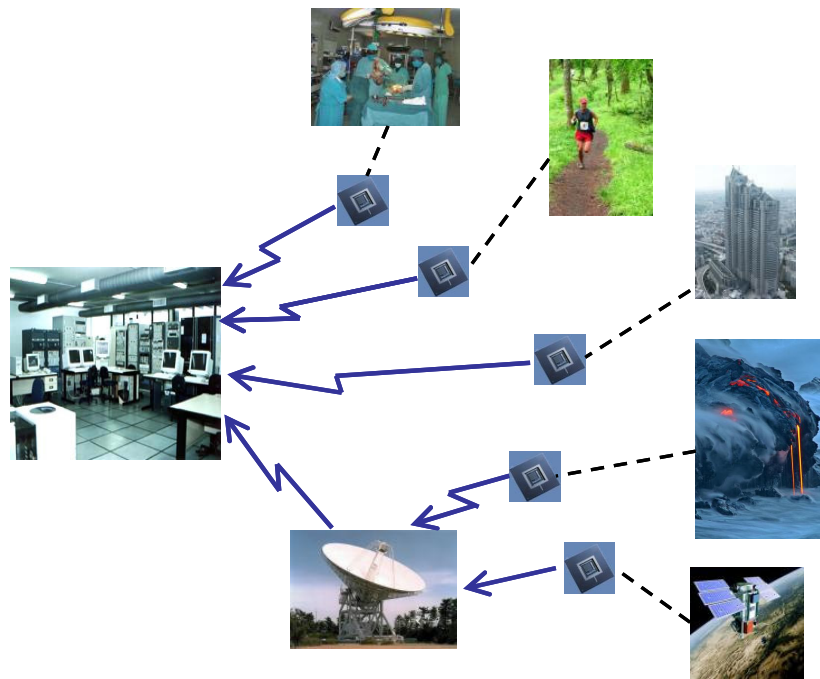


Figure IV-1 : Déploiement d'un réseau sans fil

## Génération de puissance

Les dispositifs sans fil sont conçus habituellement pour fonctionner avec des batteries. Cependant, la technologie des batteries et la consommation des circuits limitent la taille et la durée de leur autonomie.

---

<sup>49</sup> Peer to peer networking



Ceci rend la substitution des batteries épuisées impossible dans certaines applications. A ce propos, il s'avère nécessaire de trouver de nouvelles méthodes d'alimentation pour les microsystemes sans fils. Et par conséquent, l'énergie ambiante disponible dans l'environnement dans lequel se trouve le microsysteme, présente une nouvelle alternative pour remplacer ces batteries encombrantes.

Plusieurs formes d'énergie ambiantes [Rou-03] ont été explorées tel que l'énergie solaire, les champs magnétiques, les gradients thermiques, les flux fluidiques, et l'énergie produite par le corps humain; notamment les vibrations mécaniques qui constituent une source d'énergie électrique potentielle une fois qu'elles sont récoltées.

Au moment où le procédé de la conversion des vibrations en électricité est devenu accessible par la technologie des MEMS, les nouvelles avancées dans la conception des circuits VLSI à faible consommation ont aussi permis d'atteindre de faibles puissances de quelques microWATTS. Tout cela mène à des solutions de conception des microsystemes autonomes qui peuvent être définies pour remplacer les batteries conventionnelles (batteries en couches minces : diminution du volume) ou les interconnexions encombrantes (câblages) [Men-01].

Les travaux de [Kym-98] [Gly-01] [Rou-03] invoquent les matériaux piézoélectriques comme d'excellents candidats pour la récolte de l'énergie à partir des vibrations présentes dans l'environnement. Ces matériaux génèrent les charges électriques sous l'effet des vibrations mécaniques, et engendre des déformations de leur structure à la présence d'un champ électrique.

# Annexe D : Modèle du générateur dans SPICE

Le modèle du générateur de puissance raffiné de la section est Chapitre III-A.2.2.2 est aussi implémenté sous SPICE afin de vérifier la modélisation avec les éléments SimPowerSystems.

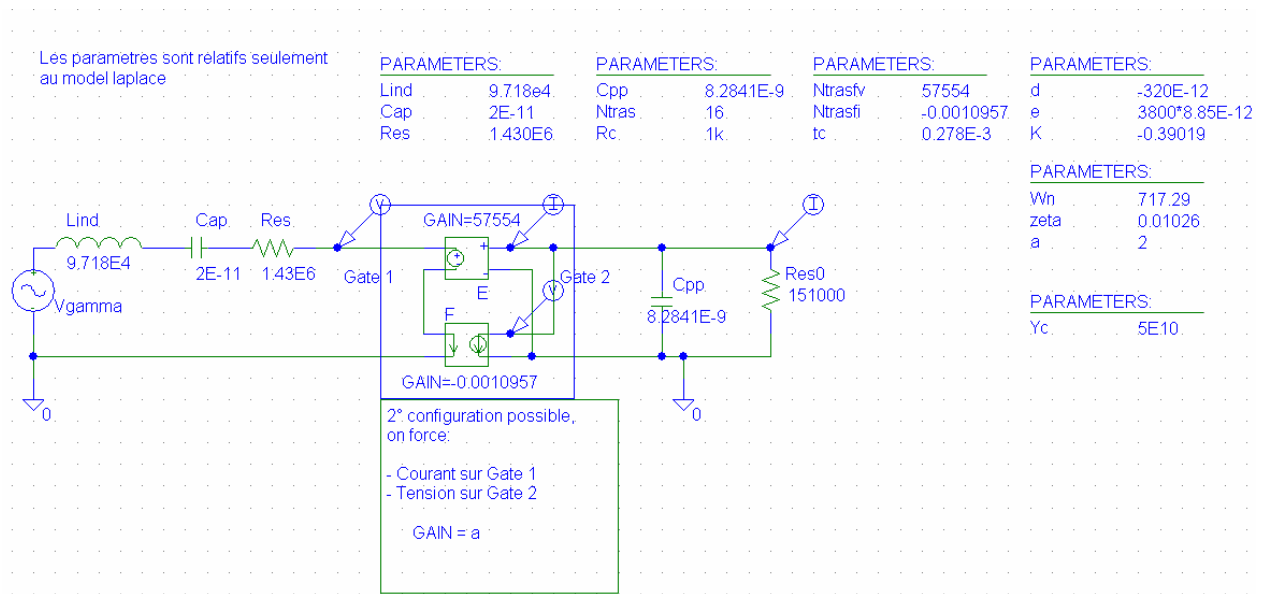


Figure IV-2 : Modèle électrique du générateur de puissance sous SPICE

# Annexe E : Paramètres du modèle unidimensionnel

---

```

Z=24e6
%%%%Paramètres géométriques
Bp=100e-6
Bm=110e-6
Lm=100e-6
Hm=100e-6
Lp=400e-6
Hp=1e-6
Hs=5e-6
xi=0
%%%%Propriétés des matériaux
%%%%Substrat
Gam_s=0
pert_m_s=1
%%%%Densité
rho_si=2330
m=Hm*Bm*Lm*rho_si
%%%%matrice
c11s=166e9*(pert_m_s)
c13s=64e9*(pert_m_s)
c33s=166e9*(pert_m_s)
%%%%Piézo
Gam_p=3.8e-3
tand=0*3e-2
pert_m_p=1
pert_d=1
%%%%matrice
c11p=132e9*(pert_m_p)
c13p=73e9*(pert_m_p)
c33p=115e9*(pert_m_p)
eps=1160*8.8542e-12*(pert_d)
e33=14.1
e31=-4.1
Ain=10
C=Bp*Lp*eps/Hp
%%%%
Yp_eff=c11p-c13p^2/c33p
Ys_eff=c11s-c13s^2/c33s
eps_eff=eps+((e33^2)/c33p)
e31_eff=e31-c13p*e33/c33p
%%%%

```

```

a1=(Ys_eff*Hs*Hs+Yp_eff*Hp*(2*Hs+Hp))/(2*(Ys_eff*Hs+Yp_eff*Hp))
z1=-a1
z2=Hs-a1
z3=z2+Hp
%%%
L=Lp+(Lm/2)
%%%%%%%%
Leq=sqrt((L^2)-(L*Lp)+((Lp^2)/3))
%%%%%%%%%
Dg=(Ys_eff*((z2^3)-(z1^3))/3)+(Yp_eff*((z3^3)-(z2^3))/3)
%%Dg=abs(Dg)
beta=e31_eff*Hp*(2*z2+Hp)/2/eps_eff;
Dg2=Dg+(e31_eff^2*((z3^3)-(z2^3))/3/eps_eff)-(eps_eff*(beta^2)/Hp)

zeta=beta*eps_eff/Hp/Dg2

J=m*((Lm^2)+(Hm^2))/3

```

# Annexe F : Etude analytique du circuit de récolte d'énergie

Dans un premier temps, les éléments dissipatifs du contrôleur et du générateur sont considérés négligeables. Le circuit équivalent est donc celui représenté dans la Figure IV-1.

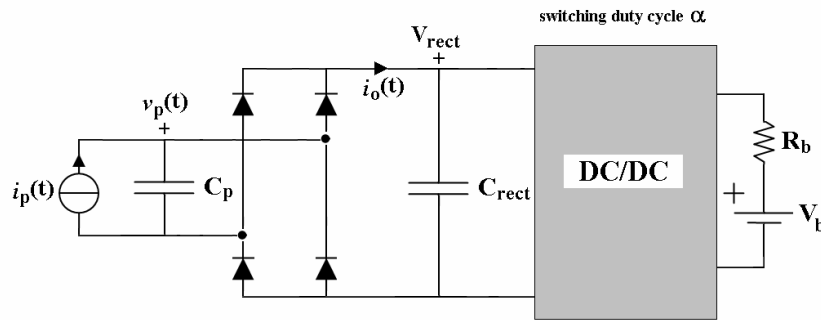


Figure IV-1 : La représentation du système complet.

Les caractéristiques de l'interrupteur et des diodes sont aussi considérées idéales.

Deux événements indépendants changent la configuration du circuit :

- L'inversion du courant généré par l'élément piézoélectrique, causée par le passage exclusif en conduction des deux couples de diodes du pont de Graëtz,
- Le changement de l'état de l'interrupteur dans le convertisseur à découpage.

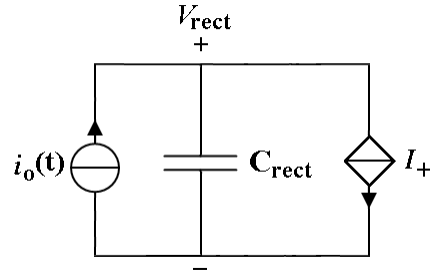
La fréquence de fonctionnement du contrôleur est plus élevée que celle du générateur de puissance.

## Redresseur AC/DC

Le dimensionnement de la capacité du redresseur AC/DC permet d'obtenir une tension à l'entrée du convertisseur DC/DC moins ondulée après le redressement.

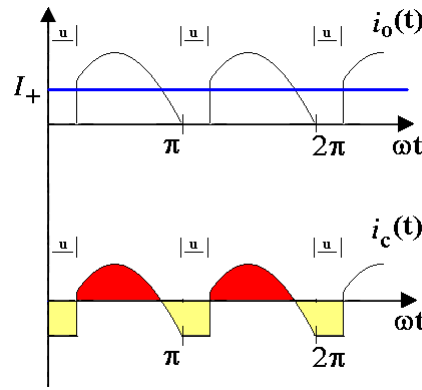
## Capacité du redresseur

Le circuit équivalent du système complet est représenté dans la Figure I-2. Dans cette figure  $i_o(t)$  et  $I_+$  sont les fonctions introduites dans l'analyse du transfert de puissance. En fait si la période du convertisseur est suffisamment petite, les perturbations seraient coupées par une capacité dimensionnée pour filtrer les basses fréquences de l'élément piézoélectrique. Le courant dans la charge est bien redressé si le condensateur absorbe toute l'ondulation du courant généré.



**Figure I-2 : Le circuit équivalent du système complet.**

Nous reportons sur la Figure I-3 les allures du courant dans la capacité de redressement et le courant de sortie.



**Figure I-3 : Les courants de sortie et dans la capacité.**

L'ondulation de la tension de sortie est donnée par :  $\Delta V_{rect} = \frac{\Delta Q}{C_{rect}}$

La valeur de la variation de la quantité de charges est donnée par :  $\Delta Q = \int I_+ - i_o(t) dt$

L'expression de  $i_o(t)$ , en supposant que  $C_{rect} \gg C_p$  :

$$\begin{aligned} i_o(t) &= 0, & 0 < \omega t < u \\ i_o(t) &\approx i_p(t) = I_p \sin(\omega_p t), & u < \omega t < \pi \end{aligned}$$

Lorsque le courant dans le condensateur n'est pas nul :

$$\omega t = \pi - \arcsin\left(\frac{I_+}{I_p}\right)$$

L'expression de la quantité de charge devient alors :

$$\Delta Q = \frac{1}{\omega_p} \int_0^u I_+ d\omega_p t + \frac{1}{\omega_p} \int_{\pi - \arcsin\left(\frac{I_+}{I_p}\right)}^{\pi} I_+ - I_p \sin(\omega_p t) d\omega_p t$$

$$\Delta Q = \frac{I_+}{\omega_p} u + \frac{I_+}{\omega_p} \arcsin\left(\frac{I_+}{I_p}\right) - \frac{I_p}{\omega_p} \left[ -1 - \cos\left(\pi - \arcsin\left(\frac{I_+}{I_p}\right)\right) \right]$$

$$\Delta Q = \left[ u + \arcsin\left(\frac{I_+}{I_p}\right) \right] \frac{I_+}{\omega_p} + \left[ 1 - \cos\left(\arcsin\left(\frac{I_+}{I_p}\right)\right) \right] \frac{I_p}{\omega_p}$$

Si on suppose l'angle résultant de la fonction arcsin dans le premier quadrant (les deux courants sont positifs et le numérateur est inférieur au dénominateur) on obtient :

$$\Delta Q = \left[ u + \arcsin\left(\frac{I_+}{I_p}\right) \right] \frac{I_+}{\omega_p} + \left[ I_p - \sqrt{I_p^2 - I_+^2} \right] \frac{1}{\omega_p}$$

La relation pour le calcul de  $u$  est :

$$\cos(u) = 1 - \frac{2V_{rect}\omega_p C_p}{I_p}$$

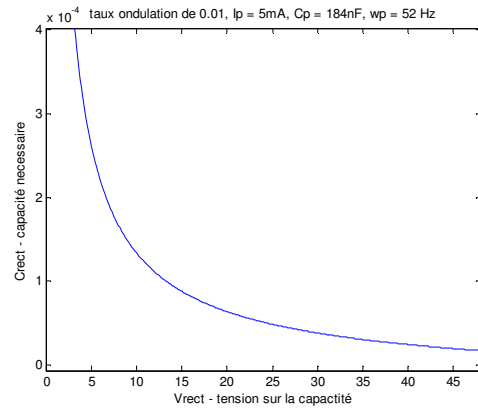
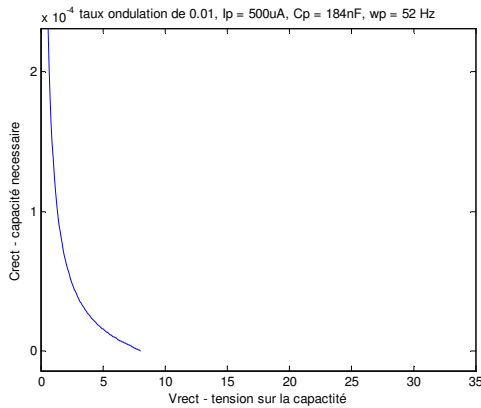
Le courant moyen est fonction de la tension aux bornes de la capacité :

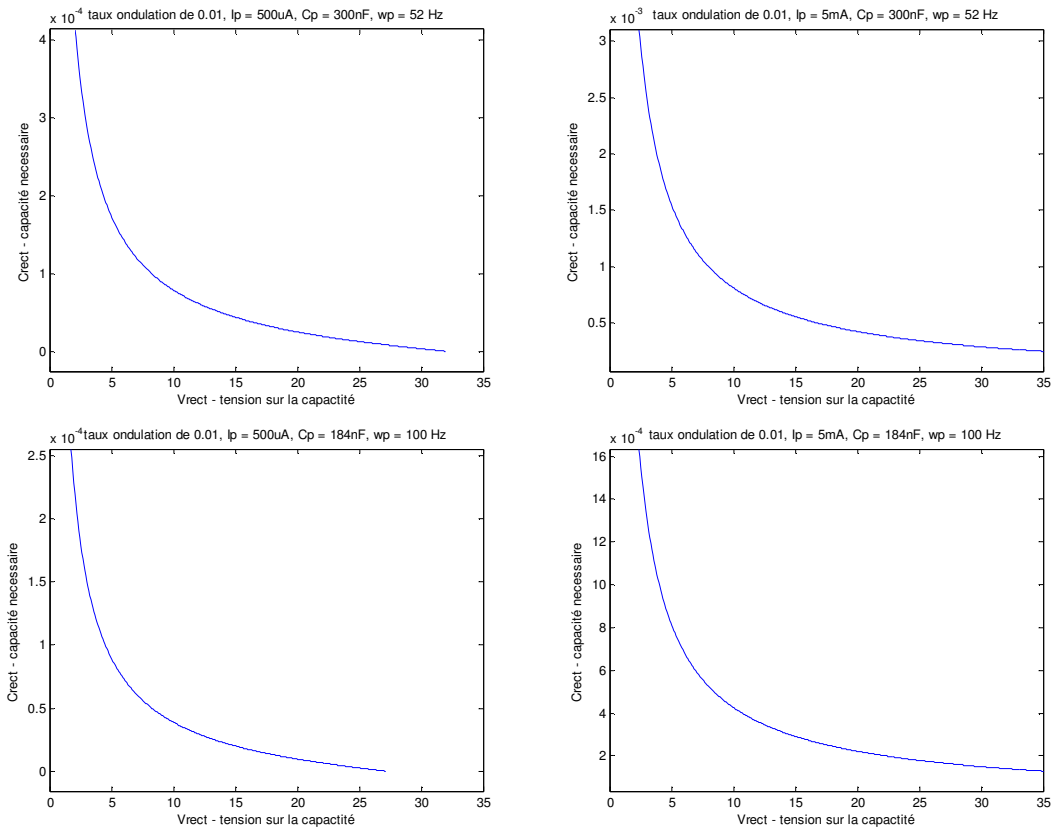
$$I_+ = \frac{2I_p}{\pi} - \frac{2V_{rect}\omega C_p}{\pi}$$

En définissant un taux d'ondulation  $\beta$ , on obtient :

$$C_{rect} = \frac{\Delta Q(V_{rect})}{\Delta V_{rect}} = \frac{1}{\beta} \frac{\Delta Q(V_{rect})}{V_{rect}}$$

Voici quelques courbes extraites en utilisant les routines MATLAB implémentées pour les relations ci-dessus :





**Figure I-4 : Les résultats de dimensionnement de la capacité  $C_{rect}$ .**

## Convertisseur DC/DC

### L'abaisseur en mode de conduction continu (buck\_CCM)

On utilisera d'ici après les notations et les hypothèses utilisées dans l'analyse du fonctionnement des convertisseurs DC/DC. En particulier  $i_o$  et  $v_o$  représentent maintenant les valeurs de courant et tension en sortie du convertisseur, et non plus de l'élément piézoélectrique.

La variation du courant dans l'inductance permet de dimensionner la valeur de l'inductance  $L$  :

$$\Delta i_L = I_M - I_m = \frac{v_i - v_o}{L} t_1$$

$$\Delta i_L = \frac{(1-\alpha)v_o}{Lf_s} = \frac{\alpha(1-\alpha)v_i}{Lf_s}$$

Les hypothèses utilisées jusqu'à maintenant considéraient la tension en sortie constante. En réalité, une petite ondulation due à la charge est toujours présente.

Les relations pour le condensateur sont :

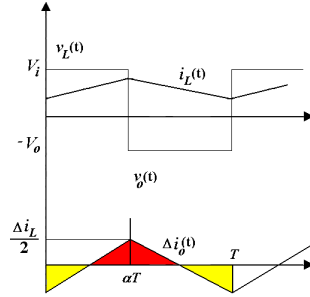
$$i_c(t) = i_L(t) - i_o$$



$$i_c(t) = C \frac{dv_o(t)}{dt}$$

Le courant dans la charge  $i_o$  est considéré comme constant afin que l'ondulation du courant soit absorbée totalement par le condensateur. En régime permanent, le courant moyen sur le condensateur est nul.

Le courant de l'inductance et la tension de la capacité d'un convertisseur buck\_CCM sont représentés sur la Figure I-5.



**Figure I-5 : Les courbes du courant dans l'inductance et la tension sur la capacité du convertisseur.**

La surface du triangle rouge (en régime égale à cela du jaune) représente la quantité de charges accumulées quand le courant est supérieur à la valeur moyenne. Cette quantité de charges est égale à :

$$\Delta Q = \frac{1}{2} \cdot \frac{\Delta i_L}{2} \cdot \frac{T}{2}$$

A partir de cette relation l'ondulation de la tension est :

$$\Delta v_o = \frac{\Delta Q}{C} = \frac{\Delta i_L}{8Cf_s} \rightarrow \Delta v_o = \frac{(1-\alpha)v_o}{8LCf_s^2} = \frac{\alpha(1-\alpha)v_i}{8LCf_s^2}$$

Ainsi, nous pouvons déduire à partir de ces relations les valeurs de  $L$ ,  $C$  et  $f_s$  en fonction des taux d'ondulation  $\beta_i$  et  $\beta_v$ . Pour les hautes fréquences, les valeurs de  $L$  et  $C$  peuvent être basses, mais sont liées aux contraintes de fabrication de ces dispositifs et des matériaux utilisés.

$$|i_L(t) - \langle i_L(t) \rangle| \leq \frac{\Delta i_L}{2} = \beta_i \langle i_L(t) \rangle \quad |v_C(t) - \langle v_C(t) \rangle| \leq \frac{\Delta v_C}{2} = \beta_v \langle v_C(t) \rangle$$

Finalement, on obtient :

$$L = \frac{\alpha(1-\alpha)v_i}{f_s \Delta i_L} = \frac{1}{2\beta_i} \frac{\alpha(1-\alpha)v_i}{f_s i_o} \quad C = \frac{(1-\alpha)v_o}{8Lf_s^2 \Delta v_o} = \frac{1}{2\beta_v} \frac{\alpha(1-\alpha)v_i}{8Lf_s^2 v_o}$$

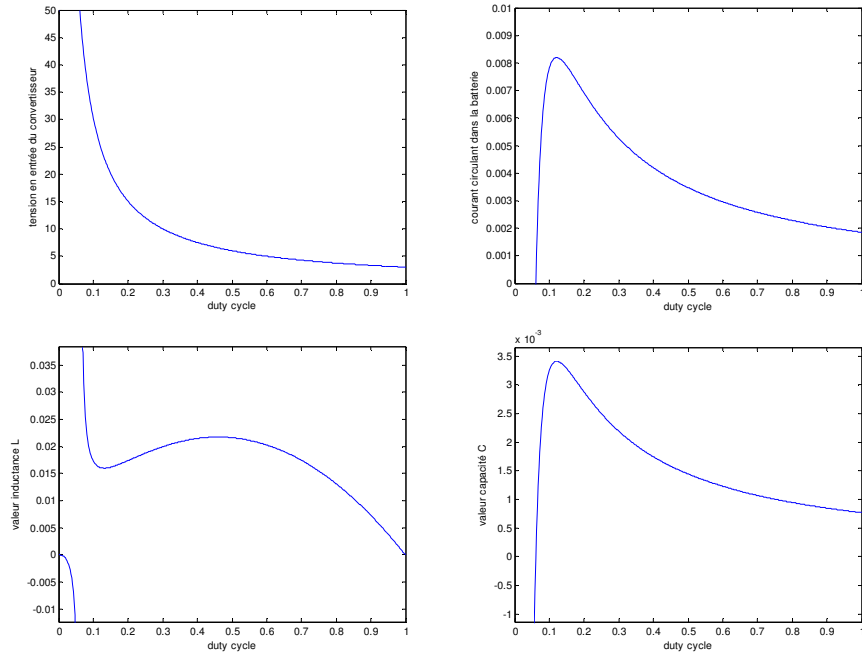
Les résultats représentés dans les courbes suivantes sont obtenus à l'aide des routines MATLAB en implémentant les relations finales de  $v_i$  et  $i_i$  (tension et courant en entrée), de  $v_o$  et  $i_b$  (tension et courant en sortie) en fonction du rapport cyclique  $\alpha$  trouvés pendant l'analyse du fonctionnement.

Ces résultats ont été obtenus pour les paramètres suivants :

Pour le générateur piézoélectrique :  $I_p = 3.1\text{mA}$ ,  $C_p = 0,184\mu\text{F}$ ,  $\omega_p = 338\text{ rad s}^{-1}$

Pour la batterie :  $V_p = 3\text{V}$ ,  $R_b = 0,1\Omega$

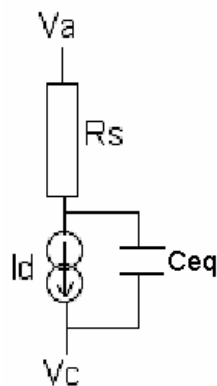
La fréquence du commutateur choisie est de 100 KHz,  $\beta_I = 10\%$  et  $\beta_V = 0.0001\%$ .



## Annexe G : Modèle SPICE de la diode

<i>Nom</i>	<i>Paramètre</i>	<i>Unité</i>	<i>Valeur par défaut</i>
<b>IS</b>	Courant de saturation	A	$10^{-14}$
<b>RS</b>	Résistance en série	$\Omega$	0
<b>N</b>	Coefficient d'émission	-	1
<b>TT</b>	Temps de transit	s	0
<b>CJO</b>	Capacité de jonction initiale	F	0
<b>VJ</b>	Tension de la jonction	V	1
<b>M</b>	Coefficient de calibration	-	0.5
<b>EG</b>	Energie du Gap	eV	1.11
<b>XTI</b>	Exposant de la température dans l'expression du courant de saturation	-	3
<b>KF</b>	Coefficient du bruit de vibrations	-	0
<b>AF</b>	Exposant du bruit de vibrations	-	1
<b>FC</b>	Coefficient pour la formule de la capacité de jonction	-	0.5
<b>BV</b>	Tension de claquage de la diode	V	$\infty$
<b>IBV</b>	Courant à la tension de claquage	A	$10^{-3}$
<b>TNOM</b>	Température à la quelle les paramètres ont été mesurés	°C	27

### Circuit équivalent



### Courant $I_D$

$$I_D = IS \left[ \exp \left( \frac{V_D}{NV_T} \right) - 1 \right]$$

### Capacités :

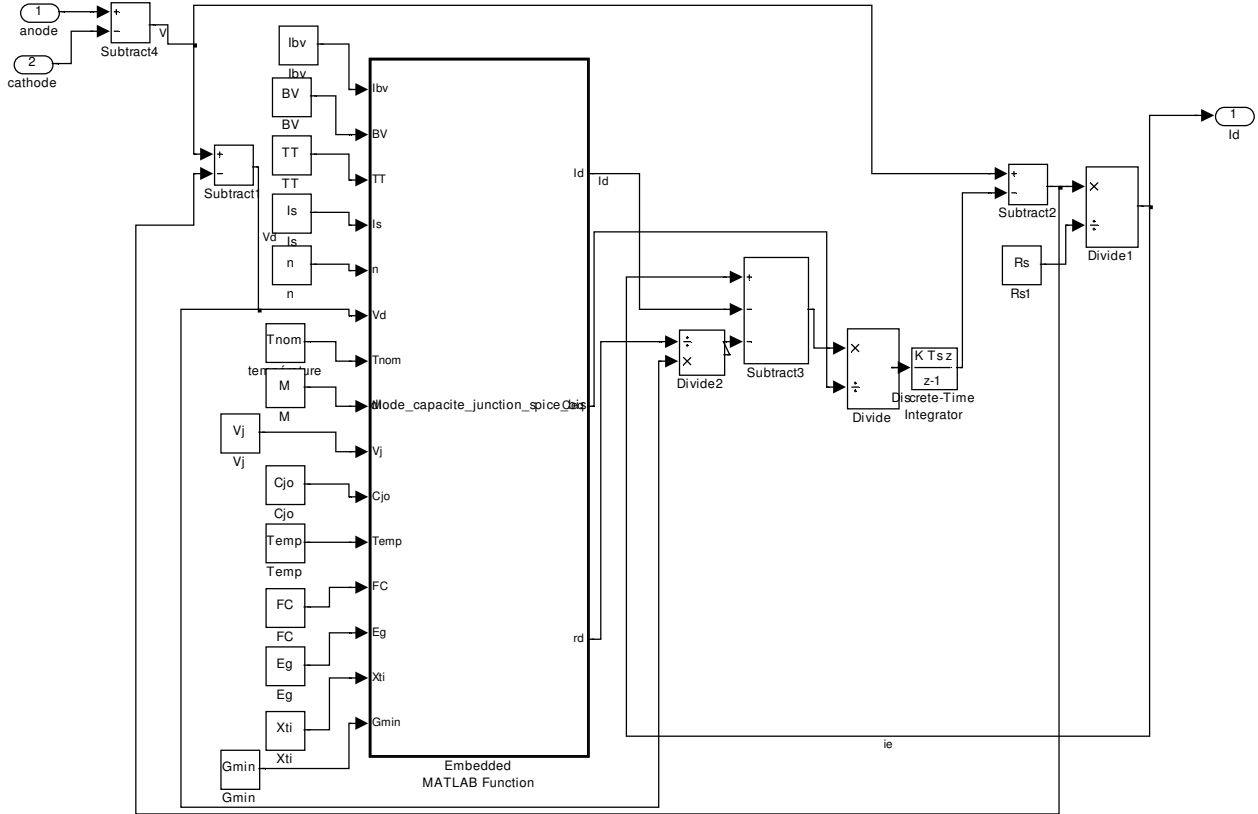
$$C_{diffusion} = TT \frac{\partial}{\partial V_D} (I_{Forward}) \approx \frac{TT}{r_D}$$

$$C_{jonction} = \begin{cases} CJO \left(1 - \frac{V_D}{VJ}\right)^{-M} & \text{pour } V_D \leq FC.VJ \\ CJO (1 - FC)^{-(1+M)} \left[1 - FC(1+M) + M \frac{V_D}{VJ}\right] & \text{pour } V_D \geq FC.VJ \end{cases}$$

### Effet de la température

$$IS(T) = IS \left(\frac{T}{T_{nom}}\right)^{\frac{XTI}{N}} \exp \left[ \left(\frac{T}{T_{nom}} - 1\right) \frac{EG}{N.V_T} \right]$$

$$CJO(T) = CJO \left\{ 1 + M \left[ 0.0004(T - T_{nom}) + \left(1 - \frac{VJ(T)}{VJ}\right) \right] \right\}$$



(b) Modèle au niveau circuit de la diode

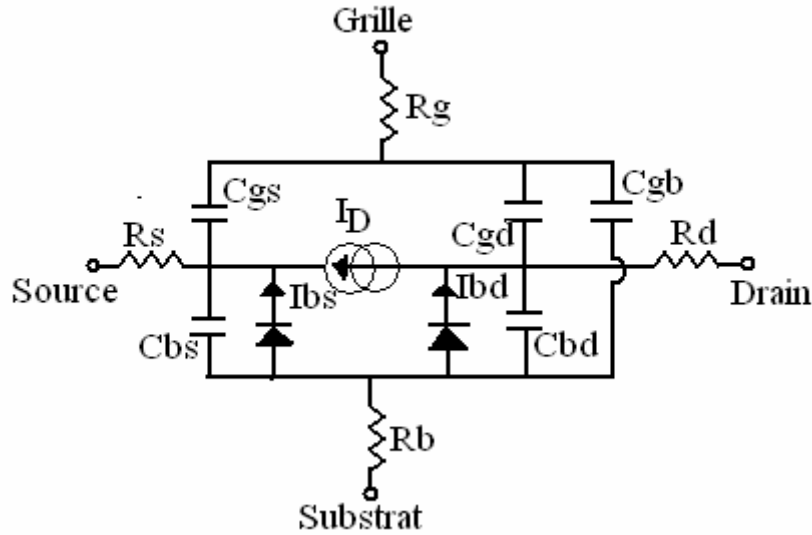
Figure I-6 : Modélisations sous Simulink de la diode

# Annexe H : Modèle du transistor MOS

Paramètres communs à tous les niveaux			
<i>Nom</i>	<i>Description du paramètre</i>	<i>Unité</i>	<i>Valeur par défaut</i>
L	Longueur du canal	m	100μ
W	Largeur du canal	M	100μ
RD	Résistance ohmique du drain	Ω	0
RS	Résistance ohmique de la source	Ω	0
RG	Résistance ohmique de la grille	Ω	0
RB	Résistance du substrat	Ω	0
CBD	Capacité de la jonction substrat-drain	F	0
CBS	Capacité de la jonction substrat-source	F	0
IS	Courant de saturation du courant de la jonction du substrat	A	10 <sup>-14</sup>
N	Coefficient d'émission de la jonction du substrat	-	1
PB	Potentiel de la jonction du substrat	V	0.8
CGSO	Capacité de chevauchement grille-source	F/m	0
CGDO	Capacité de chevauchement grille-drain	F/m	0
CGBO	Capacité de chevauchement grille-substrat	F/m	0
CJ	Capacité de la jonction du substrat	F/m <sup>2</sup>	0
MJ	Coefficient de calibration de la jonction du substrat	-	0.5
TT	Temps de transit de la jonction du substrat	s	0
KF	Coefficient du bruit des vibrations	-	0
AF	Exposant du bruit de vibrations	-	1
FC	Coefficient pour la formule de la capacité de jonction	-	0.5
TNOM	Température à la quelle les paramètres ont été mesurés	K	300

Paramètres spécifiques du level 1			
<i>Nom</i>	<i>Description des paramètres</i>	<i>Unité</i>	<i>Valeur par défaut</i>
VTO	Tension de seuil	V	0
KP	Paramètre de transconductance	A/V <sup>2</sup>	2.10 <sup>-5</sup>
GAMMA	Paramètre de seuil du substrat	V <sup>0.5</sup>	0
PHI	Potentiel de surface	V	0.6
LAMBDA	Paramètre de modulation de la longueur du canal	1/V	0
TOX	Epaisseur de l'oxyde	m	10 <sup>-7</sup>
NSUB	Dopage du substrat	cm <sup>-3</sup>	0
LD	Diffusion latérale	m	0

Schéma électrique équivalent :



Calcul des courants :

$$I_{bs} = I_{ss} \exp \left[ \left( \frac{V_{BS}}{NV_T} - 1 \right) \right]$$

$$I_{bd} = I_{ds} \exp \left[ \left( \frac{V_{BD}}{NV_T} - 1 \right) \right]$$

- $I_D = 0$  si  $V_{GS} \leq 0$
- on peut supposer que  $I_D = 0$  si  $V_{GS} \leq V_{TH}$
- si  $V_{GS} \geq V_{TH}$  et que  $V_{DS} < V_{GS} - V_{TH}$  alors  $I_D = \frac{W_{eff}}{L_{eff}} K_p \left( V_{GS} - V_{TH} - \frac{V_{DS}}{2} \right) V_{DS} (1 + \lambda V_{DS})$
- si  $V_{GS} \geq V_{TH}$  et  $V_{DS} \geq V_{GS} - V_{TH}$  alors  $I_D = \frac{W_{eff}}{L_{eff}} K_p \frac{(V_{GS} - V_{TH})^2}{2} [1 + \lambda (V_{DS} - V_{TH})]$

$$V_{TO} = V_{T0} + GAMMA \left( \sqrt{PHI - V_{BS}} - \sqrt{PHI} \right)$$

$$L_{eff} = L - 2LD$$

$$KP = UP.C_{OX}$$

$$GAMMA = \frac{\sqrt{2q\epsilon_0\epsilon_{si}NSUB}}{C_{OX}}$$

$$PHI = 2V_T \ln \left( \frac{NSUB}{n_i} \right)$$

$$C_{OX} = \frac{\epsilon_0\epsilon_{ox}}{TOX}$$

$$n_i(T) = 1.4510^{10} \left( \frac{T}{TNOM} \right)^{1.5} \exp \left[ \frac{1.16 \left( \frac{T}{TNOM} \right) - EG(T)}{V_T} \right]$$

$$E_G(T) = 1.16 - \frac{0.000702 \times T^2}{T + 1108}$$

### Calcul des capacités

Si CBS=0 alors  $C_{BS} = X_{BSJ} AS.CJ + X_{BSS} PS.CJSW + TTG_{BS}$

Sinon  $C_{BS} = X_{BSJ} CBS + X_{BSS} PS.CJSW + TTG_{BS}$

Si CBD=0 alors  $C_{BD} = X_{BDJ} AD.CJ + X_{BDS} PD.CJSW + TT.G_{DS}$

Sinon  $C_{BD} = X_{BDJ} CBD + X_{BDS} PD.CJSW + TT.G_{DS}$

$$G_{BS} = \frac{I_{BS}}{NV_T} \text{ si } V_{BS} > 0 \text{ et } G_{BS} = 0 \text{ sinon}$$

$$G_{BD} = \frac{I_{BD}}{NV_T} \text{ si } V_{BD} > 0 \text{ et } G_{BD} = 0 \text{ sinon}$$

Si  $V_{BS} \leq FC.PB$  alors

$$X_{BSJ} = \left( 1 - \frac{V_{BS}}{PB} \right)^{-MJ}$$

$$X_{BSS} = \left( 1 - \frac{V_{BS}}{PBSW} \right)^{-MJSW}$$

Sinon

$$X_{BSJ} = (1 - FC)^{-(1+MJ)} \left[ 1 - FC(1 + MJ) + MJ \frac{V_{BS}}{PB} \right]$$

$$X_{BSS} = (1 - FC)^{-(1+MJSW)} \left[ 1 - FC(1 + MJSW) + MJSW \frac{V_{BS}}{PBSW} \right]$$

Si  $V_{BD} \leq FC.PB$

$$X_{BDJ} = \left( 1 - \frac{V_{BD}}{PB} \right)^{-MJ}$$

$$X_{BDS} = \left( 1 - \frac{V_{BD}}{PBSW} \right)^{-MJSW}$$

Sinon

$$X_{BDJ} = (1 - FC)^{-(1+MJSW)} \left[ 1 - FC(1 + MJ) + MJ \frac{V_{BD}}{PB} \right]$$





# Annexe I : Algorithme de contrôle

---

## Algorithme constant

```
library std;
  use std.standard.all;
library ieee;
  use ieee.std_logic_1164.ALL;
  use ieee.math_real.all;
entity algo_fixe is
  port(vin : in real;
        compteur:out real;
        pwm : out real;
  rst : out real);
end algo_fixe;

architecture drive_algo_fixe of algo_fixe is
  signal clk : std_logic:='1';

  begin
    clk <= not clk after 0.5 ms;
    plus_one:process(clk)

  variable alpha_out : real := 0.02;
  variable alpha_in : real := 0.0;
  variable vinb : real :=0.0;-- vin(t-1)
  variable vinac : real :=0.0;-- vin(t)
  variable dvin : real := 0.0;-- dérivée partielle de vin
  variable dalpha : real := 0.0;-- dérivée partielle du rapport cyclique alpha
  variable comp : real := 0.0;-- compteur d'impulsion de l'horloge clk
  variable reset : real :=0.0;-- remise à zéro du calcul de la moyenne
  variable temp : real :=0.0;-- variable intermédiaire pour le calcul de la valeur moyenne de vin
  constant pas : real := 0.02;--
  variable nb : real := 1.0;-- nombre d'échantillon
  variable k : real := 1.0;-- pas du changement du rapport cyclique

  begin
    if (clk'event and clk='1') then
      comp := comp + 1.0;
      reset := 0.0;

      if comp>(nb×2000.0) then
        nb := nb + 1.0;
        temp := vin/2000.0;
        vinb:=vinac;
        vinac:=0.0;
        vinac:= vinac+temp;
        reset:=1.0;
        alpha_in := alpha_out;
        alpha_out := alpha_out + k×pas;
        if vinac > 0.0 then

          dvin := (vinac-vinb);
```

```

        else
            dvin := (vinb-vinac);
        end if;

        dalpha:=(alpha_out-alpha_in);

        if (dvin >0.0 and dalpha > 0.0) OR (dvin <0.0 and dalpha < 0.0)) then
            k:=1.0;
        else
            k:=-k;
        end if;

        compteur<= comp;
    end if;

    rst <= reset;
end if;

if alpha_out >1.0 then

    alpha_out := 1.0;
elseif alpha_out <0.0 then
    alpha_out := 0.0;
end if;

    pwm <= alpha_out;
end process plus_one;
end drive_algo_fixe;

```

## Algorithme adaptatif -2

```

library std;
    use std.standard.all;
library ieee;
    use ieee.math_real.all;
entity algorithm is
    port(vin : in real;
         clk  : in real;
         pwm  : out real);
end super;

architecture drive_algorithm of super is
begin
    plus_one:process(clk)
        variable value : real := 0.01;-- valeur du pwm
        variable vinb : real :=0.0;-- vin(t-1)
        variable vinac : real :=0.0;-- vin(t)
        variable dvin : real := 0.0;-- dérivée de vin
        variable comp : real := 0.0;-- compteur d'impulsion de l'horloge clk (attendre 1000 pulses)
        variable temp : real :=0.0;-- variable intermédiaire pour le calcul de la valeur moyenne de vin
        variable k : real := 1.0;-- le pas du changement du rapport
        variable pos : real := 0.0;

        variable sign : real := 1.0;    -- le signe de k
    end process plus_one;
end drive_algorithm;

```

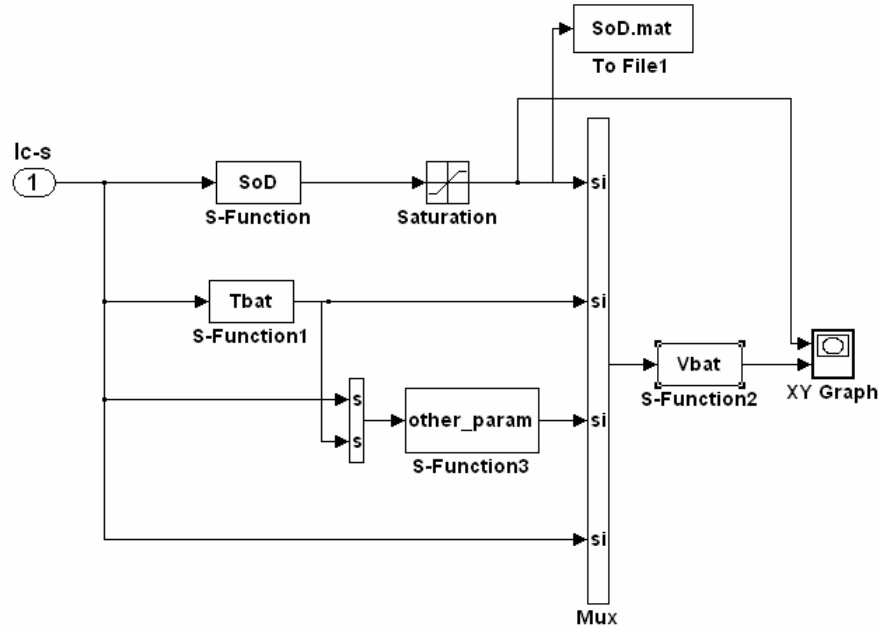
```

variable pas : real := 0.0007;
variable nb : real := 2000.00;
variable ip : real := 0.0;

begin
  if (clk'event and clk=1.0) then
    comp := comp + 1.0;
    if comp>(nb-1000.0) then
      temp := vin/1000.0;
      vinac:= vinac+temp;
    end if;
    if (comp = nb) then
      dvin := vinac-vinb;
      if dvin > 0.0 then
        if ip > 1.0 then
          k:=k×2.0;
        end if;
        ip:=ip+1.0;
      end if;
      if dvin < 0.0 then
        sign:= sign×(-1.0);
        k:=sign×0.5;
        pos:=pos+1.0;
        ip:=1.0;
        k:=k×2.0;
      end if;
      if dvin = 0.0 then
        k:=0.0;
      end if;
      value := value + k×pas;
      vinb:=vinac;
      vinac:=0.0;
      comp:=0.0;
    end if;
    if value >1.0 then
      value := 1.0;
    end if;
    if value <0.0 then
      value := 0.0;
    end if;
    pwm <= value;
  end process plus_one;
end architecture drive_algorithm;

```

# Annexe J : S-function pour le modèle de la batterie



```
function [sys,x0,str,ts] = SOD(t,x,u,flag)

global R Tamb Teta SOD T Capcrit dSOC1 dSOC2 Ibat

switch flag,
    %%%%%%%%%%%%%%%
    % Initialization %
    %%%%%%%%%%%%%%%
    case 0
        [sys,x0,str,ts] = mdlInitializeSizes;

    %%%%%%%%%%%%%%%
    % Derivatives %
    %%%%%%%%%%%%%%%
    case 1
        sys = mdlDerivatives(t,x,u,SOD);

    %%%%%%%%%%%%%%%
    % Output %
    %%%%%%%%%%%%%%%
    case 3
        sys = mdlOutputs(t,x,u,SOD);

    %%%%%%%%%%%%%%%
    % Update and Terminate %
    %%%%%%%%%%%%%%%
    case {2,4,9}
```

```

        sys = []; % do nothing

    otherwise
        error(['unhandled flag = ',num2str(flag)]);
    end

% end limintm
%
%=====
% mdlInitializeSizes
% Return the sizes, initial conditions, and sample times for the S-function.
%=====
%
function [sys,x0,str,ts] = mdlInitializeSizes

sizes = simsizes;
sizes.NumContStates = 1;
sizes.NumDiscStates = 0;
sizes.NumOutputs = 1;
sizes.NumInputs = 1;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;

sys = simsizes(sizes);
str = [];
x0 = 0.1;
ts = [0 0]; % sample time : [period, offset]

% end mdlInitializeSizes
%
%=====
% mdlDerivatives
% Compute derivatives for continuous states.
%=====
%
%% Etat de décharge:Calcul du SOD
function sys = mdlDerivatives(t,x,u,SOD)

    sys = u;

% end mdlDerivatives

%=====
% mdlOutputs
% Return the output vector for the S-function
%=====
%
function sys = mdlOutputs(t,x,u,SOD)

sys = x;
SOD = sys;

% end mdlOutputs

```

```

function [sys,x0,str,ts] = T(t,x,u,flag)

global R Tamb Teta SOD T Capcrit dSOC1 dSOC2 Ibat

R = 0.14;
Tamb = 22;
Teta = 19.6;

switch flag,
    %%%%%%%%%%%%%%%
    % Initialization %
    %%%%%%%%%%%%%%%
    case 0
        [sys,x0,str,ts] = mdlInitializeSizes;

    %%%%%%%%%%%%%%%
    % Update %
    %%%%%%%%%%%%%%%
    %case 2,
    % sys = mdlUpdate(t,x,u,T,Capcrit,dSOC1,dSOC2);

    %%%%%%%%%%%%%%%
    % Output %
    %%%%%%%%%%%%%%%
    case 3
        sys = mdlOutputs(t,x,u);

    %%%%%%%%%%%%%%%
    % Update and Terminate %
    %%%%%%%%%%%%%%%
    case {1,2,4,9}
        sys = []; % do nothing

    otherwise
        error(['unhandled flag = ',num2str(flag)]);
end

% end limintm

%
%=====
% mdlInitializeSizes
% Return the sizes, initial conditions, and sample times for the S-function.
%=====
%
function [sys,x0,str,ts] = mdlInitializeSizes

sizes = simsizes;
sizes.NumContStates = 0;
sizes.NumDiscStates = 0;
sizes.NumOutputs = 1;
sizes.NumInputs = 1;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;

sys = simsizes(sizes);
str = [];
x0 = [];

```

```

ts = [0 0]; % sample time : [period, offset]

% end mdlInitializeSizes

%=====
% mdlOutputs
% Return the output vector for the S-function
%=====
%
function sys = mdlOutputs(t,x,u)

% définitions des symboles

R = 0.14;
Tamb = 22;
Teta = 19.6;

%% Augmentation de la temperature

x = Tamb+(u.^2)×R×Teta;

sys = x ;

% end mdlOutputs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [sys,x0,str,ts] = other_param(t,x,u,flag)

global R Tamb Teta SOD Capcrit dSOC1 dSOC2 Ibat

R = 0.14;
Tamb = 22;
Teta = 19.6;

switch flag,
    %%%%%%%%%%%%%%%
    % Initialization %
    %%%%%%%%%%%%%%%
    case 0
        [sys,x0,str,ts] = mdlInitializeSizes;

    %%%%%%%%%%%%%%%
    % Update %
    %%%%%%%%%%%%%%%
    %case 2,
    % sys = mdlUpdate(t,x,u,T,Capcrit,dSOC1,dSOC2);

    %%%%%%%%%%%%%%%
    % Output %
    %%%%%%%%%%%%%%%
    case 3
        sys = mdlOutputs(t,x,u,Capcrit,dSOC1,dSOC2);

    %%%%%%%%%%%%%%%
    % Update and Terminate %
    %%%%%%%%%%%%%%%
    case {1,2,4,9}
        sys = []; % do nothing

    otherwise
        error(['unhandled flag = ',num2str(flag)]);
end

% end limintm

%
%=====
% mdlInitializeSizes
% Return the sizes, initial conditions, and sample times for the S-function.
%=====
%
function [sys,x0,str,ts] = mdlInitializeSizes

sizes = simsizes;
sizes.NumContStates = 0;
sizes.NumDiscStates = 0;
sizes.NumOutputs = 3;
sizes.NumInputs = 2;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;

sys = simsizes(sizes);
str = [];
x0 = [];

```



```

ts = [0 0]; % sample time : [period, offset]

% end mdlInitializeSizes

%=====
% mdlOutputs
% Return the output vector for the S-function
%=====
%
function sys = mdlOutputs(t,x,u,Capcrit,dSOC1,dSOC2)

% définitions des symboles

R = 0.14;
Tamb = 22;
Teta = 19.6;

%% Augmentation de la temperature

%%T = Tamb+(u.^2)×R×Teta;

%% Capacité critique

Capcrit = 0.0023×u(2)+0.0343;

%% Décalage du à la température

dSOC1 = -0.004×u(2)+0.092;

%% Décalage du au courant

dSOC2 = 0.049×u(1)-0.0343;

sys = [Capcrit dSOC1 dSOC2];

% end mdlOutputs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [sys,x0,str,ts] = batterie(t,x,u,flag)

global R Tamb Teta SOD T Capcrit dSOC1 dSOC2 Ibat

switch flag,
    %%%%%%%%%%%%%%%%%%%%%%%%%%
    % Initialization %
    %%%%%%%%%%%%%%%%%%%%%%%%%%
    case 0
        [sys,x0,str,ts] = mdlInitializeSizes;

        %%%%%%%%%%%%%
        % Output %
        %%%%%%%%%%%%%
    case 3
        sys = mdlOutputs(t,x,u);

        %%%%%%%%%%%%%%%%%%%%%%%%%%
        % Update and Terminate %
        %%%%%%%%%%%%%%%%%%%%%%%%%%
    case {1,2,4,9}
        sys = []; % do nothing

    otherwise
        error(['unhandled flag = ',num2str(flag)]);
end

% end limintm

%
%=====
% mdlInitializeSizes
% Return the sizes, initial conditions, and sample times for the S-function.
%=====
%
function [sys,x0,str,ts] = mdlInitializeSizes

sizes = simsizes;
sizes.NumContStates = 0;
sizes.NumDiscStates = 0;
sizes.NumOutputs = 1;
sizes.NumInputs = 6;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;

sys = simsizes(sizes);
str = [];
x0 = [];
ts = [0 0]; % sample time : [period, offset]

% end mdlInitializeSizes

%=====
% mdlOutputs
% Return the output vector for the S-function
%=====
%
function sys = mdlOutputs(t,x,u)

```

```

R = 0.14;

%% Tension dans la batterie

if u(1) < u(3)
    x = (4.1-1.6×u(1)+3.6×(u(1).^2)-3.5×(u(1).^3))-(R×(u(6)-0.7))+(-
0.16+0.0093×u(2)-0.00018×(u(2).^2)+(3.8e-6)×(u(2).^3))
else
    x = (4.1-1.6×(u(1)+u(4)+u(5)))+3.6×((u(1)+u(4)+u(5)).^2)-
3.5×((u(1)+u(4)+u(5)).^3)
end

sys = x;

% end mdlOutputs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

# Glossaire

---

## *A*

<b>AMS</b>	Analog and Mixed Signal
<b>ASIC</b>	Application Specific Integrated Circuit

## *C*

<b>CAO</b>	Conception Assistée par Ordinateur
<b>CSSL</b>	Continuous System Simulation Languages

## *D*

<b>Dof</b>	Degree of freedom
------------	-------------------

## *E*

<b>EDA</b>	Electronic Design Automation
------------	------------------------------

## *F*

<b>FDM</b>	Finite Difference Method
<b>FDTD</b>	Finite Difference Time Domain
<b>FEA</b>	Finite Element Analyze
<b>FEM</b>	Finite Element Method
<b>FPGA</b>	Field Programmable Gate
<b>FSM</b>	Finite State Machine

## *H*

<b>HDL</b>	Hardware Description Language
------------	-------------------------------

## *I*

<b>IP</b>	Intellectual Property
-----------	-----------------------

## *M*

<b>MCM</b>	Multi Chip Module
<b>MEMS</b>	MicroElectroMechanical System
<b>MSMD</b>	Mixed Signal Mixed Domain simulation
<b>MNA</b>	Modified Nodal Analysis
<b>MNS</b>	Micro and Nano System (groupe TIMA)
<b>MST</b>	MicroSystem Technology
<b>mote</b>	Capteur miniature

## *N*

<b>Netlist</b>	Liste des interconnexions
----------------	---------------------------

***O***

<b>ODE</b>	Ordinary Differential Equations
<b>OSMOSE</b>	

***P***

<b>PDE</b>	Partial Differential Equation
<b>PWL</b>	PieceWise Linear

***R***

<b>RF</b>	Radio Frequency
<b>RTL</b>	Register Transfer Level

***S***

<b>SoC</b>	System on Chip
<b>SiP</b>	System in Package
<b>SPMS</b>	Self Powered MicroSystem
<b>SPW</b>	Signal Processing Worksystem

***U***

<b>ULP</b>	Ultra Low Power
------------	-----------------

***V***

<b>VHDL</b>	
<b>VLSI</b>	Very Large Scale Integration

---

## TITRE

Modélisation et simulation de microsystemes multi domaines à signaux mixtes :  
vers le prototypage virtuel d'un microsysteme autonome

### Résumé

Les travaux présentés dans cette thèse portent sur la modélisation et la simulation d'un microsysteme multidisciplinaire composé de dispositifs qui sont conçus dans des domaines différents: électromécanique, électronique analogique et numérique, et électrochimique.

L'objectif est de proposer des prototypes virtuels d'un microsysteme dit autonome. Ce microsysteme permet de générer à partir de vibrations mécaniques, une énergie électrique qui sert à alimenter tout le dispositif, et le rendre électriquement indépendant.

Une méthode de modélisation collaborative sous l'environnement MATLAB/Simulink est proposée afin d'étudier le comportement du microsysteme. Elle est réalisée à différents niveaux de détails en se basant sur deux approches de simulations. La première permet de vérifier le comportement du prototype virtuel dans un même environnement et de le valider par simulation globale. La deuxième approche consiste à simuler le système complet en faisant appel à l'interface de cosimulation avec le langage VHDL sous ModelSim.

Les résultats de simulation confrontés aux mesures effectuées sur le prototype physique ont montré la fiabilité de la méthode proposée.

**Mots clés :** Microsystemes, prototypage virtuel, modélisation multi domaine à signaux mixtes, système de récupération d'énergie, MATLAB/Simulink.

---

## TITLE

Modelling and simulation of the mixed signal and multi domain Microsystems :  
toward virtual prototyping of an autonomous microsystem

### Abstract

The work carried in this thesis deals with the modeling and simulation of multi-disciplinary microsystems composed of devices coming from different domains: electromechanical, digital and analog electronics and electrochemical. Our goal is to propose virtual prototypes for autonomous microsystems. Such microsystems are able to generate, from mechanical vibrations, the needed electric energy to power the overall system, making it electrically independent.

A method for collaborative modelling under the MATLAB/Simulink environment is proposed allowing to study the complex behaviour of these microsystems. This is achieved at different levels of detail and relies on two simulation approaches. The first approach allows verifying the behavior of the virtual prototype within the same environment through global simulation. The second approach relies on a co-simulation interface with VHDL under ModelSim.

Compared to the results measured on the physical prototype, the obtained simulation results showed the reliability and effectiveness of the proposed method.

**Keywords :** Microsystems, virtuel prototyping, multi domain and mixed signal modelling, energy harvesting system, MATLAB/Simulink.

---

Laboratoire TIMA, 46 avenue Félix Viallet, 38031 Grenoble, France.

---

ISBN/ISBNE : 978-2-84813-109-2