



# **ALGORITHMES ET PROCESSEURS TEMPS RÉEL DE TRAITEMENT DE SIGNAUX NEURONAUX POUR UNE PLATEFORME OPTOGÉNÉTIQUE SANS FIL**

**Mémoire**

**GABRIEL GAGNON-TURCOTTE**

**Maîtrise en génie électrique**  
Maître ès sciences (M.Sc.)

Québec, Canada

© GABRIEL GAGNON-TURCOTTE, 2015



# Résumé

L'acquisition des signaux électriques provenant des neurones du cerveau permet aux neurobiologistes de mieux comprendre son fonctionnement. Dans le cadre de ce travail, de nouveaux algorithmes de compression des signaux neuronaux sont conçus et présentés. Ces nouveaux algorithmes sont incorporés dans trois nouveaux dispositifs optogénétiques sans fil miniature capable de stimuler optiquement l'activité neuronale et de transmettre sans fil les biopotentiels captés par plusieurs microélectrodes.

Deux de ces systèmes sont capables de compresser les signaux provenant de deux microélectrodes ainsi que de stimuler optiquement via deux diodes électroluminescentes (DEL) haute puissance. Afin de réduire la bande passante des transmetteurs sans fil utilisés, ces deux systèmes sont dotés d'un nouvel algorithme de détection des potentiels d'actions qui génère de meilleurs taux de détection que les algorithmes existants, tout en nécessitant moins de ressources matérielles et de temps de processeur.

Un troisième dispositif incorporant les algorithmes de détection et de compression fût conçu. Ce dispositif est le seul système optogénétique sans fil comportant 32 canaux de stimulation optique et 32 canaux d'enregistrement électrophysiologiques en parallèle. Il utilise une nouvelle technique de compression par ondelettes permettant d'augmenter significativement le nombre de canaux sous observation sans augmenter la consommation de l'émetteur-récepteur. Cette nouvelle méthode de compression se distingue des méthodes existantes en atteignant de meilleurs taux de compression tout en permettant de reconstruire les signaux compressés avec une meilleure qualité.

Au moment de la rédaction de ce mémoire, il s'agit des premiers dispositifs optogénétiques sans fil à offrir simultanément de la stimulation optique multicanal, de l'enregistrement électrophysiologique multicanal ainsi que de la détection/compression *in situ* des potentiels d'actions. Grâce à leur design novateur et aux innovations apportées par les nouveaux algorithmes de traitement des signaux, les systèmes conçus sont plus légers et plus compacts que les systèmes précédents, rendant ces dispositifs indispensables afin de mener des expériences sur le cerveau de petits animaux libres de leurs mouvements. Les trois systèmes ont été validés avec grand succès par des expériences *in vivo* sur des souris transgéniques au Centre de Recherche de l'Institut Universitaire en Santé Mentale de Québec (CRIUSMQ).



# Abstract

The electrical signals acquisition from the brain's neurons allows neuroscientists to better understand its functioning. In this work, new neural signals compression algorithms are designed and presented. These new algorithms are incorporated into three new miniature optogenetic wireless devices. These devices are capable to optically stimulate neural activity and to wirelessly transmit the biopotentials captured by several microelectrodes.

Two of these systems are able to compress the signals from two microelectrodes and to stimulate optically via two high-power LED. Both systems feature a new spike detection algorithm to reduce the bandwidth used by the wireless transceiver. This new spike detection algorithm differs from existing algorithms by achieving better detection rate while using less material resources and processing time.

A third device incorporating the detection and compression algorithms was designed. This device is the only optogenetic wireless system including 32 optical stimulation channels and 32 electrophysiological recording channels in parallel. This new system has the ability to compress the neural signals using a new wavelet compression technique that significantly increase the number of channels under observation without increasing the consumption of the wireless transceiver. In particular, this new compression technique differs from the existing wavelet based compression methods by achieving better compression ratio while allowing to reconstruct the compressed signals with better quality.

At the time of writing this thesis, these are the first three devices that offer simultaneous multichannel optical stimulation, multichannel electrophysiological signals recording and on-the-fly spike detection. The resulting systems are more compact and lightweight than previous systems, making these devices essentials to conduct long term experiments on the brains of small freely moving animals. The three systems were validated within *in vivo* experiments using transgenic mice at the Centre de Recherche de l'Institut Universitaire en Santé Mentale de Québec (CRIUSMQ).



# Table des matières

<b>Résumé</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Table des matières</b>	<b>vii</b>
<b>Liste des tableaux</b>	<b>xi</b>
<b>Liste des figures</b>	<b>xv</b>
<b>Acronymes</b>	<b>xix</b>
<b>Remerciements</b>	<b>xxi</b>
<b>Introduction</b>	<b>1</b>
1.1 Contexte de réalisation de la recherche . . . . .	2
1.1.1 Étude sur le cerveau . . . . .	2
Signaux neuronaux . . . . .	3
Enregistrement électrophysiologique . . . . .	4
Systèmes d’acquisition de données neuronales traditionnels . . . . .	4
1.1.2 Recherche optogénétique . . . . .	6
1.2 Problématique . . . . .	7
1.2.1 Systèmes ASIC . . . . .	8
1.2.2 Systèmes Commercial Off-The-Shelf (COTS) . . . . .	10
1.3 Objectifs du projet . . . . .	11
1.4 Défis . . . . .	12
1.4.1 Compression des signaux électrophysiologiques . . . . .	12
1.4.2 Intégration dans une plateforme optogénétique . . . . .	12
1.5 Contributions . . . . .	13
1.6 Structure du mémoire . . . . .	15
<b>2 Revue de littérature</b>	<b>17</b>
2.1 Compression en transmettant seulement les AP . . . . .	17
2.1.1 Algorithmes de détection . . . . .	17
2.1.2 Algorithmes de seuillage . . . . .	18
2.2 Compression par Ondelettes . . . . .	20
2.2.1 Présentation des ondelettes, aspects mathématiques . . . . .	20
2.2.2 Calcul de la transformée en ondelettes par la méthode de <i>Mallat</i> . . . . .	21

2.2.3	Calcul de la transformée en ondelette par la méthode de <i>Lifting</i> . . .	23
2.2.4	Compression par ondelettes dans la littérature . . . . .	24
2.3	Encodage Delta . . . . .	25
2.4	Quantification et fréquence d'échantillonnage . . . . .	26
2.5	Compression de type <i>Compressed Sensing</i> (CS) . . . . .	26
2.5.1	Présentation de l'algorithme CS, aspects mathématiques . . . . .	26
2.5.2	Compression par CS dans la littérature . . . . .	27
2.6	Méthodes basées sur la quantification vectorielle (QV) . . . . .	28
2.6.1	Présentation de l'algorithme QV, aspects mathématiques . . . . .	28
2.6.2	Algorithme de compression par QV . . . . .	29
2.7	Encodage des coefficients (ondelettes ou autre) . . . . .	29
2.7.1	Encodage SPITH . . . . .	29
2.7.2	Encodage de Huffman . . . . .	29
<b>3</b>	<b>Travaux préliminaires de conception et de simulation</b>	<b>31</b>
3.1	Méthodologie . . . . .	31
3.1.1	Tests reliés au SNDR . . . . .	32
3.1.2	Tests de <i>clustering</i> . . . . .	32
	Comparaison du nombre de <i>clusters</i> . . . . .	32
	Association au bon <i>cluster</i> . . . . .	32
3.2	Compression par transformée en ondelettes . . . . .	33
3.2.1	<i>Haar</i> . . . . .	34
3.2.2	<i>Symmlet-2</i> . . . . .	34
3.3	Encodage <i>Delta</i> . . . . .	35
3.4	Diminuer le nombre de bits de quantification (DNBQ) . . . . .	35
3.5	Compression de type <i>Compressed Sensing</i> . . . . .	35
3.5.1	Parcimonie dans le domaine des ondelettes (CSWD) . . . . .	36
3.5.2	Base parcimonieuse optimale (CSOP) . . . . .	36
3.6	Synthèse des résultats . . . . .	36
3.6.1	SNDR . . . . .	37
3.6.2	Classification des AP et déviation moyenne . . . . .	37
3.6.3	Variation du nombre de <i>clusters</i> détectés . . . . .	40
3.6.4	Fonction de coûts . . . . .	42
3.6.5	Conclusion des simulations . . . . .	43
<b>4</b>	<b>Système optogénétique sans fil avec détection en temps réel des AP</b>	<b>45</b>
4.1	Présentation du projet . . . . .	45
4.2	Conception du <i>headstage</i> . . . . .	47
4.2.1	Circuit <i>Analog Front End</i> (AFE) . . . . .	48
4.2.2	Circuit de stimulation optique . . . . .	50
4.2.3	Circuit de transmission sans fil . . . . .	51
4.2.4	Circuit de gestion de l'énergie (PMU) . . . . .	51
4.2.5	<i>Microcontroller Unit</i> (MCU) . . . . .	51
4.2.6	PCB . . . . .	53
4.3	Compression par détection des AP . . . . .	54
4.3.1	Sélection de l'algorithme de détection optimal . . . . .	54
	Signaux neuronaux synthétiques . . . . .	54
	Résultats de détection . . . . .	55



	Temps d'exécution des algorithmes . . . . .	55
4.3.2	Algorithme de seuillage adaptatif . . . . .	56
	Aspects mathématiques . . . . .	56
	Implémentation optimisée pour MCU . . . . .	59
	Innovations de l'algorithme de seuillage adaptatif . . . . .	60
4.3.3	Reconstruction temporelle des signaux neuronaux . . . . .	61
4.4	Caractéristiques du <i>headstage</i> . . . . .	61
4.4.1	Caractéristiques physiques . . . . .	61
4.4.2	Autonomie et consommation électrique . . . . .	61
4.4.3	Portée du transmetteur . . . . .	63
4.5	Validation <i>in vitro</i> . . . . .	63
4.5.1	Résultats d'enregistrements . . . . .	64
4.5.2	Performances de détection des AP . . . . .	64
	Paramètres d'implémentation . . . . .	64
	Résultats de détection . . . . .	66
	Temps d'exécution du détecteur . . . . .	67
4.6	Validation <i>in vivo</i> . . . . .	68
4.6.1	Préparation de l'animal . . . . .	68
4.6.2	Résultats . . . . .	69
4.7	Système amélioré pour commercialisation . . . . .	71
4.7.1	Améliorations . . . . .	71
4.7.2	Validation <i>in vivo</i> . . . . .	73
<b>5</b>	<b>Système optogénétique sans fil à 32 canaux avec compression et détection en temps réel des AP</b> . . . . .	<b>77</b>
5.1	Présentation du projet . . . . .	77
5.2	Conception du <i>headstage</i> . . . . .	78
5.2.1	Circuit d'acquisition neuronal . . . . .	79
5.2.2	Circuit de stimulation optique . . . . .	80
5.2.3	Unité de gestion de l'énergie . . . . .	80
5.2.4	Processeur de traitement de signal . . . . .	80
5.2.5	Système de transmission dans fil . . . . .	81
5.2.6	PCB . . . . .	81
5.3	Algorithme modifié de compression basé sur les ondelettes . . . . .	82
5.3.1	Implémentation séquentielle de l'ondelette <i>Symmlet-2</i> par <i>lifting</i> . . . . .	83
5.3.2	Calcul du seuil des coefficients . . . . .	85
5.3.3	Compression des données . . . . .	85
5.3.4	Innovations . . . . .	87
5.4	Algorithme de détection des AP . . . . .	88
5.4.1	Aspects mathématiques . . . . .	88
5.4.2	Implémentation optimisée pour FPGA . . . . .	89
5.5	Seuillage adaptatif . . . . .	89
5.5.1	Aspects mathématiques . . . . .	89
5.5.2	Implémentation optimisée pour FPGA . . . . .	90
<b>6</b>	<b>Résultats expérimentaux</b> . . . . .	<b>93</b>
6.1	Caractéristiques du système . . . . .	93
6.2	Résultats expérimentaux <i>in vitro</i> . . . . .	94

6.2.1	Résultats de détection des AP . . . . .	94
	Paramètres d'implémentation . . . . .	95
	Résultats de détection . . . . .	95
6.2.2	Résultats de compression du <i>headstage</i> avec différents SNR . . . . .	96
	Méthodologie . . . . .	96
	Résultats . . . . .	97
6.2.3	Résultats comparatifs de compression pour différents CR . . . . .	98
	Méthodologie . . . . .	98
	Résultats . . . . .	99
6.2.4	Détection/compression sur 32 canaux simultanément . . . . .	101
6.3	Résultats expérimentaux <i>in vivo</i> . . . . .	101
6.3.1	Préparation de l'animal . . . . .	102
6.3.2	Paramètres du <i>headstage</i> . . . . .	102
6.3.3	Enregistrements à partir de toutes les microélectrodes . . . . .	103
6.3.4	Résultats de détection et de compression . . . . .	104
6.3.5	Résultats de stimulation . . . . .	105
<b>7</b>	<b>Analyse comparative</b>	<b>107</b>
	<b>Conclusion</b>	<b>113</b>
<b>A</b>	<b>Résultats détaillés des simulations</b>	<b>117</b>
A.1	Ondelettes . . . . .	117
	A.1.1 <i>Haar</i> . . . . .	117
	A.1.2 <i>Symmlet-2</i> . . . . .	118
A.2	Encodage <i>Delta</i> . . . . .	119
A.3	Diminuer le nombre de bits de quantification (DNBQ) . . . . .	120
A.4	Compression de type <i>Compressed Sensing</i> . . . . .	121
	A.4.1 Parcimonie dans le domaine des ondelettes (CSWD) . . . . .	121
	A.4.2 Base parcimonieuse optimale (CSOP) . . . . .	122
<b>B</b>	<b>Résultats détaillés obtenus avec le <i>headstage</i> à 32 canaux</b>	<b>125</b>
B.1	Tests reliés au PSNR . . . . .	125
B.2	Détection/compression sur 32 canaux simultanément . . . . .	127
<b>C</b>	<b>Cartes de développement et station de base</b>	<b>129</b>
C.1	Cartes de développement . . . . .	129
C.2	Station de base . . . . .	130
<b>D</b>	<b>Interfaces usager</b>	<b>131</b>
D.1	Interface systèmes à 2 canaux . . . . .	131
D.2	Interface système à 32 canaux . . . . .	131
	<b>Bibliographie</b>	<b>133</b>

# Liste des tableaux

1.1	Comparaison de systèmes d'enregistrement électrophysiologiques basés sur technologie ASIC. . . . .	9
1.2	Comparaison de systèmes d'enregistrement électrophysiologiques basés sur des composants COTS. . . . .	10
3.1	Score calculé pour chacun des algorithmes de compression en fonction d'un CR de 2 à 8. . . . .	43
3.2	Fonction de coût calculée pour chacun des algorithmes de compression. . . . .	43
4.1	Requis de conception du <i>headstage</i> optogénétique à deux canaux exigés par le partenaire industriel <i>Doric Lenses</i> . . . . .	46
4.2	Résumé des caractéristiques mesurées de l'AFE du <i>headstage</i> optogénétique à deux canaux. . . . .	49
4.3	Résumé des caractéristiques mesurées du circuit de stimulation optique. . . . .	50
4.4	Taux de détection du <i>headstage</i> à deux canaux avec les algorithmes à valeur absolue, NEO et MTEO selon différents SNR : 20dB, 15dB et 10dB. . . . .	55
4.5	Temps d'exécution du <i>headstage</i> à deux canaux pour traiter 32 échantillons par canal avec les algorithmes à valeur absolue, NEO et MTEO. . . . .	56
4.6	Gains en consommation du <i>headstage</i> à deux canaux engendré par la détection des AP en fonction du <i>firing rate</i> . . . . .	63
4.7	Résultats comparatifs de détection entre la méthode de seuillage adaptative proposée exécutée par le <i>headstage</i> à deux canaux versus d'autres méthodes exécutés sur les signaux bruts importés dans <i>Matlab</i> et possédant un <i>firing rate</i> variant de 25, 75 à 125 AP/s. . . . .	66
4.8	Variations des résultats de détection présentés au tableau 4.7 entre la méthode de seuillage adaptative proposée et les autres méthodes. . . . .	67
4.9	Comparaison du temps d'exécution du MCU du <i>headstage</i> à deux canaux avec la méthode de détection proposée versus d'autres méthodes pour une fréquence de 8 MHz. . . . .	68
4.10	Résultats de détection du <i>headstage</i> à deux canaux commercial pour 8 séances d'enregistrement électrophysiologique <i>in vivo</i> , comparés aux résultats obtenus avec les méthodes de seuillage RMS et MAD exécutées sur les signaux bruts importés dans <i>Matlab</i> . . . . .	75
6.1	Caractéristiques du <i>headstage</i> optogénétique à 32 canaux. . . . .	93
6.2	Comparaison des performances de détection entre la méthode proposée exécutée par le <i>headstage</i> à 32 canaux versus d'autres méthodes exécutées sur les signaux bruts importés dans <i>Matlab</i> et possédant un <i>firing rate</i> variant de 25 à 125 AP/s. . . . .	96

6.3	Paramètres de compression utilisés afin de comparer les performances des algorithmes de compression pour différent CR. . . . .	99
7.1	Comparaison des <i>headstages</i> présentés dans ce mémoire avec d'autres systèmes dotés de compression des signaux neuronaux. Les résultats de la fonction de coût sont présentés dans la dernière colonne du tableau afin de pouvoir comparer directement les performances de chacun des systèmes. . . . .	110
A.1	Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés avec ondelette de type <i>Haar</i> . . . . .	117
A.2	Calcul du nombre de <i>clusters</i> avant compression ( <i>clusters</i> originaux), après compression/décompression ( <i>clusters</i> compressés), avec AP compressés/décompressés joint aux AP originaux ( <i>clusters</i> mélangés) pour la compression avec ondelette de type <i>Haar</i> . . . . .	117
A.3	Déviation moyenne des AP compressés/décompressés par rapport au centre des <i>clusters</i> originaux, pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l'algorithme de compression par ondelette de type <i>Haar</i> . . . . .	118
A.4	Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés avec ondelette de type <i>Symmlet-2</i> . . . . .	118
A.5	Calcul du nombre de <i>clusters</i> avant compression ( <i>clusters</i> originaux), après compression/décompression ( <i>clusters</i> compressés), avec AP compressés/décompressés joint aux AP originaux ( <i>clusters</i> mélangés) pour la compression avec ondelette de type <i>Symmlet-2</i> . . . . .	118
A.6	Déviation moyenne des AP compressés/décompressés par rapport au centre des <i>clusters</i> originaux, pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l'algorithme de compression par ondelette de type <i>Symmlet-2</i> . . . . .	119
A.7	Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés avec l'algorithme <i>Delta</i> . . . . .	119
A.8	Calcul du nombre de <i>clusters</i> avant compression ( <i>clusters</i> originaux), après compression/décompression ( <i>clusters</i> compressés), avec AP compressés/décompressés joint aux AP originaux ( <i>clusters</i> mélangés) pour l'algorithme de compression <i>Delta</i> . . . . .	120
A.9	Déviation moyenne des AP compressés/décompressés par rapport au centre des <i>clusters</i> originaux, pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l'algorithme de compression <i>Delta</i> . . . . .	120
A.10	Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés en diminuant le nombre de bits de quantification. . . . .	120
A.11	Calcul du nombre de <i>clusters</i> avant compression ( <i>clusters</i> originaux), après compression/décompression ( <i>clusters</i> compressés), avec AP compressés/décompressés joint aux AP originaux ( <i>clusters</i> mélangés) pour la compression obtenue en diminuant le nombre de bits de quantification. . . . .	121

A.12	Déviati on moyenne des AP compressés/décompressés par rapport au centre des <i>clusters</i> originaux, pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue par compression en diminuant le nombre de bits de quantification. . . . .	121
A.13	Calcul du SNDR moyen, minimum, maximum et de la déviati on moyenne des signaux compressés/décompressés avec l'algorithme CS et parcimonie dans le domaine des ondelettes ( <i>Daubechie-8</i> ). . . . .	121
A.14	Calcul du nombre de <i>clusters</i> avant compression ( <i>clusters</i> originaux), après compression/décompression ( <i>clusters</i> compressés), avec AP compressés/décompressés joint aux AP originaux ( <i>clusters</i> mélangés) pour la compression avec l'algorithme CS et parcimonie dans le domaine des ondelettes ( <i>Daubechie-8</i> ). . . . .	122
A.15	Déviati on moyenne des AP compressés/décompressés par rapport au centre des <i>clusters</i> originaux, pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l'algorithme de compression CS et parcimonie dans le domaine des ondelettes ( <i>Daubechie-8</i> ). . . . .	122
A.16	Calcul du SNDR moyen, minimum, maximum et de la déviati on moyenne des signaux compressés/décompressés avec l'algorithme CS et matrice de parcimonie optimale. . . . .	123
A.17	Calcul du nombre de <i>clusters</i> avant compression ( <i>clusters</i> originaux), après compression/décompression ( <i>clusters</i> compressés), avec AP compressés/décompressés joint aux AP originaux ( <i>clusters</i> mélangés) pour la compression avec l'algorithme CS et matrice de parcimonie optimale. . . . .	123
A.18	Déviati on moyenne des AP compressés/décompressés par rapport au centre des <i>clusters</i> originaux, pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d'association au bon <i>cluster</i> pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l'algorithme de compression CS et matrice de parcimonie optimale. . . . .	123
B.1	Résultats du SNDR moyen, minimum, maximum, ainsi que l'écart-type pour des signaux ayant une SNR variant de 25 dB à 0 dB. Les résultats ont été obtenus à partir des AP détectés, compressés et transmis par le <i>headstage</i> à 32 canaux. . . . .	125
B.2	PSNR moyen, minimum, maximum ainsi que l'écart-type relatif au PSNR moyen pour l'algorithme de compression par ondelettes avec maximisation dynamique de la quantification. Ces résultats ont été obtenus avec des signaux neuronaux synthétiques acquisitionnés avec le <i>headstage</i> à 32 canaux et importés dans <i>Matlab</i> . . . . .	126
B.3	PSNR moyen, minimum, maximum ainsi que l'écart-type relatif au PSNR moyen pour l'algorithme de compression par ondelettes sans maximisation dynamique de la quantification. Ces résultats ont été obtenus avec des signaux neuronaux synthétiques acquisitionnés avec le <i>headstage</i> à 32 canaux et importés dans <i>Matlab</i> . . . . .	126
B.4	PSNR moyen, minimum, maximum ainsi que l'écart-type relatif au PSNR moyen pour l'algorithme de compression <i>Delta</i> . Ces résultats ont été obtenus avec des signaux neuronaux synthétiques acquisitionnés avec le <i>headstage</i> à 32 canaux et importés dans <i>Matlab</i> . . . . .	126

B.5 PSNR moyen, minimum, maximum ainsi que l'écart-type relatif au PSNR moyen pour l'algorithme de compression CS avec matrice de parcimonie optimale. Ces résultats ont été obtenus avec des signaux neuronaux synthétiques acquisitionnés avec le <i>headstage</i> à 32 canaux et importés dans <i>Matlab</i> . . . . .	126
---	-----

# Liste des figures

1.1	Forme d'onde d'un AP intracellulaire comportant différentes phases. . . . .	3
1.2	Système d'enregistrement de signaux électrophysiologiques câblé. . . . .	5
1.3	Concept et schéma blocs d'une chaîne d'acquisition sans fil de signaux électrophysiologiques. . . . .	5
1.4	a) Stimulation électrique versus stimulation/inhibition optique des neurones (image tirée de [1]), b) souris transgénique en pleine expérience optogénétique (image tirée de [2]). . . . .	6
1.5	Signal reconstruit avec les AP détectés (en rouge) versus le signal original (en bleu). . . . .	8
2.1	Représentation schématique d'une transformée en ondelettes de 4 niveaux de profondeur par la méthode de <i>Mallat</i> . . . . .	22
2.2	Représentation schématique d'une transformée en ondelettes inverse de 4 niveaux de profondeur par la méthode de <i>Mallat</i> . . . . .	23
2.3	Représentation schématique d'une transformée en ondelettes de type <i>Haar</i> de 4 niveaux de profondeur par la méthode de <i>lifting</i> . . . . .	24
3.1	a) <i>Clustering</i> automatique des 3237 AP originaux traités avec une PCA suivie de l'algorithme <i>mean shift clustering</i> , b) affichage simultané des 3237 AP détectés, où chaque couleur correspond à une association AP à neurone. . . . .	33
3.2	Formes des fonctions $\psi$ pour les ondelettes de type <i>Haar</i> et <i>Daubechie-2</i> . . . . .	34
3.3	SNDR moyen obtenu avec tous les algorithmes testés pour un CR de 2 à 8. . . . .	38
3.4	SNDR minimum obtenu avec tous les algorithmes testés pour un CR de 2 à 8. . . . .	38
3.5	SNDR maximum obtenu avec tous les algorithmes testés pour un CR de 2 à 8. . . . .	38
3.6	Déviations moyennes du SNDR obtenu avec tous les algorithmes testés pour un CR de 2 à 8. . . . .	39
3.7	Pourcentage des AP compressés dans le bon <i>cluster</i> avec PCA effectuée sur les AP compressés joint aux AP originaux. . . . .	40
3.8	Pourcentage des AP compressés dans le bon <i>cluster</i> avec PCA effectuée en ajoutant les AP compressés un à la fois aux AP originaux. . . . .	40
3.9	Nombre de <i>clusters</i> trouvés après compression. . . . .	41
3.10	Nombre de <i>clusters</i> trouvés avec les AP compressés joints aux AP originaux. . . . .	41
4.1	Concept de la plateforme optogénétique sans fil à deux canaux. . . . .	46
4.2	Diagramme en bloc du <i>headstage</i> optogénétique à deux canaux. . . . .	47
4.3	Circuit de l'AFE. . . . .	48
4.4	Réponse en fréquence de l'AFE mesurée avec un analyseur de spectre Agilent 35670A. . . . .	49
4.5	a) Circuit de stimulation optique, b) circuit PMU. . . . .	50
4.6	Diagramme fonctionnel du <i>firmware</i> de contrôle utilisant un RTOS. . . . .	52

4.7	Fenêtre glissante de détection et de transmission des AP. . . . .	53
4.8	a) Design du PCB flexible rigide, b) vue 3D du PCB flexible rigide replié et connecté à un module implantable. . . . .	54
4.9	Fonction de densité de la loi demi-normale présentée avec des multiples de son écart-type ( $\sigma$ ). . . . .	57
4.10	Boucle de contrôle permettant de calculer le seuil de détection basé sur un multiple de l'écart-type estimé ( $\sigma_a$ ) du signal. . . . .	57
4.11	Diagramme fonctionnel de l'algorithme permettant de calculer le seuil de détection. . . . .	59
4.12	a) Vue non repliée du PCB flexible rigide, b) PCB flexible rigide avec la batterie sur le côté et le module implantable avec une DEL installée, c) vue repliée sur le côté, d) vue repliée sur le dessus. . . . .	62
4.13	Consommation électrique des sous-systèmes du <i>headstage</i> . a) Consommation avec transmission des signaux bruts, b) Consommation avec détection et transmission des AP avec les gains de consommation pour un <i>firing rate</i> de 50 AP/s. . . . .	62
4.14	Pertes de paquets mesurés en fonction de la distance entre l'émetteur ( <i>headstage</i> ) et le récepteur (station de base). . . . .	63
4.15	Montage de test <i>in vitro</i> du <i>headstage</i> optogénétique à deux canaux. . . . .	64
4.16	Signal neuronal constitué d'AP d'amplitude <i>peak-to-peak</i> <150 $\mu$ V capté à la sortie de l'AFE en même temps que le patron de stimulation aux bornes de la DEL, acquisitionnés par un oscilloscope MSO-X 2024A d'Agilent. . . . .	65
4.17	Signaux neuronaux acquisitionnés en utilisant l'ensemble de la chaîne d'acquisition de données, du <i>headstage</i> à la station de base. a) Patron de stimulation et signal neuronal captés par le <i>headstage</i> sur le même graphique, b) grossissement sur des AP de faible amplitude. . . . .	65
4.18	a) Variation automatique du seuil pour un SNR variant de 2dB, 20dB, 15dB, 10dB à 5dB avec un <i>firing rate</i> constant de 100 AP/s, b) signal reconstruit (en rouge) à partir des AP détectés dans un signal (en noir) ayant un SNR de 20 dB. . . . .	67
4.19	a) <i>Headstage</i> avec le module implantable, b) montage expérimental, c) <i>headstage</i> installé sur la souris en fonctionnement, d) <i>headstage</i> installé sur la souris avec la DEL en fonctionnement. . . . .	69
4.20	a) Résultats expérimentaux montrant un enregistrement extracellulaire (en noir) en réponse à une activation lumineuse locale (courant dans de DEL en rouge), les signaux montrés proviennent du <i>headstage</i> , b) exemple d'enregistrement <i>in vivo</i> sans réponse électrophysiologique, c) exemple de réponse électrophysiologique en fonction de trois temps d'activation lumineux différents (5, 10, 20 ms). . . . .	70
4.21	Résultats expérimentaux montrant la réponse extracellulaire d'AP à trois séquences de stimulation (1 à 5 pulses) et à des temps d'exposition lumineux différents (5, 10, 20 ms), b) grossissement sur l'enregistrement extracellulaire face à 5 pulses lumineux de 5 ms de long chacun, c) grossissement sur l'enregistrement extracellulaire face à 5 pulses lumineux de 20 ms de long chacun, montrant des AP <i>failures</i> . . . . .	71
4.22	Vue de chaque côté et non repliée du PCB flexible rigide optimisé en taille et en poids (1.1 g) avec chacune des parties électroniques clairement identifiées. . . . .	72
4.23	a) Vue 3D du PCB flexible rigide replié dans sa position finale afin de laisser un espace pour la batterie au centre, b) <i>headstage</i> replié avec la batterie installée dans une enceinte protectrice, c) système complet avec le module implantable installé. . . . .	72
4.24	a) Souris transgénique anesthésiée avec le module implantable (installé de façon permanente sur le crâne de l'animal) connecté au <i>headstage</i> , b) souris transgénique éveillée en pleine expérience d'enregistrement électrophysiologique avec le <i>headstage</i> . . . . .	73



4.25	a) Seuil calculé par le <i>headstage</i> (en rouge) superposé au signal brut (en noir) lors d'une expérience d'enregistrement électrophysiologique d'une durée de 50 secondes, b) AP détectés par le <i>headstage</i> , classés puis superposés. . . . .	74
4.26	a) Seuil calculé par le <i>headstage</i> (en rouge), seuil calculé par la méthode RMS (en vert) et MAD (en jaune) superposé au signal brut (en noir), b) AP détectés par le <i>headstage</i> , c) AP détectés par seuillage RMS, d) AP détectés par seuillage MAD . . . . .	75
5.1	Concept et diagramme en bloc du <i>headstage</i> optogénétique à 32 canaux. . . . .	78
5.2	a) Design du PCB flexible rigide, b) vue 3D du PCB flexible rigide déplié, c) vue 3D du PCB flexible rigide replié et attaché à un module implantable montrant une DEL couplée à une fibre optique et 9 microélectrodes. . . . .	82
5.3	Représentation schématique d'une transformée en ondelettes de type <i>Symmlet-2</i> de 4 niveaux de profondeur par la méthode de <i>lifting</i> . . . . .	83
5.4	Diagramme système de l'implantation séquentielle de la transformée en ondelettes de type <i>Symmlet-2</i> par <i>lifting</i> . . . . .	84
5.5	Coefficients obtenus après une transformée en ondelettes <i>Symmlet-2</i> de niveau 4 et seuillés afin de conserver 20 coefficients sur 48. . . . .	86
5.6	Algorithme de détection et de stockage en mémoire des AP. . . . .	88
5.7	Boucle de rétroaction permettant de calculer le seuil de détection pour chacun des 32 canaux en parallèle. . . . .	90
5.8	Algorithme de calcul du seuil de détection. . . . .	90
6.1	a) Vue recto verso du PCB flexible rigide non replié, b) vue du <i>headstage</i> avec un module implantable installé, c) vue du dessus du <i>headstage</i> replié avec la batterie sur le côté, d) vue de devant du <i>headstage</i> replié avec la batterie sur le côté. . . . .	94
6.2	Montage de test <i>in vitro</i> du <i>headstage</i> optogénétique à 32 canaux. . . . .	95
6.3	Variation du seuil pour un signal ayant un SNR passant de 15 dB, 0 dB, 5 dB à 10 dB et un <i>firing rate</i> constant de 75 AP/s. . . . .	97
6.4	a) Résultats du SNDR pour un SNR variant de 25 dB à 0 dB, b) meilleure reconstruction suivant le SNDR pour un signal ayant un SNR de 10 dB, c) pire reconstruction suivant le SNDR pour un signal ayant un SNR de 10 dB . . . . .	98
6.5	Comparaison des performances de compression du <i>headstage</i> versus celles obtenues avec l'algorithme <i>Delta</i> , l'algorithme CS avec base parcimonieuse optimale et l'algorithme du <i>headstage</i> sans maximisation la plage dynamique des coefficients, a) SNDR moyen et plage couverte par le SNDR, b) écart-type du SNDR relatif au SNDR moyen. . . . .	100
6.6	Module implantable incorporant une DEL couplée à une fibre optique insérée dans un tube de polyimide avec 8 microélectrodes d'enregistrement, une microélectrode de référence et un fil de mise à la masse. . . . .	101
6.7	a) Montage <i>in vivo</i> , b) <i>headstage</i> en fonctionnement pendant une expérience optogénétique. . . . .	102
6.8	AP détectés et regroupés en <i>clusters</i> à partir de l'enregistrement simultané des signaux provenant de 8 microélectrodes pendant une expérience <i>in vivo</i> d'une durée de 200 secondes. . . . .	103
6.9	Signal neuronal d'une durée de 60 secondes reconstruit avec les AP détectés et décompressés (en rouge) superposés au signal original (en noir). . . . .	104
6.10	Résultats expérimentaux <i>in vivo</i> montrant la réponse extracellulaire suite à une activation lumineuse pulsée de 10ms et ayant une période de 100 ms. . . . .	105

7.1	Histogramme des résultats de la fonction de coût pour tous les systèmes répertoriés dans le tableau 7.1. . . . .	111
B.1	AP détectés et compressés par le <i>headstage</i> à partir des 32 canaux simultanément (signal neuronal synthétique dupliqué sur tous les canaux), a) canaux de 1 à 8, b) canaux de 9 à 16, c) canaux de 17 à 24, d) canaux de 25 à 32. . . . .	128
C.1	Prototype de développement du <i>headstage</i> à deux canaux présenté au chapitre 4. . .	129
C.2	Prototype de développement du <i>headstage</i> à 32 canaux présenté au chapitre 5. . . . .	130
C.3	Station de base compatible avec tous les <i>headstages</i> . . . . .	130
D.1	Interface d'acquisition des <i>headstages</i> à 2 canaux. . . . .	131
D.2	Interface d'acquisition du <i>headstage</i> à 32 canaux. . . . .	132

## Acronymes

<b>ADC</b>	Analog to Digital Converter
<b>AFE</b>	Analog Front End
<b>AP</b>	Action Potential
<b>ASIC</b>	Application Specific Integrated Circuit
<b>BMI</b>	Brain Computer Interface
<b>CMRR</b>	Common Mode Rejection Ratio
<b>CR</b>	Compression Ratio
<b>CS</b>	Compressed Sensing
<b>COTS</b>	Commercial Off-The-Shelf
<b>CWT</b>	Continuous Wavelet Transform
<b>DEL</b>	Diode Électroluminescente
<b>DMA</b>	Direct Memory Access
<b>DSP</b>	Digital Signal Processor
<b>DWT</b>	Discrete Wavelet Transform
<b>ECG</b>	Electrocardiography
<b>EEG</b>	Electroencephalography
<b>EMG</b>	Electromyography
<b>EPE</b>	Energy Packing Efficiency
<b>FPGA</b>	Field-Programmable Gate Array
<b>K-SVD</b>	K-Means Singular Value Decomposition
<b>LFP</b>	Local Field Potential
<b>LSB</b>	Least Significant Bit
<b>MAD</b>	Median of the Absolute Deviation
<b>MCU</b>	Microcontroller Unit
<b>MSB</b>	Most Significant Bit
<b>MTEO</b>	Multiresolution TEO
<b>NEO</b>	Non Linear Energy Operator
<b>PCA</b>	Principal Component Analysis
<b>PCB</b>	Printed Circuit Board
<b>PMU</b>	Power Management Unit
<b>PSNR</b>	Peak Signal To Noise Ratio
<b>PSRR</b>	Power Supply Rejection Ratio
<b>PRBS</b>	Pseudo Random Binary Sequence
<b>PWM</b>	Pulse Width Modulation

**QV** Quantification Vectorielle  
**RFI** Radio Frequency Interference  
**RMS** Root Mean Square  
**RTOS** Real Time Operating System  
**SDWT** Stationary Discrete Wavelet Transform  
**SNEO** Smoothed Non Linear Energy Operator  
**SNR** Signal to Noise Ratio  
**SNDR** Signal to Noise-plus-Distortion ratio  
**SPI** Serial Peripheral Interface  
**SPIHT** Set Partitioning In Hierarchical Trees  
**STFT** Short-Time Fourier Transform  
**TEO** Teager's Operator  
**VLSI** Very-Large-Scale Integration

# Remerciements

J'aimerais tout d'abord remercier mon superviseur de recherche, Dr Benoit Gosselin de l'Université Laval pour son support indéfectible tout au long de ma maîtrise. Benoit Gosselin m'a donné la liberté et les outils nécessaires à la réalisation de mon projet de recherche. Il m'a épaulé, avec l'apport de nouvelles idées et défis, ce qui a contribué à repousser mes limites et à acquérir de nouvelles connaissances et expériences. Je suis privilégié de l'avoir eu comme directeur de recherche.

Je souhaiterais remercier mon co-superviseur, Dr Mohamad Sawan de l'École polytechnique de Montréal, de m'avoir accepté comme étudiant. Ce fut pour moi un honneur de l'avoir côtoyé.

J'aimerais aussi remercier l'entreprise *Doric Lenses*, Canada, pour leur support constructif à la réalisation de mon projet de recherche. En plus d'un appui financier, *Doric Lenses* a contribué en me fournissant de l'expertise technique et en m'ouvrant les portes du Centre de recherche de l'Institut Universitaire en Santé Mentale de Québec (CRIUSMQ) afin de mener des expériences *in vivo* avec les dispositifs développés. Parmi les employés chez *Doric Lenses*, j'aimerais particulièrement remercier Dr Yoan LeChasseur pour ses précieux conseils et son implication.

Durant mon séjour au laboratoire de recherche sur les microsystèmes biomédicaux (BioUL), j'ai eu la chance de travailler avec des collègues fantastiques. J'aimerais particulièrement remercier Reza Ameli, Alireza Avakh Kisomi et Charles-Olivier Dufresne Camaro, collègues et amis, qui ont apporté un apport constructif à la réalisation de mon projet de recherche.



# Introduction

La recherche fondamentale en neuroscience a pour but de percer les mystères entourant les maladies neurologiques et le fonctionnement du cerveau. Selon l'Institut de Recherche en Santé du Canada (IRSC), un Canadien sur cinq sera aux prises avec une maladie neurologique au cours de sa vie [3]. Avec le vieillissement de la population, tout porte à croire que ces statistiques vont continuer d'augmenter si de nouvelles percées ne sont pas effectuées en neuroscience. Pour cette raison, plusieurs organismes publics et privés sont motivés à injecter de plus en plus d'argent en recherche fondamentale sur le cerveau, qui jouit depuis peu d'un financement très important.

Parmi les projets ayant bénéficié d'un financement généreux, notons le *White House BRAIN Initiative* [4] lancé par l'administration Obama en 2013 aux États-Unis, dont le but est de soutenir le développement et l'application de technologies novatrices pouvant mener à une meilleure compréhension des fonctions cérébrales. Ce projet, dont le financement atteint les 4.5 milliards de dollars sur 12 ans, vise à financer le développement de nouveaux outils dédiés à l'analyse et à la recherche sur le cerveau afin de percer les mystères entourant certains troubles neurologiques, comme la maladie d'Alzheimer, le Parkinson ou bien la dépression, voire même le développement de nouvelles interfaces cerveau machines (BMI).

En Europe, depuis 2013, le *Human Brain Project* [5] est un projet financé par l'Union européenne à hauteur d'un milliard de dollars sur 10 ans visant à simuler le fonctionnement du cerveau humain sur superordinateur. Entre autres, ce projet vise à créer de nouveaux outils pour mieux comprendre le cerveau et ses mécanismes de base ainsi que de développer des modèles informatiques de maladies afin d'en améliorer le diagnostic et les traitements.

Au Canada en 2014, le *Fonds canadien de recherche sur le cerveau* [6] a accordé 51.4 millions de dollars, en fonds publics et privés, à des projets dédiés à la recherche sur le cerveau ainsi qu'au développement des outils de recherche nécessaires, et ce, afin que le Canada se joigne à l'effort mondial de compréhension du cerveau et des maladies neurologiques. À l'Université Laval seulement, le Centre de recherche de l'Institut Universitaire en Santé Mentale de Québec (CRIUSMQ) a bénéficié d'une des plus importantes subventions du Conseil de recherches en Sciences Naturelles et en Génie du Canada (CRSNG) de l'année 2015, soit de 510 000\$, afin de faire avancer la recherche sur les signalisations inter neurones.

En somme, partout en occident, la recherche sur le cerveau est en pleins essors et jouit d'un financement sans précédent. Malheureusement, les outils de recherche actuels ne sont pas adéquats et les besoins en recherche sont grandissants. C'est dans cette optique que ce mémoire présente de nouveaux algorithmes de détection et de compression de signaux neuro-naux qui permettent de récolter les signaux provenant d'une multitude de neurones à l'aide de dispositifs sans fil afin d'étudier les microcircuits du cerveau. Ces algorithmes ont été implémentés dans trois nouveaux outils de recherche innovateurs qui permettent d'effectuer des expériences optogénétiques inédites sur le cerveau de petits mammifères évoluant en toute liberté [7]. Les nouvelles expériences qui pourront être menées en neurosciences grâce à ces nouveaux systèmes sont multiples, et vont de la recherche fondamentale portant sur les communications entre neurones, à l'étude des maladies neurodégénératives en passant par la recherche de pointe en neurosciences comportementales.

Afin de cerner les enjeux influant sur la création de ce type d'outil et de clairement identifier le problème étudié et les motivations sous-jacentes à la rédaction de ce mémoire, le contexte de réalisation de la recherche, les problématiques, les défis et les contributions apportées par ce travail sont abordés en introduction.

## **1.1 Contexte de réalisation de la recherche**

### **1.1.1 Étude sur le cerveau**

Chez les mammifères, le cerveau est l'organe du système nerveux qui régule et qui contrôle l'organisme en plus de constituer le siège des fonctions cognitives menant à des processus de prise de décisions complexes. Le cerveau n'est pas un organe comme les autres, puisqu'il est constitué d'une structure extrêmement élaborée basée sur l'activité simultanée de milliards de neurones. En fait, le cerveau est la structure biologique la plus complexe connue [8] et dont l'architecture possède des caractéristiques qui sont communes à un grand nombre d'espèces [9]. Le cerveau est divisé en sous-sections qui sont constituées de groupes de neurones dédiés à la réalisation de tâches spécifiques, conscientes ou inconscientes, comme faire battre le cœur et veiller à la motricité des membres du corps. Afin de fonctionner correctement, ces groupes de neurones communiquent entre eux, individuellement ou en groupes, en s'envoyant des stimuli électriques dont l'information est encodée de manière temporelle et spatiale, et dont le fonctionnement n'est pas encore totalement compris [10]. Les mystères régissant les diverses fonctionnalités du cerveau sont nombreux et très complexes. Décoder/interpréter les messages envoyés par les neurones ou par des groupes de neurones permettrait aux neurobiologistes de mieux comprendre le fonctionnement du cerveau ainsi que de nombreux troubles neurologiques. Afin de mettre le lecteur en contexte, cette section fait une brève revue des types de signaux neuro-naux, du domaine et des méthodes d'enregistrement électrophysiologique en général ainsi que des systèmes d'acquisition de données



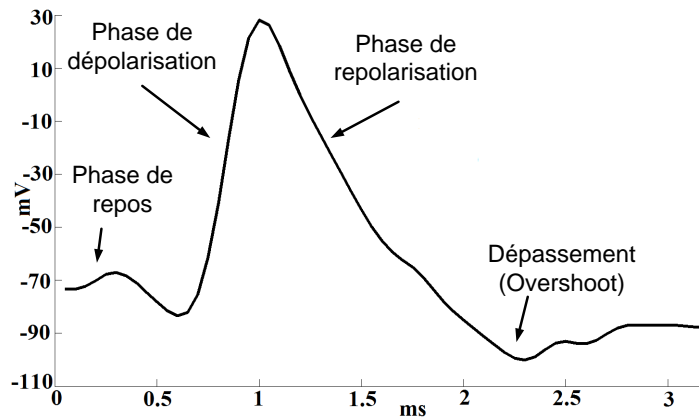


FIGURE 1.1 – Forme d'onde d'un AP intracellulaire comportant différentes phases.

traditionnels.

### Signaux neuronaux

Deux types de signaux principaux sont générés par les neurones et/ou par les groupes de neurones [11]. Premièrement, il y a les potentiels d'action (AP) qui sont générés par un neurone spécifique et qui sont issus de la dépolarisation de la membrane cellulaire. La dépolarisation de la membrane cellulaire s'effectue en suivant les phases suivantes :

- **Phase de repos** : Pompes à sodium ( $\text{Na}^+$ ) et à potassium ( $\text{K}^+$ ) fermés. Potentiel intracellulaire stable de  $\sim -70$  mV.
- **Phase de dépolarisation** : Pompes à sodium activées, pénétration des ions  $\text{Na}^+$  dans la cellule et ouverture des pompes à potassium après un court délai. Le débit entrant des ions  $\text{Na}^+$  est supérieur au débit de sortie des ions  $\text{K}^+$  ce qui génère une tension montante.
- **Phase de repolarisation** : Les pompes à sodium se referment et les pompes à potassium expulsent un maximum d'ions  $\text{K}^+$ , ce qui tend à repolariser la membrane cellulaire.
- **Dépassement** : Les pompes à potassium expulsent un surplus d'ions  $\text{K}^+$ , le potentiel de la membrane cellulaire se stabilise progressivement.

La figure 1.1 présente la forme typique d'un AP intracellulaire issu d'une banque de signaux neuronaux de l'Université Leicester [12]. On peut y voir les différentes phases de la dépolarisation cellulaire créant un AP ainsi que son amplitude et sa durée. L'AP généré par la dépolarisation de la membrane cellulaire se propage en suivant l'axone du neurone jusqu'à ce que le signal soit transmis via des neurotransmetteurs vers un neurone voisin. L'amplitude intracellulaire crête à crête d'un AP est de  $\sim 100$  mV alors que son amplitude extracellulaire crête à crête peut varier entre  $50 \mu\text{V}$  et  $500 \mu\text{V}$  [13], selon le type de cellule et la position de l'électrode. La durée d'un AP est de quelques millisecondes et ce type de

signal possède un contenu fréquentiel haute fréquence variant entre 300 Hz et 5 kHz [13]. En plus des AP, les neurones génèrent des signaux appelés potentiels de champ locaux (LFP), qui sont constitués de la somme des courants circulant dans les neurones environnants. Ce type de signal possède une amplitude de quelques millivolts et un contenu fréquentiel basse fréquence variant entre 0.5 Hz et 200 Hz [14]. Bien que les signaux LFP possèdent des caractéristiques intéressantes, les algorithmes de compression et les systèmes d'enregistrements présentés dans ce mémoire sont dédiés aux signaux neuronaux contenant les AP. À moins d'indications contraires, l'expression «signaux neuronaux» réfère implicitement aux signaux contenant les AP.

### **Enregistrement électrophysiologique**

L'enregistrement électrophysiologique est réalisé à partir de microélectrodes hautes impédances (100 k $\Omega$  - 10 M $\Omega$ ) implantées dans le cerveau et interfacées à l'aide d'un circuit d'amplification faible bruit [15], [16], [17], [18]. Parmi les types de matériaux à haute impédance constituant les microélectrodes, le verre, le Tungsten, l'acier inoxydable, des alliages à base de Platinium et d'Iridium ainsi que le Silicium sont utilisés. Ces électrodes peuvent être utilisées selon différentes topologies, variant selon le type d'expérience à mener, le dispositif d'enregistrement utilisé et le matériel constituant l'électrode (rigide ou flexible). Parmi les typologies existantes, notons les microélectrodes métalliques à pénétration simple [19], les microélectrodes de surface flexible [20], les microélectrodes de type *linear microelectrode array* [21] ainsi que les *microelectrodes array* [22], [23].

À partir des microélectrodes, deux types d'enregistrement des AP existent, soit l'enregistrement intracellulaire et extracellulaire. L'enregistrement intracellulaire implique l'insertion de la microélectrode à l'intérieur de la cellule (typiquement une micropipette de verre pour sa haute impédance) et permet d'enregistrer le voltage au repos de la membrane cellulaire ainsi que les AP provenant de l'axone et de l'intérieur de la cellule [11]. Ce type d'enregistrement est plus difficile à réaliser sur de petits neurones et endommage malheureusement la cellule. L'enregistrement extracellulaire est quant à lui plus approprié pour mesurer les AP externes à la cellule et est compatible avec une plus grande variété d'électrodes. De plus, ce type d'enregistrement est très approprié pour effectuer des expériences utilisant des sujets vivants, puisque la cellule en observation n'est pas endommagée par l'électrode.

### **Systèmes d'acquisition de données neuronales traditionnels**

L'acquisition extracellulaire des signaux neuronaux est effectuée à l'aide de microélectrodes interfacées à l'aide d'un circuit électrique, soit un circuit constitué de plusieurs étages de gains à faible bruit et de filtres analogiques [16], [17], [18]. Par la suite, les signaux conditionnés sont quantifiés sur un certain nombre de bits avec un convertisseur analogique à numérique (ADC) pour être traité dans le domaine numérique, dans le domaine des ordina-

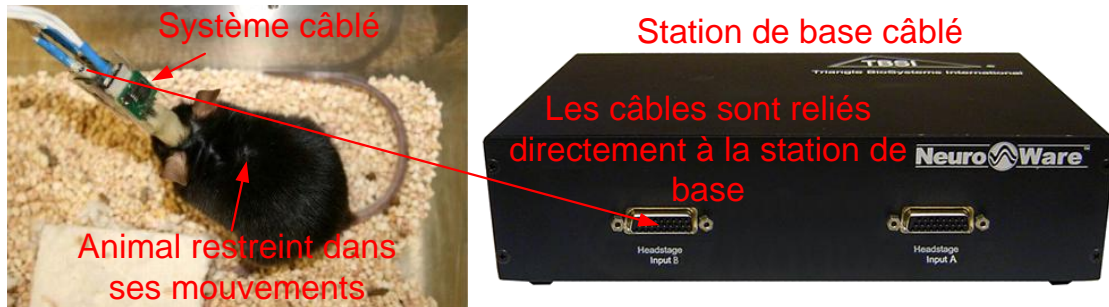


FIGURE 1.2 – Système d’enregistrement de signaux électrophysiologiques câblé.

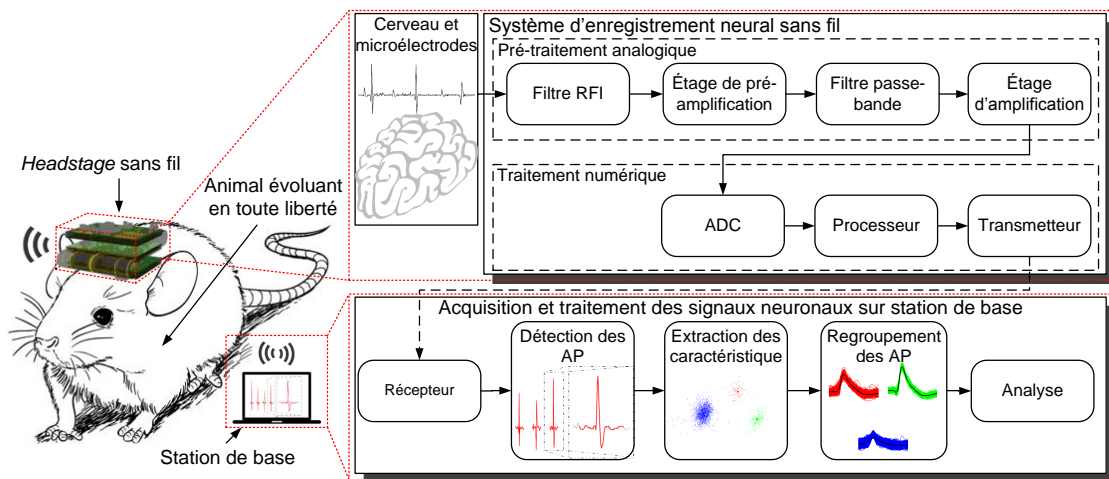


FIGURE 1.3 – Concept et schéma blocs d’une chaîne d’acquisition sans fil de signaux électrophysiologiques.

teurs par un processeur numérique, soit un microcontrôleur (MCU), un circuit de traitement de signal spécialisé (DSP) ou bien un FPGA. Les systèmes d’acquisitions traditionnels sont câblés, c’est-à-dire que les microélectrodes sont reliées du cerveau jusqu’à une station de base par des câbles, ce qui pose énormément de problèmes, de défis et de limitations lors des expérimentations. La figure 1.2 présente ce cas de figure.

Dans le cas des dispositifs d’enregistrement sans fil, le conditionnement et la quantification des signaux sont effectués *in situ* dans le dispositif installé sur l’animal, souvent sur la tête, d’où l’appellation *headstage*, et le processeur est interfacé avec un transmetteur utilisé pour recevoir des commandes et pour transmettre les signaux neuronaux numérisés. La figure 1.3 résume sous forme conceptuelle et de schéma blocs une chaîne d’acquisition sans fil de signaux neuronaux.

Indépendamment du type de système, sans fil ou câblé, la chaîne de traitement se poursuit du côté de la station de base (ordinateur) où les AP peuvent être extraits du signal original afin qu’ils soient classifiés et associés à un neurone particulier, ce sont les étapes classiques

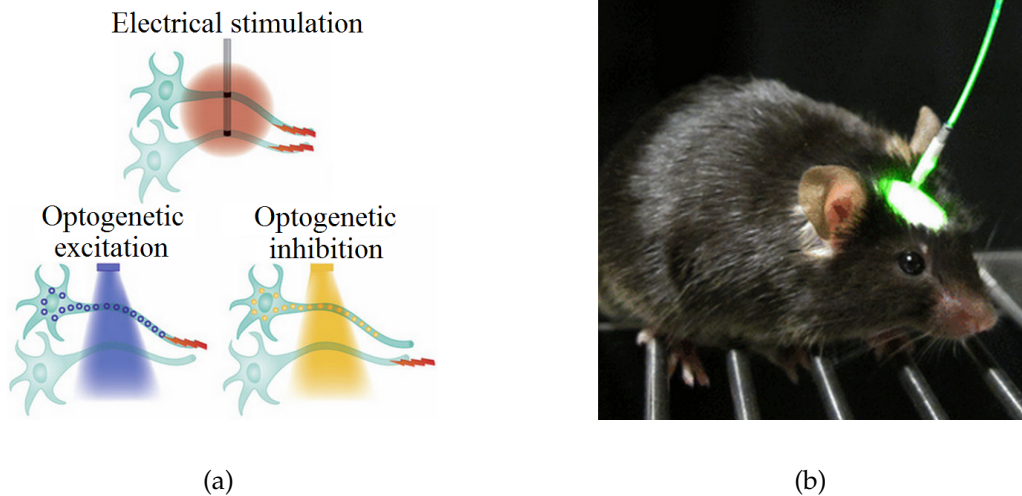


FIGURE 1.4 – a) Stimulation électrique versus stimulation/inhibition optique des neurones (image tirée de [1]), b) souris transgénique en pleine expérience optogénétique (image tirée de [2]).

d'analyse en neurosciences comportementale [24]. Une fois ces étapes effectuées, une analyse plus poussée peut être faite selon le type d'application ou de recherche [25].

### 1.1.2 Recherche optogénétique

L'optogénétique est une technique révolutionnaire de stimulation neuronale qui permet d'activer ou de désactiver des neurones incorporant une protéine sensible à la lumière avec rapidité et grande précision temporelle et spatiale [1] (voir la figure 1.4). Ce nouveau domaine de recherche, alliant optique et génétique, a été sacré *Méthode de Recherche de l'Année* en 2010 par *Nature Methods*, car elle permet de mener des expériences inédites sur le cerveau par l'activation ou la désactivation de groupes de neurones génétiquement modifiés ciblés par de la lumière. Contrairement à l'approche classique de stimulation électrique des neurones qui consiste à placer une microélectrode de stimulation à proximité des tissus à stimuler avec un courant électrique diffusé, la stimulation optique est robuste aux interférences de l'environnement et permet de cibler la zone à stimuler avec une plus grande précision. De plus, la stimulation optique ne produit pas d'artéfacts détectables et souvent dommageables pour l'équipement lors de l'enregistrement de l'activité neurale en même temps que la stimulation [26].

L'optogénétique est possible grâce à l'insertion de protéines sensibles à la lumière dans la membrane cellulaire des neurones [1]. Les protéines permettant de stimuler un neurone, comme ChR1, ChR2, ChETA, SFO, VChR1 ou VChR2, dépolarisent la cellule en laissant entrer des ions  $\text{Na}^+$  (phase de dépolarisation) lorsqu'elles sont exposées à une longueur d'onde adéquate. Les protéines permettant d'inhiber un neurone, comme NpHR, Arch ou GtR3, hyperpolarisent la cellule en laissant entrer des ions  $\text{Cl}^-$ , ce qui les rend inexcitables, lorsqu'elles

sont exposées à une bonne longueur d'onde. Par exemple, les protéines ChR2, ChETA et SFO sont dépolarisées par la lumière bleue (470 nm) tandis que les protéines Arch et GtR3 sont hyperpolarisées avec la lumière jaune (575 nm) et bleue (472 nm) respectivement.

Différentes techniques d'intégration de gènes existent afin que les neurones deviennent sensibles à la lumière. Entre autres, il est possible d'intégrer un virus exprimant les gènes produisant la bonne protéine à l'ADN chromosomique du sujet, ou bien d'utiliser une approche à base d'animaux transgéniques exprimant les gènes désirés [27]. Notamment, il est possible de se procurer ce type d'animaux transgéniques de fournisseurs privés, tel que *The Jackson Laboratory* [28], ou bien dans des animaleries de laboratoire de recherche en neurosciences.

Plusieurs laboratoires de recherche en neurosciences utilisent l'optogénétique afin de mener des expériences sur le cerveau d'animaux transgéniques. Parmi ceux-ci, le laboratoire de recherche du Dr Yves De Koninck au CRIUSMQ de Québec [29] est connu pour ses travaux de recherche portant sur les causes et les traitements reliés aux maladies neurologiques.

## 1.2 Problématique

Les neurologistes désirent présentement effectuer des observations avec des mammifères évoluant en toute liberté. Or, les appareils de mesures disponibles ne sont pas appropriés, car la plupart utilisent des câbles [30], [31], [32], [33]. En fait, les appareils de mesure électrophysiologique câblés réduisent la liberté de mouvement du sujet en observation, en plus de représenter des dangers d'infections à cause de la tension dans les câbles attachés à la partie implantée. Le même problème se pose avec les appareils de stimulation optogénétiques commerciaux [34], [35]. Dans ce type de systèmes, la lumière est propagée via des fibres optiques implantées dans le cerveau et qui sont reliées à des appareils externes, générant les mêmes problèmes que pour les appareils d'enregistrement câblés.

Une alternative à l'utilisation d'appareils câblés consiste à créer des systèmes d'enregistrement électrophysiologique et/ou de stimulation optique sans fil. Or, peu de dispositifs optogénétiques sans fil permettent de stimuler et d'enregistrer en même temps, et ceux qui le font comportent très peu de canaux d'enregistrement et de stimulation alors qu'il en faut beaucoup plus pour étudier les communications inter neuronales efficacement [36], [37], [38], [17], [39]. Dans le contexte des expériences optogénétiques, la nécessité d'enregistrer l'activité neuronale est aussi importante que la stimulation elle-même, puisqu'elle fournit une vision claire des événements qui se produisent pendant la stimulation. Notamment, l'enregistrement peut être focalisé sur les neurones stimulés ou non et être utilisé afin de déclencher des modes de stimulation spécifiques et pertinents pour le contexte de l'expérience.

La faible quantité du nombre de canaux d'enregistrement des dispositifs sans fil actuels est principalement due à la difficulté de gérer la grande quantité de données à transmettre pour

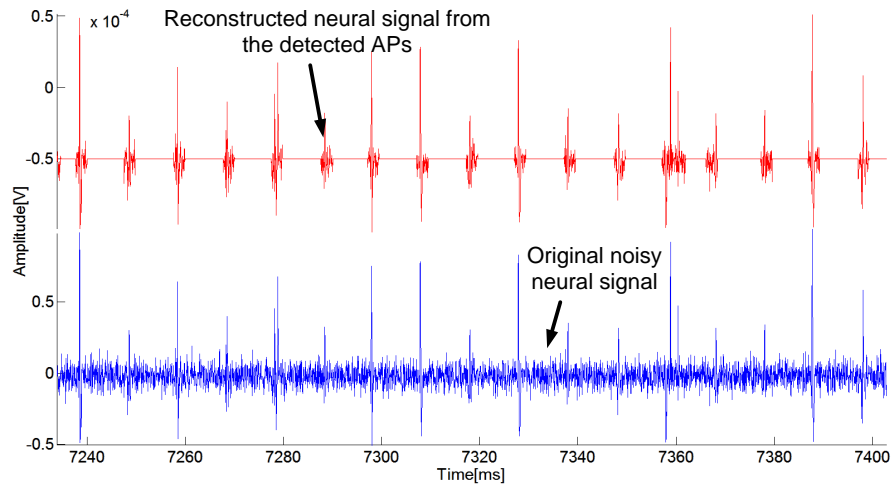


FIGURE 1.5 – Signal reconstruit avec les AP détectés (en rouge) versus le signal original (en bleu).

chaque microélectrode, pouvant atteindre plus de 320 kbps par canal (fréquence d'échantillonnage de 20 kHz avec des échantillons de 16 bits), à l'intérieur d'un dispositif électronique miniature, donc limité. À cause de ce problème, la majorité des systèmes capables de s'interfacer avec plusieurs dizaines de microélectrodes sont malheureusement câblés [30], [31], [32], [33], ou ils utilisent des transmetteurs radio fréquence faits sur mesure (*custom*) très dispendieux, énergivores, peu flexible et longs à développer [40], [41], [42], [43].

Bref, la majorité des systèmes sans fil actuels ne sont pas adaptés aux besoins de la recherche, car :

1. Ils n'offrent pas une résolution suffisante ou une durée de vie adéquate.
2. Ils ne transmettent pas de l'information appropriée (en transmettant seulement les temps d'occurrence des AP par exemple).
3. Ils sont trop lourds/volumineux ou trop dispendieux.

La suite de cette section présente une revue de l'état de l'art des dispositifs dédiés à l'enregistrement électrophysiologique et/ou à la stimulation optique sans fil, ainsi que les problèmes inhérents reliés aux dispositifs existants répertoriés. Deux catégories de systèmes sont présentées, soit les systèmes à base de circuit intégrés dédiés (ASIC) et les systèmes à base de composants commerciales (COTS).

### 1.2.1 Systèmes ASIC

Des systèmes d'enregistrement de signaux neuronaux ont été développés sous forme d'ASIC afin d'être le plus compacts, le moins énergivore et comporter le plus de canaux en parallèle possible. Le tableau 1.1 présente une liste non exhaustive de systèmes ASIC existants. Il est possible d'y constater que la majorité des systèmes répertoriés compressent les données neuronales afin d'augmenter le nombre de canaux de transmission. Par exemple, les

systèmes ASIC proposés dans [16], [41], [44], [42], [43], [45] détectent et transmettent seulement les AP au lieu de transmettre le signal complet alors que ceux présentés dans [46] et [47] compressent les signaux complets sans faire de détection d'AP. Détecter et transmettre seulement les AP permet de réduire la quantité de données à transmettre en envoyant seulement les formes d'ondes pertinentes, sans transmettre le signal bruité entre chacune des occurrences des AP, voir la figure 1.5.

Certains systèmes, tels que [41], [44], [43] et [45] ont un très grand nombre de canaux, mais ils transmettent seulement quelques caractéristiques des AP détectés comme leur temps d'occurrence, leur amplitude ou bien leur durée, ce qui les rend beaucoup moins attrayant pour la recherche. Généralement, les neurophysiologistes aiment travailler avec les formes d'onde d'AP complètes afin de pouvoir les classer et les associer à des neurones particuliers à l'aide de méthodes de traitement automatique dédiées. Bien que certains de ces systèmes offrent un grand nombre de canaux, la majorité d'entre eux ne disposent d'aucun transmetteur intégré prêt à s'interfacer avec eux. Il faudrait les relier à un transmetteur à haut débit externe pour qu'ils deviennent utiles, et ceux qui en disposent, utilisent des transmetteurs haut débit faits sur mesure non commerciaux. En fait, le nombre de canaux d'enregistrement pour ces systèmes sont théoriques, car ils n'ont jamais été validés dans des situations réelles (nombre de canaux que la puce peut traiter) et sont contraints par le type de transmetteur utilisé. En effet, la majorité des systèmes répertoriés dans le tableau 1.1 n'ont fait l'objet d'aucune réalisation physique et sont souvent demeurés au stade de prototypes de recherche. Les systèmes de type ASIC sont longs à développer et coûtent chers à produire, ce qui rend leur déploiement difficile à réaliser à moins de cibler des applications à très grand volume. De plus, aucun système de type ASIC n'a encore été répertorié avec stimulation optique intégrée, ce qui est peu pratique pour l'approche optogénétique.

TABLE 1.1 – Comparaison de systèmes d'enregistrement électrophysiologiques basés sur technologie ASIC.

Systèmes tirés de	# canaux d'enregistrement	# canaux de stimulation	Échantillonnage (ksps)	Résolution (bits)	Type de transmetteur	Détection des AP	Compression supplémentaire
[48]	12	0	40	10	Externe (N/A)	Non	Aucune
[46]	16	0	N/A	N/A	Externe (N/A)	Non	<i>Delta</i>
[47]	32	0	25	12	Externe (N/A)	Non	Ondelette
[49]	96	0	31	10	Externe (N/A)	Non	Aucune
[16]	16	0	30	8	Externe (N/A)	Oui	Aucune

[41]	64	0	7.8	8	Custom (FSK)	Oui	Occurrence des AP
[44]	64	0	24	8	Externe (N/A)	Oui	Caractéristiques des AP
[42]	100	0	15	10	Custom (FSK)	Oui	Occurrence des AP
[43]	128	0	40	9	Custom (UWB)	Oui	Caractéristiques des AP
[45]	256	0	20	5	Externe (N/A)	Oui	Caractéristiques des AP

### 1.2.2 Systèmes COTS

Contrairement aux systèmes ASIC qui sont intégrés sur puce, les systèmes à base de composants *commercial off-the-shelf* (COTS) consomment plus d'énergie et traitent typiquement moins de canaux, mais sont, en contrepartie, beaucoup moins dispendieux, plus flexibles, et leur temps de développement est moindre.

Le tableau 1.2 présente une liste non exhaustive de systèmes à base de composants COTS existants. On peut y constater que les systèmes qui possèdent de la stimulation optique et de l'enregistrement électrophysiologique sont limités à un nombre très restreint de canaux de stimulation et d'enregistrement [37], [38], [17], [39]. Afin d'augmenter le nombre de canaux, peu de ces systèmes effectuent de la détection *in situ* d'AP et aucun ne fait de la compression en dehors de la détection. Le seul système capable de transmettre sans compression un nombre important de canaux est celui présenté dans [40] et il utilise un transmetteur fait sur mesure augmentant significativement les coûts d'un tel système<sup>1</sup>, en plus de ne posséder aucun canal de stimulation optique.

TABLE 1.2 – Comparaison de systèmes d'enregistrement électrophysiologiques basés sur des composants COTS.

Systèmes tirés de	# canaux d'enregistrement	# canaux de stimulation	Échantillonnage (ksps)	Résolution (bits)	Type de transmetteur	Détection des AP	Compression supplémentaire
[37]	1	1	10	N/A	Commercial (UHF)	Non	Aucune
[38]	1	1	12	12	Bluetooth	Non	Aucune
[50]	1	0	N/A	N/A	Custom (FM)	Non	Aucune

1. 8000\$ l'unité selon un estimé produit par TBSI en juin 2015.



[51]	1	0	N/A	N/A	Custom (FM)	Non	Aucune
[17]	2	1	20	8	Commercial (GFSK)	Non	Aucune
[52]	0	4	N/A	10	Custom (Inductive)	Non	Aucune
[53]	15	0	N/A	N/A	Custom (N/A)	Non	Aucune
[36]	0	16	N/A	N/A	Commercial (ISM)	Non	Aucune
[40]	32	0	30	12	Custom (FSK)	Non	Aucune
[39]	1	1	11.7	N/A	Custom (Infrared)	Oui	Aucune
[54]	16	0	25	10	Bluetooth	Oui	Aucune

### 1.3 Objectifs du projet

Ce projet de recherche vise à concevoir de nouveaux systèmes optogénétiques sans fil dotés de méthodes de traitement des signaux neuronaux avancées afin de permettre des expériences sur le cerveau de petits mammifères évoluant en toute liberté. Les objectifs du projet sont les suivants :

- Développer des algorithmes de compression de données neuronales performants adaptés aux signaux produits par les neurones.
- Concevoir de nouvelles plateformes optogénétiques sans fil à base de composants COTS possédant les caractéristiques suivantes :
  - Adaptés pour mener des expériences sur le cerveau de petits mammifères évoluant en toute liberté.
  - Posséder plusieurs canaux d’enregistrement.
  - Posséder plusieurs canaux stimulation.
  - Être petits, légers et peu énergivores.
- Intégrer les nouveaux algorithmes de traitement de signal développés à l’intérieur des nouvelles plateformes optogénétiques afin d’améliorer leur performances.
- Valider les nouveaux systèmes et les nouveaux algorithmes conçus à l’aide de tests *in vitro* et *in vivo*.

À terme, ce projet de recherche va permettre de concevoir des dispositifs optogénétiques plus compacts, moins énergivores, et permettant de transmettre les AP provenant de beaucoup plus de neurones. Ces systèmes pourront être utilisés avec de plus petits animaux,

seront moins coûteux, et permettront de mener de nouvelles expériences en neurosciences, ce qui contribuera à faire avancer la recherche sur le cerveau en général.

## 1.4 Défis

### 1.4.1 Compression des signaux électrophysiologiques

Le défi consistant à transmettre les signaux issus de plusieurs neurones se pose au niveau de la quantité de données à transmettre. Puisque le contenu fréquentiel des AP atteint 5 kHz, la fréquence d'échantillonnage minimale requise pour ne pas perdre d'information est de 10kHz. Étant donné que les filtres ne sont jamais parfaits et pour obtenir une meilleure précision, il est courant de voir des systèmes d'enregistrements échantillonner à une fréquence  $\geq 20$  kHz. Avec des échantillons d'une résolution de 16 bits et une fréquence d'échantillonnage de 20 kHz, la quantité de données à transmettre par microélectrode est de 320 kbps. Les transmetteurs à faible consommation comme les transmetteurs *Bluetooth* ou *ZigBee* ont des débits (*on-air*)  $\leq 3$ Mbps, permettant de transmettre un maximum théorique de 9 électrodes. En réalité, les débits effectifs atteints sont moindres et ces types de transmetteurs sont rarement utilisés pour transmettre les signaux de plus de deux microélectrodes.

Les algorithmes de compression de signaux neuronaux recherchés permettront de réduire la bande passante nécessaire et de réduire la consommation électrique en transmettant moins de données et d'augmenter significativement le nombre de microélectrodes pouvant être utilisées, pour un transmetteur donné. Pour chaque type de signal biologique, un algorithme de compression optimal exploitant les propriétés intrinsèques de la forme d'onde doit être développé. Puisque la compression est effectuée à l'intérieur de systèmes embarqués, des contraintes de complexité et de puissance de calcul font leur apparition. Dans le cas de l'enregistrement neuronal sans fil, l'algorithme doit être réalisable en utilisant peu de ressources, doit être rapide à exécuter, doit compresser suffisamment, tout en conservant la qualité des signaux originaux afin de permettre une analyse des signaux après compression (classer les AP en groupes, ou *clusters*, par exemple, à l'aide d'algorithmes, dont l'analyse PCA).

### 1.4.2 Intégration dans une plateforme optogénétique

En plus de la complexité engendrée par la création d'un nouvel algorithme de détection et/ou de compression des signaux neuronaux, une part importante de ce mémoire porte sur l'intégration de ces algorithmes dans de nouveaux systèmes optogénétiques sans fil à multiples canaux d'enregistrement et de stimulation miniatures et à faible consommation énergétique. Les défis sont nombreux, notons ceux-ci :

- Développer des systèmes assez compact et léger pour être utilisé sur une souris ou un autre petit mammifère.

- Développer des systèmes économes en énergie afin de posséder plusieurs heures d'enregistrements et de stimulation.
- Développer des systèmes assez puissants pour compresser les signaux neuronaux.
- Développer des systèmes possédant plusieurs canaux d'enregistrements.
- Développer des systèmes possédant plusieurs canaux de stimulation.
- Utiliser des composantes COTS afin d'augmenter la flexibilité et réduire les coûts de développements.

## 1.5 Contributions

Ce travail de recherche apporte plusieurs contributions scientifiques, allant de la création de nouveaux algorithmes permettant de compresser les signaux neuronaux, au développement de nouvelles plateformes de recherche optogénétiques qui permettent d'effectuer de nouvelles expériences sur de petits rongeurs transgéniques évoluant en toute liberté.

Certains algorithmes présentés dans ce mémoire ont fait l'objet de deux déclarations d'invention distinctes auprès de l'agent de brevet de l'Université Laval (# de divulgation 01691-DIV-02 et 01736-DIV-01 respectivement). L'une des deux déclarations d'invention a été déclarée brevetable par la société de valorisation *Sovar* de l'Université Laval le 17 juillet 2015 et un brevet provisoire aux États-Unis a été obtenu (# de référence : 000819-0303), alors que l'autre est en attente de traitement.

Les algorithmes de compression ainsi que les systèmes optogénétiques sans fil présentés dans ce mémoire ont fait l'objet de plusieurs publications scientifiques dont l'auteur de ce mémoire est le premier auteur, en voici la liste exhaustive :

### – Articles de conférence :

- G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. De Koninck and B. Gosselin, "An Optimized Adaptive Spike Detector for Behavioural Experiments", *IEEE International Symposium On Circuits and Systems (ISCAS'16), Montréal Canada, Mai 2016 (pending review)*.
- G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. De Koninck and B. Gosselin, "A Wireless Optogenetic Headstage with Multichannel Neural Signal Compression", *IEEE Conference on Biomedical Circuits and System (BioCAS'15), Atlanta USA, Octobre 2015 (accepted paper)*.
- G. Gagnon-Turcotte and B. Gosselin, "Multichannel Spike Detector With an Adaptive Threshold Based On a Sigma-Delta Control Loop", *IEEE Annual Conference on Engineering in Medicine and Biology Society (EMBC'15), Milan Italy, Août 2015 (accepted paper)*.

- G. Gagnon-Turcotte, M. Sawan and B. Gosselin, "Low-Power Adaptive Spike Detector Based on a Sigma-Delta Control Loop", *IEEE Annual Conference on Engineering in Medicine and Biology Society (EMBC'15), Milan Italy, Août 2015 (accepted paper)*.
- G. Gagnon-Turcotte, C.-O. Dufresne Camaro, A. Avakh Kisomi, R. Ameli and B. Gosselin, "A Wireless Multichannel Optogenetic Headstage With On-The-Fly Spike Detection", *IEEE International Symposium On Circuits and Systems (ISCAS'15), Lisbon Portugal, pp. 1758-1761, Mai 2015*.
- G. Gagnon-Turcotte, C.O Dufresne-Camaro and B. Gosselin, "Comparison of Low-Power Biopotential Processors for On-The-Fly Spike Detection", *IEEE International Symposium On Circuits and Systems (ISCAS'15), Lisbon Portugal, pp. 802-805, Mai 2015*.
- **Articles de journaux :**
  - G. Gagnon-Turcotte, A. Avakh Kisomi, R. Ameli, C.O Dufresne-Camaro, Y. LeChasseur, J.L Neron, P. B. Bareil, P. Fortier, C. Bories, Y. De Koninck and B. Gosselin, "A Wireless Optogenetic Stimulator Headstage with Multichannel Neural Recording Capability", *Sensors, vol. 15, no. 9, pp. 22776-22797, 2015*.
  - G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. Messaddeq, Y. De Koninck and B. Gosselin, "A Wireless Optogenetic Headstage with 32 Channels Neural Recording Capability", *IEEE Transactions on Biomedical Circuits and Systems, 2015 (pending review)*.
  - G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. De Koninck and B. Gosselin, "A Low-Power Adaptive Spike Detector for In-Vivo Behavioural Neuroscience Experiments", *Transactions on Neural Systems & Rehabilitation Engineering, 2015 (pending review)*.

En plus des publications scientifiques, les algorithmes de compression ainsi que les systèmes optogénétiques sans fil ont fait l'objet d'exposés lors d'évènements d'envergure internationaux, en voici la liste exhaustive :

- G. Gagnon-Turcotte, R. Ameli, A. Avakh Kisomi, C.-O. Dufresne, B. Gosselin, "A Multichannel Optogenetic Wireless Headstage With on-the-fly Spike Detection", *IEEE EMBS Brain Grand Challenge Conference, Washington D.C, Novembre 2014*.
- G. Gagnon-Turcotte, R. Ameli, A. Avakh Kisomi, C.-O. Dufresne, B. Gosselin, "A Multichannel Optogenetic Wireless Headstage With on-the-fly Spike Detection", *3<sup>ième</sup> Forum de MEDTEQ, Québec, Novembre 2014*.
- G. Gagnon-Turcotte, R. Ameli, A. Avakh Kisomi, C.-O. Dufresne, B. Gosselin, "A Multichannel Optogenetic Wireless Headstage With on-the-fly Spike Detection", *Journée de l'innovation ReSMiQ, Montréal, Septembre 2014*.

Il est à noter que l'exposé effectué lors du 3<sup>ième</sup> Forum de MEDTEQ à Québec a reçu un prix FRQ décerné aux meilleurs projets et affiches scientifiques.

## 1.6 Structure du mémoire

Ce mémoire présente au chapitre 2 une revue de littérature faisant la synthèse des différents types de compression pour signaux biologiques ainsi que leurs plateformes matérielles associées. La détection en temps réel de potentiels d'action, la compression par ondelettes, la compression *Delta*, la compression de type *Compress sensing* ainsi que les méthodes basées sur la quantification vectorielle (QV) en plus de leurs implémentations sur différentes plateformes matérielles telles que sur microcontrôleur, FPGA, CPLD ou ASIC, seront explorées en détail et comparées.

Le chapitre 3 présente les résultats de simulations *Matlab* obtenus avec les algorithmes de compression identifiés comme étant les plus prometteurs en utilisant des signaux biologiques synthétiques et réels.

Le chapitre 4 présente la conception d'un *headstage* optogénétique sans fil doté de deux canaux d'enregistrement et de deux canaux de stimulation ainsi que de compression par détection *in situ* des AP. Tous les sous-systèmes et les algorithmes développés sont présentés en détail ainsi que les résultats *in vitro* et *in vivo* obtenus avec le système. Une version commerciale plus légère et réalisée dans le cadre d'un projet Mitacs [55] avec le partenaire *Doric Lenses* est aussi présentée.

Le chapitre 5 présente la conception d'un *headstage* optogénétique sans fil à 32 canaux d'enregistrement et possédant 32 canaux de stimulation doté de détection *in situ* des AP et de compression numérique. Tous les sous-systèmes sont présentés en détail, incluant les algorithmes de détection et de compression.

Le chapitre 6 est dédié aux résultats *in vitro* et *in vivo* obtenus avec le système 32 canaux.

Le chapitre 7 présente une analyse comparative des systèmes développés incluant le calcul d'une fonction de coût permettant d'isoler les systèmes les plus performants.

Finalement, la conclusion de ce mémoire est présentée, suivie des annexes.



## Chapitre 2

# Revue de littérature

Ce chapitre propose une revue des différents algorithmes de compression de signaux biologiques répertoriés dans la littérature ainsi que leurs systèmes matériels associés.

### 2.1 Compression en transmettant seulement les AP

#### 2.1.1 Algorithmes de détection

Une méthode simple afin de réduire la quantité de données à transmettre est d'effectuer une détection des AP au niveau du capteur et de transmettre seulement les AP détectés. Par exemple, le capteur développé dans [42] permet de compresser de 15Mb/s à 330kb/s les données de 100 électrodes en transmettant seulement l'état, détecté ou non, en temps réel. Le système, complètement développé avec la technologie CMOS 0.5  $\mu\text{m}$ , détecte les AP en comparant les signaux avec un seuil. Chaque comparateur est interrogé à toutes les millisecondes et la valeur (détectée ou non) est transmise dans un paquet avec les données des autres électrodes.

Au lieu de transmettre l'état de détection pour chacun des canaux, certains dispositifs transmettent des caractéristiques spécifiques à chacun des AP détectés. Par exemple, [44] regroupe les potentiels en *clusters* et transmet seulement l'information reliée au *clusters*, sans transmettre l'AP. [43] quant à lui transmet la valeur crête à crête maximale ainsi que la dérivée maximale et minimale des AP détectés, tandis que [45] transmet l'amplitude maximale et minimale des AP ainsi que le délai temporel entre ces deux valeurs.

Envoyer seulement l'information sur la détection ou certaines caractéristiques des AP est souvent insuffisant. Transmettre la forme d'onde complète pour chacun des AP détectés est toutefois désiré. Une approche réalisée dans [13] consiste à effectuer une détection des AP analogiquement et à utiliser un filtre FIR (*switched capacitors*) comme tampon de la forme d'onde complète. Cette technique permet d'activer l'ADC ainsi que les autres composantes numériques seulement lorsqu'il y a une détection d'AP afin d'économiser de l'énergie.

Enfin, il existe plusieurs méthodes de détection des AP dans le domaine numérique. [56] présente et compare trois groupes d'algorithmes de détection, soit à base de seuil, d'énergie et de filtres. Les méthodes incluses dans chacun des groupes sont les suivantes :

– **Seuil :**

- Seuillage absolu [56] :

$$y(n) = |x(n)|, \begin{cases} \text{Si } y(n) > \text{seuil, un AP est détecté} \\ \text{sinon, aucun AP est détecté} \end{cases}$$

- Seuillage positif [41], [42], [45], [39], [54] :

$$y(n) = x(n), \begin{cases} \text{Si } y(n) > \text{seuil, un AP est détecté} \\ \text{sinon, aucun AP est détecté} \end{cases}$$

- Seuillage négatif [41], [45] :

$$y(n) = x(n), \begin{cases} \text{Si } y(n) < \text{seuil, un AP est détecté} \\ \text{sinon, aucun AP est détecté} \end{cases}$$

– **Énergie :**

- NEO (connus parfois sous l'acronyme TEO) [44], [43], [57], [58] :

$$NEO(n) = x(n)^2 - x(n-1) \times x(n+1), \begin{cases} \text{Si } NEO(n) > \text{seuil, un AP est détecté} \\ \text{sinon, aucun AP est détecté} \end{cases}$$

- MTEO :

$$MTEO(n) = x(n)^2 - x(n-k) \times x(n+k), \begin{cases} \text{Si } MTEO(n) > \text{seuil, un AP est détecté} \\ \text{sinon, aucun AP est détecté} \end{cases}$$

– **Filtres :**

- Convolution du signal avec une banque de filtres.
- Convolution du signal généré par NEO avec une banque de filtres.
- Convolution du signal en valeur absolue avec une banque de filtres.

### 2.1.2 Algorithmes de seuillage

Certains algorithmes de détection, comme ceux à base de seuil ou d'énergie, doivent être utilisés conjointement à un seuil. Une méthode simple consiste à utiliser un seuil fixe pouvant être réglé par l'utilisateur en fonction du niveau de bruit inclus dans le signal. Cette



méthode est utilisée dans [42], [41], [39] et [54], où le seuil doit être ajusté manuellement par l'utilisateur en fonction du niveau de bruit.

Les microélectrodes utilisées pour capter les signaux neuronaux possèdent une grande impédance (jusqu'à plusieurs dizaines de  $M\Omega$ ) ce qui peut générer un ratio signal sur bruit (SNR) faible. De plus, le SNR du signal n'est pas constant dans le temps et peut varier en fonction de plusieurs facteurs tels que le déplacement des électrodes, en fonction du phénomène de *microelectrode drifting* et à cause du bruit neuronal. Une technique de seuillage adaptative est donc hautement désirée. La majorité des algorithmes de seuillage adaptatif sont basés soit sur l'écart-type du signal, sur la valeur RMS du signal ou bien sur la valeur moyenne du signal.

Une méthode, utilisée dans le programme de *clustering* automatique *open source* nommé *Wave Clus* [59], [60], calcule une approximation de l'écart-type du signal en utilisant la déviation médiane absolue (MAD) selon cette équation :

$$\delta = \text{median}\left\{\frac{|x|}{0.6745}\right\} \quad (2.1)$$

Où  $x$  est le signal filtré passe-bande. Avec cette technique, le seuil est ensuite calculé comme un multiple de  $\delta$ . Une autre technique permettant de calculer une estimation de l'écart type est présentée dans [61]. Cette technique, entièrement analogique, estime l'écart-type en utilisant une boucle de contrôle et en exploitant certaines propriétés statistiques de la loi normale.

La méthode proposée dans [62] calcule la valeur RMS du signal à partir d'un nombre restreint d'échantillons :

$$X_{RMS} = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \quad (2.2)$$

Où  $x_i$  est le  $i^{eme}$  échantillon filtré passe-bande. Le seuil est ensuite évalué comme un multiple de  $X_{rms}$ .

Une autre méthode de seuillage automatique adaptée à l'algorithme NEO est présentée dans [44]. Cet algorithme calcule la valeur moyenne de NEO selon l'équation suivante :

$$Thr = \frac{1}{N} \sum_{i=1}^N NEO(i) \quad (2.3)$$

Où  $N$  est le nombre d'échantillons utilisés dans les calculs. Le seuil est ensuite évalué comme un multiple de  $Thr$ . De façon similaire, [58] calcul le seuil de l'algorithme NEO selon sa moyenne trouvée analogiquement avec un filtre passe-bas possédant une très grande constante de temps.

Transmettre seulement les formes d'onde détectées constitue une méthode simple pour réduire la quantité de données à transmettre. Cette technique est parfois employée avec d'autres méthodes de compression qui sont présentées dans les sections subséquentes.

## 2.2 Compression par Ondelettes

### 2.2.1 Présentation des ondelettes, aspects mathématiques

La transformée en ondelettes est une technique mathématique employée afin de conserver à la fois l'information temporelle et fréquentielle d'un signal. Cette transformée est particulièrement appropriée lorsque les différentes composantes d'un signal sont localisées à la fois en temps et en fréquence [63]. Elle est notamment employée en traitement de signal, en analyse d'images, dans les systèmes de communication, en analyse temps-fréquence, ainsi qu'en compression de signaux numériques.

L'analyse par ondelettes continue (CWT) est calculée en appliquant une convolution entre un signal et un ensemble de fonctions générées par une *mother wavelet* selon l'équation suivante :

$$CWT_{a,b}\{x(t)\} = \int_{-\infty}^{+\infty} x(t) \left(\frac{1}{\sqrt{a}}\right) \psi\left(\frac{t-b}{a}\right) dt. \quad (2.4)$$

Où  $\left(\frac{1}{\sqrt{a}}\right)\psi\left(t - \frac{b}{a}\right)$  est la *mother wavelet*. La convolution est appliquée sur différents segments du signal dans le domaine temporel, facteur de translation  $b$ , et en fréquence, facteur de dilatation  $a$  i.e. il s'agit de représenter un signal comme une somme pondérée de la *mother wavelet* dilatée et translatée. Cette transformée est analogue à la *short time Fourier transform* (STFT) [64] présentée à l'équation 2.5.

$$STFT_{a,b}\{x(t)\} = \int_{-\infty}^{+\infty} x(t) g(t-b) e^{-jat} dt. \quad (2.5)$$

Contrairement à la *short time Fourier transform*, la transformée en ondelettes permet une excellente localisation dans le domaine temporel via le facteur de translation  $b$  et en fréquence via le facteur de dilatation  $a$ . Ainsi, la fenêtre du signal décroît lorsque la fréquence augmente à cause du facteur de dilatation  $a$ . Cette caractéristique avantage la CWT par rapport à la STFT, dans laquelle la largeur de fenêtre reste constante [65] ce qui résulte en une impossibilité d'associer les caractéristiques du domaine fréquentiel avec leur emplacement dans le temps. Il s'agit d'une particularité très importante qui avantage la CWT lorsque le signal à traiter possède des composantes hautes fréquences pour de courtes périodes de temps et basses fréquences pour de longues périodes [64]. Cette caractéristique rend possible la modification du contenu fréquentiel de certaines parties du signal seulement, ce qui est avantageux lorsque l'on veut compresser des signaux en retirant de l'information inutile tout en conservant certaines parties avec une meilleure précision (par exemple, compresser sans affecter significativement les formes d'ondes ressemblant à des AP). C'est pourquoi la transformée par ondelettes est appliquée de nos jours à une variété de signaux biomédicaux (AP, EMG, ECG, etc.) [66] qui ont la particularité de présenter des composantes hautes fréquences pour de courtes périodes de temps et basses fréquences pour de longues périodes.

La CWT est la transformée par ondelettes dans le domaine continue. Dans le domaine discret, c'est la transformée en ondelettes discrète (DWT) qui est utilisée. Une approche pré-

sentée dans [67] afin de discrétiser la CWT est d'utiliser les paramètres de dilatation et de translation discrets suivants :

$$\begin{cases} a_j = a_0^{-j} \\ b_{j,k} = k\tau_0 b_0^{-j} \end{cases} \quad a_j < 1, \tau_0 \neq 0, j \in \mathbb{Z}, k \in \mathbb{Z} \quad (2.6)$$

Où  $\mathbb{Z}$  est l'ensemble des entiers discrets. Ce qui mène à la famille de *mother wavelet* discrète suivante :

$$\psi_{j,k} = \frac{1}{\sqrt{a_0^{-j}}} \psi\left(\frac{t - k\tau_0 b_0^{-j}}{a_0^{-j}}\right) \quad (2.7)$$

Généralement, les valeurs retenues sont  $a_0 = 2$ ,  $b_0 = 2$  et  $\tau_0 = 1$  [68], correspondant à une discrétisation *dyadic*, générant la *mother wavelet* discrète suivante :

$$\psi_{j,k} = \psi\left(\frac{t - k2^j}{2^j}\right) \quad (2.8)$$

En utilisant l'équation 2.8, la DWT se formule :

$$DWT(j, k) = \frac{1}{\sqrt{2^j}} \int_{-\infty}^{+\infty} x(t) \psi\left(\frac{t - k2^j}{2^j}\right) dt. \quad (2.9)$$

Où  $DWT(j, k)$  représente les coefficients de la transformée avec  $j$  le facteur de dilatation et  $k$  le facteur de translation. Concrètement, pour calculer une DWT, c'est une analyse à multirésolution sur une base orthogonale qui est utilisée. Pour ce faire, deux méthodes sont employées, la méthode de *Mallat* qui est présentée à la section 2.2.2, et la méthode par *lifting* qui est présentée à la section 2.2.3. Différentes bases d'ondelettes possédant la propriété d'orthogonalité sont couramment utilisées. En voici une liste non exhaustive :

- Haar
- Daubechies
- Coiflet
- Symlet
- Biorthogonal
- Meyer

## 2.2.2 Calcul de la transformée en ondelettes par la méthode de Mallat

La méthode de *Mallat* [69] est un algorithme de type pyramidal utilisé dans [70], [71], [72] et [73] afin de calculer une DWT à l'aide de convolutions et de décimations successives.

Cette méthode repose sur le fait qu'il est possible de calculer les coefficients d'un ensemble discret de fonctions issues de la *mother wavelet*. En prenant des versions de la *mother wavelet* dilatées et translatées par une puissance de 2, on obtient l'équation 2.8, soit  $\psi_{j,k} =$

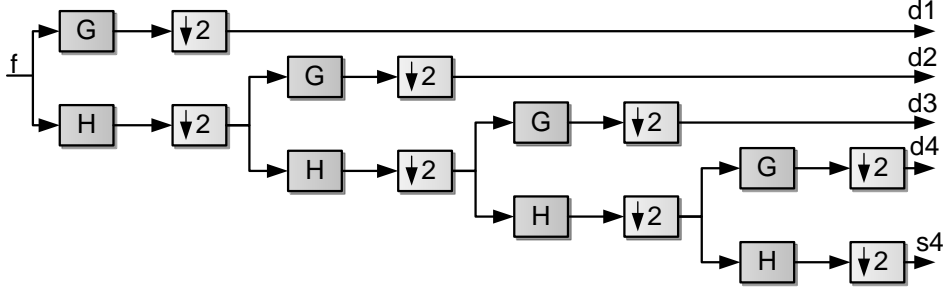


FIGURE 2.1 – Représentation schématique d'une transformée en ondelettes de 4 niveaux de profondeur par la méthode de *Mallat*.

$\frac{1}{\sqrt{2^j}} \psi\left(\frac{t-k2^j}{2^j}\right)$ . En fixant  $j$ , le calcul de la DWT dans l'équation 2.9 devient fonction de  $k$  seulement et peut être interprété comme une convolution du signal  $x(t)$  par un filtre. En discrétisant le filtre obtenu, l'on obtient le filtre passe-bas  $h(n)$  d'où l'on peut dériver sa version passe-haut  $g(n)$ . Avec  $j = 1$ , le filtre devient  $h(n) = \frac{1}{\sqrt{2}} \psi\left(\frac{t-k2}{2}\right)$ . L'opération consistant à prendre  $j = 1, 2, \dots, \infty$  revient à filtrer successivement le signal par  $h(n)$ , tandis que le facteur  $k2^j$  peut être réalisé par une opération de sous-échantillonnage entre les étapes de filtrage.

La convolution des filtres  $h(n)$  et  $g(n)$  avec un signal de taille  $N$ , suivi d'une décimation d'un facteur de deux permet d'obtenir  $\frac{N}{2}$  coefficients détaillés et  $\frac{N}{2}$  coefficients approximatifs. L'algorithme peut être répété sur  $y$  niveaux, avec  $y \leq \log_2(N)$ , sur l'ensemble des coefficients approximatifs obtenus après l'étape de décimation. Pour chacun des niveaux, les coefficients détaillés obtenus après l'étape de décimation sont les coefficients de la transformée en ondelette. Par exemple, le premier niveau génère  $\frac{N}{2}$  coefficients de la transformée, le second  $\frac{N}{4}$  et ainsi de suite. Au dernier niveau  $y$ , les coefficients approximatifs sont incorporés aux coefficients de la transformée déjà obtenus pour obtenir  $N$  coefficients de résolution  $y$ . Une représentation schématique de la transformée par la méthode de *Mallat* est présentée à la figure 2.1 et les opérations mathématiques à effectuer pour chacun des niveaux sont présentées dans les équations suivantes :

$$a_{j,k} = \sum_m h(m - 2k) a_{j-1,m} \quad (2.10)$$

$$d_{j,k} = \sum_m g(m - 2k) a_{j-1,m} \quad (2.11)$$

Où  $a_{j,k}$  et  $d_{j,k}$  sont les coefficients approximatifs et détaillés respectivement,  $j$  est le niveau,  $m$  l'indice de l'échantillon traité et  $k$  l'indice du coefficient généré.

La méthode de *Mallat* permet de reconstruire parfaitement le signal original en suivant le même algorithme, mais en remplaçant les filtres  $h(n)$  et  $g(n)$  par leur contrepartie  $\tilde{h}(n)$  et  $\tilde{g}(n)$  et en remplaçant l'étape de sous échantillonnage par une étape de sur échantillonnage avant la convolution. La reconstruction débute au niveau  $y$  et remonte la pyramide dans le sens inverse. La figure 2.2 présente de façon schématique cette opération.

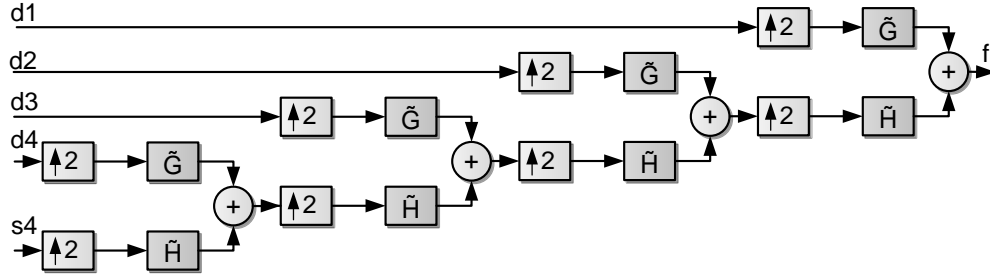


FIGURE 2.2 – Représentation schématique d’une transformée en ondelettes inverse de 4 niveaux de profondeur par la méthode de *Mallat*.

### 2.2.3 Calcul de la transformée en ondelette par la méthode de *Lifting*

La méthode par *lifting* [74] permet, au même titre que la méthode de *Mallat*, d’obtenir les coefficients de la *DWT* sur  $y$  niveaux. Cette technique possède la caractéristique de minimiser la mémoire utilisée et le nombre d’opérations arithmétiques nécessaires comparativement à la méthode de *Mallat*, puisque ces opérations sont réalisées *in situ*. Cette caractéristique réduit considérablement les requis matériels, temporels et logiciels pour une implémentation sur plusieurs niveaux [75]. Le calcul de la *DWT* par *lifting* possède deux types d’implémentation ayant leur avantage et inconvénient, soit une implémentation de type pipeline et une implémentation de type séquentielle. Il a été démontré que pour un nombre de canaux et un nombre de niveaux restreint, une implémentation de type pipeline consomme moins d’énergie (fréquence de fonctionnement plus faible) au prix de l’utilisation d’un plus grand nombre de ressources matérielles [47]. Inversement, une implémentation de type pipeline génère un produit taille-puissance-ressources qui croit significativement en augmentant le nombre de canaux et le nombre de niveaux, ce qui rend l’approche séquentielle plus avantageuse lorsqu’un nombre élevé de canaux et de niveaux sont désirés.

La clé de la méthode par *lifting* est présentée dans [73] et réside en trois étapes : *split*, *predict* et *update*. L’étape *split* consiste à décomposer un signal en ses parties d’indices paires et impaires. L’étape *predict* transforme la partie impaire du signal avec une combinaison linéaire d’échantillons de la partie paire. L’étape *update* transforme la partie paire du signal avec une combinaison linéaire des échantillons obtenus à l’étape *predict*. Tout comme pour l’algorithme de *Mallat*, la séquence *split*, *predict* et *update* peut être appliquée sur  $y$  niveaux. Les opérations linéaires à effectuer entre les parties dépendent du type d’ondelette utilisé. Par exemple, pour l’ondelette de *Haar*, appliquer la méthode de *lifting* revient à effectuer l’opération présentée à l’équation 2.12 et présentée schématiquement à la figure 2.3.

$$\begin{pmatrix} Approx \\ Detail \end{pmatrix} = \begin{pmatrix} \sqrt{2} & 0 \\ 0 & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 & 0.5 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} X_o \\ X_e \end{pmatrix} \quad (2.12)$$

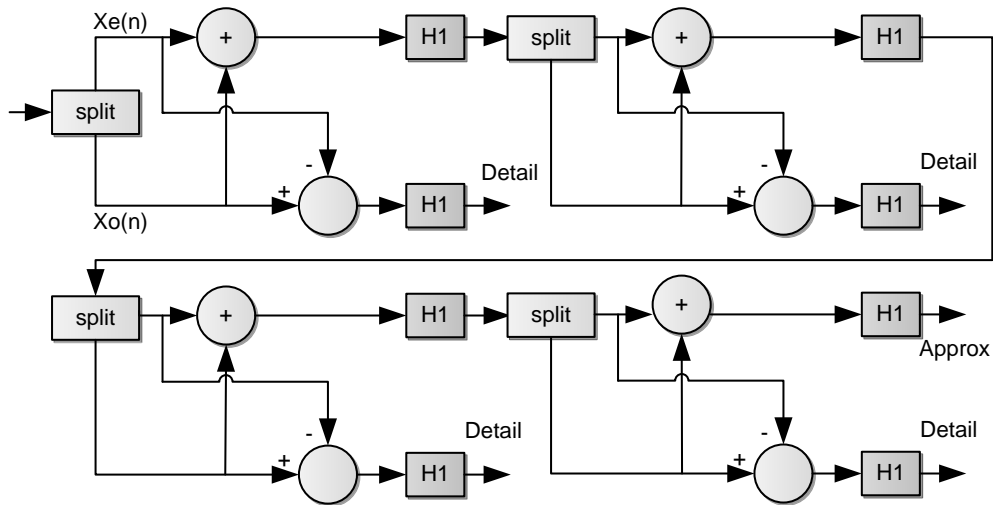


FIGURE 2.3 – Représentation schématique d’une transformée en ondelettes de type *Haar* de 4 niveaux de profondeur par la méthode de *lifting*.

Dans l’équation 2.12,  $X_e$  et  $X_o$  représentent la partie paire et impaire respectivement. La matrice résultante contient les facteurs approximatifs et détaillés du niveau.

Le calcul d’une DWT par la méthode de *lifting* séquentielle est utilisé dans [76] afin d’effectuer une transformée en ondelettes en temps réel dans un FPGA Virtex de Xilinx. Le système développé dans [76] traite 8 canaux en parallèle afin de détecter des AP à partir des coefficients de la transformée. La *mother wavelet* utilisée est une Daubechie-2 sur 5 niveaux effectuée sur 32 échantillons échantillonnés à 20 kHz. Certains compromis d’implémentation sont détaillés dans l’article, notamment l’approximation des coefficients de l’ondelette par des puissances de deux afin d’éliminer l’utilisation de nombres fractionnaires et l’utilisation d’opérations de décalage de bits au lieu de multiplications.

Une autre implémentation du *lifting* utilisée afin de compresser des signaux neuronaux est proposée dans [47]. Cet article présente la réalisation d’un *lifting* sur 4 niveaux avec une ondelette de type *Symmlet-4*, entièrement réalisé en technologie CMOS 0.18  $\mu\text{m}$ . Dans cette implémentation, le nombre de bits utilisé pour les échantillons et les coefficients sont de 10 bits et de 5 bits respectivement. De plus, ils ont opté pour une implémentation séquentielle afin de consommer moins d’énergie.

## 2.2.4 Compression par ondelettes dans la littérature

La clé de la compression par ondelettes réside dans l’information incluse dans les coefficients des différents niveaux. Les coefficients à haute énergie contiennent plus d’information que ceux à faible énergie qui contiennent les détails visuels du signal. En retirant certains coefficients de moindre importance, comme les coefficients nuls par exemple, le signal est com-

pressé et débruité par le fait même.

La compression par ondelettes est utilisée dans [77] afin de compresser des signaux ECG. En fait, [77] compresse en sélectionnant les coefficients à conserver selon un seuil basé sur un pourcentage fixe de l'amplitude maximale des coefficients.

Une technique adaptative d'élimination des coefficients de la DWT basée sur un seuillage adaptatif des coefficients de façon indépendante pour chacun des niveaux (sous-bandes) est proposée dans [78]. Les valeurs initiales des seuils sont calculées séparément pour chacune des sous-bandes en trouvant la moyenne ( $\mu$ ) et l'écart type ( $\sigma$ ) de l'amplitude des coefficients non nuls. Si  $\sigma \geq \mu$ , le seuil pour cette bande est ajusté à  $2\mu$ , sinon, le seuil est ajusté à  $\mu - \sigma$ . Les coefficients conservés sont ensuite encodés par un algorithme SPIHT (voir section 2.7.1) avant transmission, ce qui permet de garantir un taux de transmission maximal en conservant un maximum fixe de coefficients.

Une technique alternative est proposée dans [78], nommée *Energy packing efficiency strategy* (EPE), afin de sélectionner les coefficients à conserver. Cette technique est aussi utilisée dans [79] et [80] pour la compression de signaux ECG. Elle consiste à calculer l'énergie contenue dans chacune des sous-bandes et de ne retenir qu'un nombre minimum de coefficients pour conserver un pourcentage fixe de l'énergie initiale (taux de compression). L'équation suivante présente la formule utilisée pour calculer l'énergie de chacune des sous-bandes :

$$E_j = \sum_{i=1}^L C_i^2 \quad (2.13)$$

Où  $j$  est l'indice de la sous-bande,  $L$  est le nombre de coefficients contenus dans la sous-bande et  $C_i$  est le  $i^{eme}$  coefficient de la sous-bande.

Une méthode de compression de signaux neuronaux à base d'ondelettes et d'une base de données incrémentales (*vocabulary-based compression algorithm*) est présentée dans [81]. Pour chaque AP détecté, une DWT est effectuée. Les coefficients résultants sont alors comparés avec ceux contenus dans une base de données afin d'identifier un potentiel similaire. Si un potentiel similaire est trouvé, c'est un code représentant ce potentiel qui est transmis. Sinon, la base de données est mise à jour avec ce nouvel AP et les coefficients de la transformée ainsi que le nouveau code de l'AP sont transmis afin de synchroniser la base de données au niveau de la station de base.

## 2.3 Encodage Delta

L'encodage Delta consiste à encoder non pas le signal directement, mais plutôt la différence entre les échantillons subséquents. Lorsque la différence entre les échantillons est petite, l'encodage delta permet de réduire la plage dynamique du signal. Cette particularité est utilisée

dans [82] afin de compresser sans perte des signaux ECG avec un taux de compression variant autour d'un facteur 3.

L'encodage Delta est aussi implémenté en technologie CMOS 0.18  $\mu\text{m}$  dans [46] afin de compresser les signaux neuronaux provenant de 16 canaux d'enregistrement. La technique employée dans [46] consiste à encoder la différence entre deux échantillons successifs à l'aide d'un modulateur Sigma-Delta. Le modulateur Sigma-Delta sur échantillonne cette différence d'un facteur 4, 8, 16 ou bien 32, ce qui permet d'obtenir des taux de compression de 1.3, 2, 2.7 et 4.6 respectivement pour nombre de bits effectif équivalent. Par exemple, l'article stipule que le nombre de bits effectif équivalent pour un sur échantillonnage de 32 est de 6.9 bits et de 2.4 bits pour un sur échantillonnage d'un facteur 4.

## 2.4 Quantification et fréquence d'échantillonnage

Une méthode simple afin de diminuer la quantité de données à transmettre est de réduire la fréquence d'échantillonnage et le nombre de bits de quantification des signaux neuronaux. L'article présenté dans [83] démontre les conséquences de la diminution de ces deux paramètres sur la classification des formes d'ondes obtenues par l'algorithme d'analyse en composantes principales (PCA) suivi de l'algorithme de classification (*clustering*) k-means [84]. Il y est notamment démontré, par des tests comparatifs, qu'une diminution du nombre de bits de quantification jusqu'à 4 bits par échantillon n'affecte pratiquement pas les résultats de classification des AP. En deçà de 4 bits, les résultats de classification se dégradent rapidement. Il y est aussi démontré que réduire la fréquence d'échantillonnage diminue les performances de classification. Selon les auteurs, ce fait est attribuable à la phase des AP versus les temps d'échantillonnage qui résultent en des formes d'ondes légèrement différentes pour des AP provenant d'un même neurone. Une piste de solution à ce problème est proposée avec l'utilisation d'un filtre d'interpolation avant l'étape de classification.

## 2.5 Compression de type *Compressed Sensing* (CS)

### 2.5.1 Présentation de l'algorithme CS, aspects mathématiques

La compression de type *Compressed sensing* (CS) est basée sur deux concepts clés, la parcimonie d'un signal et l'échantillonnage incohérent. Ce type de compression est applicable pour un signal discret de taille  $N$  qui est dit parcimonieux lorsqu'il est exprimé dans une base  $\psi$  avec  $\psi \in \mathbb{R}^{N \times N}$ . C'est-à-dire qu'il possède un nombre limité de coefficients significatifs  $K$  sur cette base, ou  $K \ll N$ . Pour ainsi dire, plus un signal est parcimonieux dans une base (cas généralement de signaux biophysiques), plus il sera possible de le reconstruire avec fidélité après compression. Il faut noter qu'une transformation en ondelettes permet généralement d'exprimer un AP dans une base parcimonieuse. Il est possible d'exprimer un signal  $x$  en



fonction de la base parcimonieuse  $\psi$  selon l'équation suivante :

$$x = \psi\alpha, \text{ avec } \alpha \in \mathbb{R}^N \quad (2.14)$$

Où  $\alpha$  est un vecteur appelé *K-sparse*, c'est à dire ayant  $K$  coefficients non nuls. La compression du signal  $x$  est effectuée en sélectionnant  $M$  points de  $x$  sur les  $N$  points, où  $M \ll N$ . Cette opération revient à multiplier  $x$  par une matrice  $\phi$ , où  $\phi$  est une matrice de  $M$  lignes possédant un seul élément unitaire par ligne, ce qui produit un vecteur  $y$  de taille  $M \times 1$ . Cette opération est présentée à l'équation suivante :

$$y = \phi x = \phi\psi\alpha = \Theta\alpha \quad (2.15)$$

La compression de type CS génère un taux de compression de  $\frac{N}{M}$ . La reconstruction de  $x$  à partir de  $y$  est possible si  $M \geq K$  et si la propriété d'isométrie (RIP) est respectée et la cohérence entre les matrices  $\phi$  et  $\psi$  est minimale. Selon la théorie de la compression par CS, une matrice aléatoire de type Gaussienne ou une matrice de Bernoulli remplit ces conditions avec une grande probabilité. Retrouver  $\alpha$ , soit  $\psi^{-1}x$ , consiste à minimiser la norme  $l_1$  selon l'équation suivante :

$$\alpha_{approx} = \min \|\tilde{\alpha}\|_{l_1}, \text{ avec } y = \Theta\tilde{\alpha} \quad (2.16)$$

Ce qui est logique puisque  $\min \|\tilde{\alpha}\|_{l_1}$  consiste à trouver le vecteur  $\tilde{\alpha}$  le plus parcimonieux, soit une condition connue de notre signal de départ. Il suffit d'appliquer la transformée  $\psi\alpha_{approx}$  ( $x = \psi\psi^{-1}x \approx \psi\alpha_{approx}$ ) pour retrouver une approximation de  $x$  i.e. retrouver la version décompressée de  $x$ .

## 2.5.2 Compression par CS dans la littérature

Une implantation matérielle de l'algorithme CS appliquée au traitement de signaux biophysiques de type EEG est présentée dans [85]. La matrice de compression  $\phi$  utilisée est de type Bernoulli (entrée de  $\pm 1$ ). Puisque chaque élément de  $y$  est une combinaison linéaire des échantillons de l'ADC, chacun des  $y_i$  sont réalisés avec un accumulateur pouvant soit additionner, soit soustraire. Le choix de l'opération est réalisé avec une séquence pseudo aléatoire dont le *seed* est régénéré à tous les  $N$  échantillons. Deux circuits de type *pseudo random binary sequence* (PRBS) jumelé à des portes XOR permettent de générer les coefficients aléatoires pour chacune des  $M$  lignes de  $\phi$ .

La matrice  $\psi$  permettant d'obtenir une parcimonie optimale dépend du signal à traiter. [86] compare l'utilisation d'une matrice de transformation en ondelettes *Daubechie-8* de 6 niveaux versus deux matrices de parcimonie optimisées et issues d'un programme d'apprentissage [87] pour compresser des signaux neuronaux réels et simulés. L'évaluation de la qualité de la reconstruction des signaux compressés est effectuée selon le taux de succès de *clustering* des AP avant et après la compression en utilisant le programme *Osort*. Il y est démontré que la qualité de la reconstruction se dégrade significativement avec un SNR bas, ce qui est

le cas des signaux réels utilisés dans l'article. Avec un taux de compression de 2.05 et des données échantillonnées à 20 kHz, [86] obtient, dans le pire cas, un succès du taux de classement de 50% pour la matrice de transformation en ondelettes et de 40% avec les matrices de parcimonie optimisées. Suite à ces résultats, [86] tire la conclusion qu'ils ne peuvent pas recommander l'utilisation de l'algorithme CS pour compresser des AP en général.

[88] présente une méthode permettant de trouver une matrice de parcimonie optimale en utilisant l'algorithme K-SVD [89] (optimale pour compresser des AP) et compare les résultats avec ceux obtenus avec une matrice de parcimonie dans le domaine des ondelettes. L'algorithme K-SVD est un algorithme itératif nécessitant une période d'apprentissage afin de construire une matrice exprimant le signal original de façon parcimonieuse. L'article démontre qu'il est possible d'obtenir la même qualité de reconstruction qu'en utilisant une matrice parcimonieuse dans le domaine des ondelettes (*Daubechie-8*) en retenant 2.5 fois moins d'échantillons, soit un taux de compression 2.5 fois plus élevé.

De façon similaire, [90] présente une technique permettant de construire une base parcimonieuse de façon itérative en utilisant un algorithme *basis pursuit denoising* (BPDM) modifié. Au début, les AP sont décompressés en utilisant la parcimonie dans le domaine des ondelettes et chaque potentiel décompressé est ajouté dans une union servant à construire une base parcimonieuse qui est raffinée progressivement. Cette méthode permet d'éviter la période d'apprentissage nécessaire dans [88].

## 2.6 Méthodes basées sur la quantification vectorielle (QV)

### 2.6.1 Présentation de l'algorithme QV, aspects mathématiques

L'algorithme de quantification vectorielle (QV) consiste à diviser un espace à plusieurs dimensions en régions d'encodage (*coding region*). Par exemple, deux droites se croisant dans le plan cartésien séparent ce plan en quatre régions. À chacune des régions est associé un vecteur de code (*code vector*), représentant une approximation de tous les vecteurs contenus dans la région. L'objectif de l'algorithme QV est de trouver l'ensemble des vecteurs de code, ainsi que des partitions, qui génèrent une distorsion minimale une fois les signaux compressés. En connaissant à priori la région de l'espace où une majorité de vecteurs peuvent se trouver, il est possible de générer une partition optimisée permettant d'approximer les vecteurs avec leurs vecteurs de code respectif. Pour des signaux neuronaux, [91] propose d'utiliser la fonction de coût à minimiser suivante :

$$J(C) = \sum_{i=1}^N \omega_i \|x_i - c_i\|^2 \quad (2.17)$$

Où  $N$  est le nombre de signaux d'entraînements,  $x_i$  est un signal parmi les signaux d'entraînement,  $c_i$  est le signal approximé par la partition et  $\omega_i$  est un poids discriminatoire associé

aux vecteurs  $x_i$ . Cet algorithme diffère par le facteur  $\omega_i$  de l'algorithme *Linde-Buzo-Gray algorithm* (LBG), ceci afin d'obtenir une plus grande densité de vecteur de code où les AP résident dans l'espace.

## 2.6.2 Algorithme de compression par QV

L'algorithme de sélection des paramètres présenté dans [91] est le suivant :

- Choisir la dimension de la partition qui va permettre de séparer efficacement les AP.
- Utiliser un algorithme discriminatif pour identifier les  $\omega_i$  afin de distribuer les vecteurs de code sur la partition.
- Utiliser un code de Huffman (codage par entropie, voir section 2.7.2) pour représenter chacun des vecteurs de code avec un code binaire unique.

L'algorithme de compression et de reconstruction présenté dans [91] est le suivant :

- Pour chacune des formes d'ondes, trouvez le vecteur de code le plus près.
- Envoi de la représentation binaire du vecteur de code.
- Reconstruction du signal neuronal en utilisant le *codebook*, soit l'ensemble des vecteurs de code.

## 2.7 Encodage des coefficients (ondelettes ou autre)

### 2.7.1 Encodage SPITH

L'algorithme SPITH est utilisé dans [92], [78] et [72] afin de compresser les coefficients issus de la transformée en ondelettes de signaux ECG. La méthode est la suivante : premièrement, il faut identifier la position  $n$  du bit **MSB** parmi tous les coefficients. Ensuite, pour toutes les positions  $j$  de  $n$  jusqu'à 0, il faut transmettre de façon sérielle un bit à 1 si le **MSB** du coefficient est à la position  $j$  ou un bit à 0 si le **MSB** du coefficient est à une position plus petite que  $j$ . Pour les coefficients dont le **MSB** est à une position plus grande que  $j$ , transmettre le prochain bit non transmit. Cette technique possède l'avantage de communiquer l'information la plus significative en premier, permettant ainsi de reconstruire le signal au fur et à mesure que les bits sont reçus. Pour compresser, il suffit de stopper l'algorithme avec le niveau  $j$  désiré. En descendant  $j$  au niveau 0, l'encodage est sans perte.

### 2.7.2 Encodage de Huffman

Le code de Huffman est un encodage sans perte basé sur l'entropie [93]. Il s'agit d'associer à chacun des symboles (mots de  $x$  bits) un code unique en représentant les symboles les plus fréquents avec moins de bits. Un code de Huffman est créé par la construction d'un arbre binaire où chacun des symboles est associé à un nœud, dont sa position est déterminée par sa fréquence d'occurrence. Les symboles les moins fréquents sont associés aux feuilles les

plus basses de l'arbre. Pour associer un code à un symbole, l'arbre est parcouru à partir de sa racine en ajoutant 0 au code du parent si le nœud n'est pas une feuille et 1 s'il en est une. Par exemple, le symbole le plus fréquent se voit attribuer le code 1, le second plus fréquent le code 01 et troisième plus fréquent le code 001 et ainsi de suite. Il existe trois variantes de l'algorithme, statique, semi-adaptatif et adaptatif. Dans les deux premières versions, le code est construit avec des connaissances à priori sur le signal à encoder. Dans la version adaptative, l'arbre est construit au fur et à mesure que les signaux sont reçus.

## Chapitre 3

# Travaux préliminaires de conception et de simulation

Certains algorithmes de compressions présentés au chapitre 2 ont été retenus afin d'être implémentés, simulés et comparés dans *Matlab* dans le but d'identifier les algorithmes les plus prometteurs. Notamment, cette section présente des résultats relatifs à la compression par ondelettes de type *Haar* et de type *Symmlet-2* (voir section 2.2), à la compression par encodage *Delta* (voir section 2.3), à la compression par diminution du nombre de bits de quantification (voir section 2.4) et à la compression de type CS (voir section 2.5) avec base parcimonieuse dans le domaine des ondelettes et avec base parcimonieuse optimale. L'algorithme de compression QV (voir section 2.6) n'a pas été retenu à cause de sa grande complexité qui limite son utilisation/implémentation dans un système embarqué.

Les algorithmes ont été implémentés de façon à compresser les AP seulement, car ces derniers sont considérés comme détectés a priori par un algorithme de détection. Il est entendu qu'un taux de compression (CR) beaucoup plus important est atteint en utilisant de la détection d'AP et en compressant les formes d'onde des AP détectés.

### 3.1 Méthodologie

Les résultats présentés dans ce chapitre ont été obtenus avec des données neuronales quantifiées sur 16 bits et échantillonnées à 24 kHz issues du fichier *C\_Easy1\_noise01.mat* tirées de la banque de signaux neuronaux en ligne de l'Université Leicester [12]. À partir de ces données importées dans *Matlab*, 3237 AP ont été détectés avec l'opérateur MAD (voir l'équation 2.1) et sont utilisés pour comparer les différents algorithmes de compression. Afin d'alléger la lecture de ce chapitre, les résultats sont présentés sous forme graphique à la section 3.6, tandis que les résultats détaillés sont présentés dans des tableaux à l'annexe A pour consultation.

### 3.1.1 Tests reliés au SNDR

Plusieurs méthodes de comparaison sont utilisées pour quantifier la qualité de compression des algorithmes. Parmi ces méthodes, le *signal to noise ratio-plus-distortion ratio* (SNDR) minimum, maximum, moyen ainsi que sa déviation moyenne sont utilisés. Le SNDR est une métrique utilisée dans [90] et [88] afin de quantifier la distorsion induite par la compression de signaux neuronaux et elle est définie par l'équation suivante :

$$SNDR = 20 \log \left[ \frac{|x|}{|\hat{x} - x|} \right], \quad (3.1)$$

Où  $x$  est un vecteur contenant les 48 échantillons de la forme d'onde originale et  $\hat{x}$  est le vecteur de la forme d'onde une fois décompressée. Un signal possédant un haut SNDR indique une reconstruction de meilleure qualité. Les SNDR maximum et minimum sont utilisés afin d'avoir un aperçu de la meilleure et de la pire reconstruction parmi les 3237 AP compressés, alors que la déviation moyenne est un indicatif sur la variabilité de la qualité de la compression/reconstruction. Une déviation moyenne faible du SNDR est souhaitée afin que les AP soient compressés/décompressés avec une qualité similaire, peu importe la forme des AP.

### 3.1.2 Tests de *clustering*

#### Comparaison du nombre de *clusters*

Une autre méthode de comparaison consiste à comparer le nombre de *clusters* issus des signaux avant et après compression. La méthode proposée consiste à identifier le nombre de *clusters* en effectuant une PCA suivie de l'algorithme de *clustering* nommé *mean shift clustering* [94]. L'algorithme *mean shift clustering* a été choisi puisqu'il détermine automatiquement le nombre de *clusters* sans intervention externe. Cette caractéristique est utile lorsque le nombre de *clusters* est inconnu au départ, ce qui est le cas des signaux compressés qui peuvent générer un nombre indéterminé de *clusters* selon la qualité de la compression. Trois itérations de *clustering* sont effectuées, la première étant sur les 3237 AP originaux, servant de base de comparaison et dont la PCA et les associations AP à neurones sont présentées aux figure 3.1a et 3.1b respectivement (on peut y voir clairement quatre *clusters* distincts). La seconde itération effectue un *clustering* sur les 3237 AP compressés par chacun des algorithmes de compression. La dernière itération effectue un *clustering* sur les 3237 AP compressés par chacun des algorithmes joints aux 3237 AP originaux. Ces deux dernières étapes de *clustering* permettent de vérifier si la compression a mené à des divergences dans la quantité de *clusters* trouvée par l'algorithme *mean shift clustering*.

#### Association au bon *cluster*

Une autre méthode de comparaison consiste à vérifier si un AP compressé est associé au même *cluster* que sa version originale. Pour ce faire, deux tests ont été effectués.

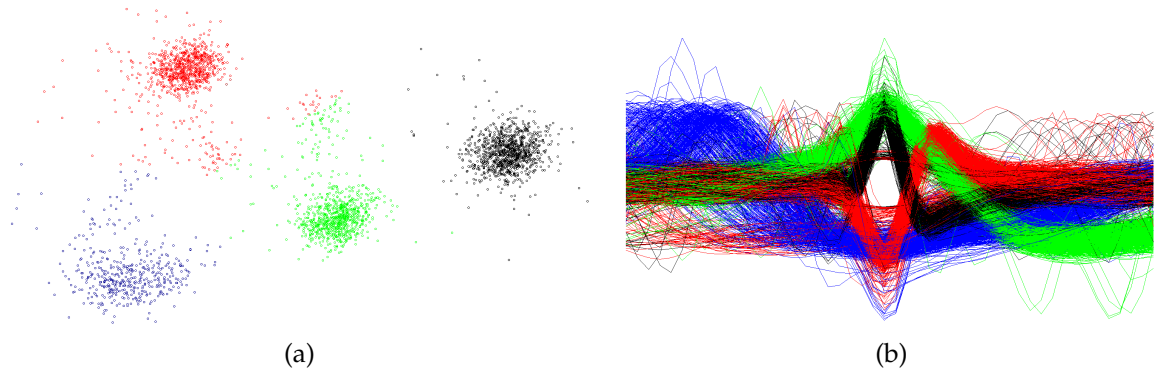


FIGURE 3.1 – a) *Clustering* automatique des 3237 AP originaux traités avec une PCA suivie de l'algorithme *mean shift clustering*, b) affichage simultané des 3237 AP détectés, ou chaque couleur correspond à une association AP à neurone.

- Le premier test se décrit comme suit<sup>1</sup> : pour chacun des AP compressés, une décomposition PCA est effectuée sur les données originales en leur ajoutant un AP compressé à la fois, suivi de l'algorithme *mean shift clustering*. Une vérification est ensuite faite afin de déterminer si l'AP compressé se trouve dans le même *cluster* que sa version originale. Avec ce test, il est possible de déterminer la quantité d'AP compressés ayant conduit à un mauvais *clustering* ou à la création d'un nouveau *cluster* non désiré.
- Le second test se décrit comme suit<sup>2</sup> : une PCA est effectuée sur les signaux compressés et originaux joints dans le même ensemble. Ensuite, il y a une comparaison entre chacun des AP compressés et sa version originale afin de vérifier s'ils se trouvent dans le même *cluster*. Encore une fois, cette méthode permet de donner un indice sur la qualité de la classification des AP après compression.

## 3.2 Compression par transformée en ondelettes

La compression par ondelettes se fait généralement en retirant les coefficients de moindre importance issus de la DWT. Selon le nombre de bits de quantification utilisé pour encoder les coefficients conservés, un taux de compression plus élevé peut être atteint. Il est à noter qu'un même CR peut être obtenu de manière différente en variant sur ces deux paramètres. Afin d'optimiser la plage dynamique de la quantification, une normalisation est effectuée sur les coefficients. De cette façon, chaque bit est utilisé à son plein potentiel, avec comme inconvénient de devoir transmettre la valeur maximale parmi tous les coefficients. Dans les tests, la valeur maximale est considérée sur 8 bits. De plus, il faut tenir compte de la transmission du *mappage* des coefficients qui ont été retenus. Les tests ont été effectués sur des AP de 48 points, ce qui génère un *mappage* de 48 bits. En tenant compte de ces paramètres, le CR

1. *Regroupement unitaire* dans les tableaux en annexe A  
 2. *Regroupement joint* dans les tableaux en annexe A

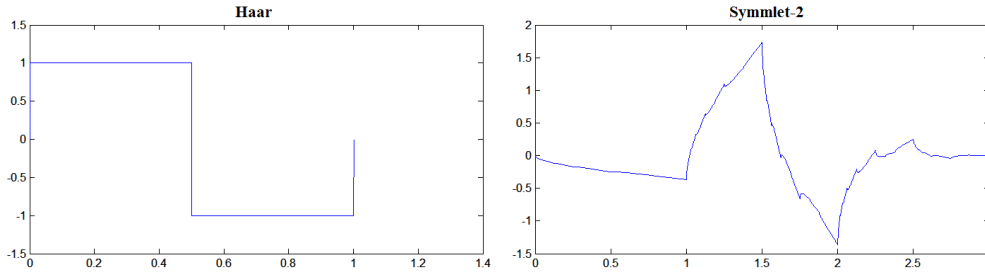


FIGURE 3.2 – Formes des fonctions  $\psi$  pour les ondelettes de type *Haar* et *Daubechie-2*

est calculé comme suit :

$$CR = \frac{16 \text{ bits} \times 48 \text{ echantillons}}{nbBitsQuantification \times nbCoefsConservees + 48 \text{ bits} + 8 \text{ bits}} \quad (3.2)$$

Deux types d'ondelettes ont été retenus pour être testés, soit l'ondelette de type *Haar* et l'ondelette de type *Symmlet-2*. L'ondelette de type *Haar* a été retenue puisqu'elle est l'ondelette la plus simple à réaliser i.e. ses filtres  $h(n)$  et  $g(n)$  possèdent seulement deux coefficients. L'ondelette de type *Symmlet-2* a été retenue pour deux raisons. Premièrement, il a été démontré que la famille d'ondelettes de type *Symmlet* peut générer une compression quasi optimale des signaux neuronaux [75] et deuxièmement, puisque sa fonction  $\psi$  possède une forme similaire à un AP. Les formes des fonctions  $\psi$  pour les ondelettes de type *Haar* et *Symmlet-2* sont présentées à la figure 3.2.

### 3.2.1 *Haar*

Les résultats détaillés reliés au SNDR minimum, maximum, moyen et à la déviation moyenne sont présentés en annexe A dans le tableau A.1. Les résultats permettant de comparer le nombre de *clusters* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.2. Les résultats permettant de comparer les associations des AP au bon *cluster* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.3.

### 3.2.2 *Symmlet-2*

Les résultats détaillés reliés au SNDR minimum, maximum, moyen et à la déviation moyenne sont présentés en annexe A dans le tableau A.4. Les résultats permettant de comparer le nombre de *clusters* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.5. Les résultats permettant de comparer les associations des AP au bon *cluster* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.6.



### 3.3 Encodage *Delta*

La compression *Delta* est effectuée en diminuant le nombre de bits de quantification de la différence entre deux échantillons successifs. Afin de minimiser l'effet de divergence pouvant apparaître lors de la reconstruction (somme progressive des erreurs de quantification), une modification a été apportée à l'algorithme original afin d'éliminer cette erreur. Au lieu de comparer l'échantillon suivant à l'échantillon précédent, l'échantillon suivant est comparé au signal reconstruit progressivement. Ainsi, la différence quantifiée est la différence réelle entre l'échantillon précédent du signal reconstruit versus l'échantillon suivant, éliminant l'erreur progressive de reconstruction. Afin d'optimiser la plage dynamique de la quantification, une normalisation a été effectuée sur chacune des différences. De cette façon, chaque bit est utilisé à son plein potentiel, avec comme inconvénient de devoir transmettre l'amplitude maximale des coefficients, considérée sur 8 bits dans les tests. Le CR de l'algorithme *Delta* est calculé comme suit :

$$CR = \frac{16 \text{ bits} \times 48 \text{ échantillons}}{nbBitsQuantification \times 48 \text{ différences} + 8 \text{ bits}} \quad (3.3)$$

Les résultats détaillés reliés au SNDR minimum, maximum, moyen et à la déviation moyenne sont présentés en annexe A dans le tableau A.7. Les résultats permettant de comparer le nombre de *clusters* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.8. Les résultats permettant de comparer les associations des AP au bon *cluster* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.9.

### 3.4 Diminuer le nombre de bits de quantification (DNBQ)

Cet algorithme est simple à réaliser. Il s'agit de diminuer le nombre de bits de quantification des échantillons. Le CR de l'algorithme est calculé comme suit :

$$CR = \frac{16 \text{ bits}}{nbBitsQuantification} \quad (3.4)$$

Les résultats détaillés reliés au SNDR minimum, maximum, moyen et à la déviation moyenne sont présentés en annexe A dans le tableau A.10. Les résultats permettant de comparer le nombre de *clusters* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.11. Les résultats permettant de comparer les associations des AP au bon *cluster* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.12.

### 3.5 Compression de type *Compressed Sensing*

La compression de type CS est effectuée en sélectionnant aléatoirement les échantillons à conserver (vecteur  $y$  dans l'équation 2.15) et en quantifiant ces derniers sur 8 bits. Avant la

quantification, une normalisation des coefficients est effectuée afin de maximiser leur plage binaire, au prix de devoir transmettre l’amplitude du coefficient maximal en pleine résolution sur 16 bits afin de pouvoir effectuer une normalisation inverse avant la reconstruction.

### 3.5.1 Parcimonie dans le domaine des ondelettes (CSWD)

Dans cette section, les tests ont été effectués en considérant une parcimonie dans le domaine des ondelettes de type *Daubechie-8* [86], [88]. Il est à noter que l’algorithme fait intervenir une matrice aléatoire pouvant influencer les résultats. La matrice aléatoire utilisée est de type *Bernoulli* possédant des entrées de  $\pm 1$  [85]. Le CR de l’algorithme CS avec base parcimonieuse dans le domaine des ondelettes est calculé comme suit :

$$CR = \frac{16 \text{ bits} \times 48 \text{ echantillons}}{nbPointsConservees \times 8 \text{ bits} + 16 \text{ bits}} \quad (3.5)$$

Les résultats détaillés reliés au SNDR minimum, maximum, moyen et à la déviation moyenne sont présentés en annexe A dans le tableau A.13. Les résultats permettant de comparer le nombre de *clusters* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.14. Les résultats permettant de comparer les associations des AP au bon *cluster* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.15.

### 3.5.2 Base parcimonieuse optimale (CSOP)

Dans cette section, les tests ont été effectués en trouvant une matrice de parcimonie optimale en utilisant l’algorithme K-SVD [88], [89]. L’entraînement de l’algorithme K-SVD est effectué sur 5% des AP, soit sur 161 formes d’ondes. Il est à noter que l’algorithme fait intervenir une matrice aléatoire pouvant influencer les résultats, la matrice aléatoire utilisée est de type *Bernoulli* (entrées de  $\pm 1$ ). Le CR de l’algorithme CS avec base parcimonieuse optimale est calculé comme suit :

$$CR = \frac{16 \text{ bits} \cdot 48 \text{ echantillons}}{nbPointsConservees \cdot 8 \text{ bits} + 16 \text{ bits}} \quad (3.6)$$

Les résultats détaillés reliés au SNDR minimum, maximum, moyen et à la déviation moyenne sont présentés en annexe A dans le tableau A.16. Les résultats permettant de comparer le nombre de *clusters* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.17. Les résultats permettant de comparer les associations des AP au bon *cluster* (test décrit à la section 3.1.2) sont présentés en annexe A dans le tableau A.18.

## 3.6 Synthèse des résultats

Cette section présente une analyse des résultats obtenus à la section précédente. Les résultats sont exprimés sous forme graphique, afin d’en faciliter l’interprétation et sont présentés

pour les CR suivants : 2, 3, 4, 5, 6, 7 et 8. Il est à noter que certains de ces taux ne sont pas réalisables par tous les algorithmes, par exemple, l'algorithme de compression *Delta* ne peut pas produire les CR de 6 et 8.

### 3.6.1 SNDR

En observant les figures 3.3, 3.4, 3.5 et 3.6, on constate que tous les algorithmes subissent une réduction des SNDR moyens, minimums et maximums ainsi qu'une augmentation de la déviation moyenne du SNDR lorsque le CR augmente.

C'est l'algorithme de compressions de type CS avec matrice de parcimonie optimale qui offre le meilleur SNDR moyen pour presque tous les CR, suivi de l'algorithme à base de l'ondelette *Symmlet-2*. Les résultats sont similaires pour le SNDR maximum, à la différence que l'algorithme de compressions de type *Delta* a généré les meilleurs résultats pour un  $CR \leq 3$  et qu'il se classe deuxième pour les CR supérieurs. Les bons résultats de CS avec base parcimonieuse optimale étaient prévisibles dans ces deux tests, puisque les AP ayant une forme similaire aux AP ayant servi à construire la matrice de parcimonie optimale ont bénéficié d'une reconstruction presque parfaite.

Les résultats sont complètement différents pour le SNDR minimum, dans lequel l'algorithme de compression à base de l'ondelette *Symmlet-2* a généré les meilleurs résultats pour tous les CR, suivi de l'algorithme de compression à base de l'ondelette *Haar*. Les deux algorithmes de type CS ont généré des SNDR négatifs pour des  $CR \geq 4$ , ce qui est très mauvais, alors que l'algorithme à faible quantification oscille autour de 0 dB pour les  $CR \geq 5$ . Il est à noter que l'algorithme CS avec matrice de parcimonie optimale reproduit très mal les AP qui sont légèrement différents de ceux qui ont été utilisés pour construire la matrice de parcimonie. Cette caractéristique est un désavantage majeur, puisque dans des conditions réelles, des AP ayant des formes différentes peuvent se produire à tout moment.

Tel que stipulé, certains algorithmes ont de la difficulté à reproduire fidèlement toutes les formes d'AP. C'est le cas des deux algorithmes de type CS, de l'algorithme *Delta* et de l'algorithme à faible quantification qui ont tous générés des déviations moyennes du SNDR pouvant dépasser les 30 % dans certains cas, alors que les deux algorithmes de compression par ondelettes n'ont jamais dépassé les 10.5 % pour tous les types d'AP.

### 3.6.2 Classification des AP et déviation moyenne

Pour les deux tests de classification effectués (figures 3.7 et 3.8), les algorithmes de compression à base de l'ondelette *Symmlet-2* et *Haar* permettent d'obtenir les meilleurs résultats pour presque tous les CR. Les performances de l'algorithme *Delta* et à faible quantification se dégradent progressivement dans les tests de la figure 3.7 et sont stables dans ceux de la figure 3.8. Les résultats de compression de l'algorithme CS avec parcimonie dans le domaine des

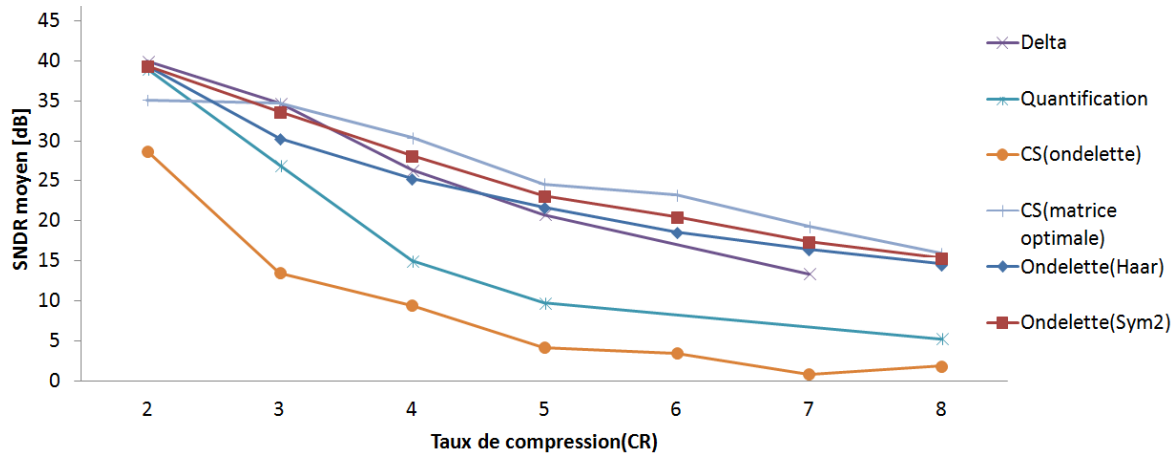


FIGURE 3.3 – SNDR moyen obtenu avec tous les algorithmes testés pour un CR de 2 à 8.

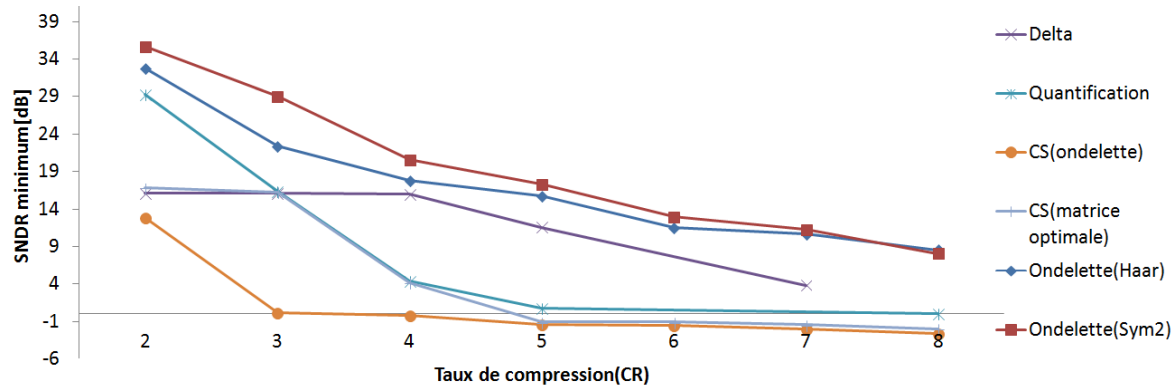


FIGURE 3.4 – SNDR minimum obtenu avec tous les algorithmes testés pour un CR de 2 à 8.

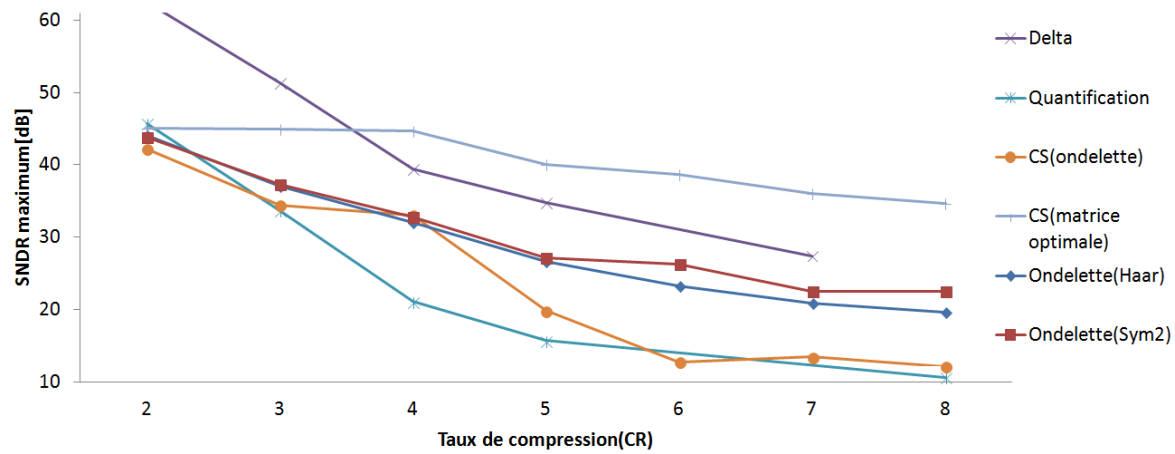


FIGURE 3.5 – SNDR maximum obtenu avec tous les algorithmes testés pour un CR de 2 à 8.

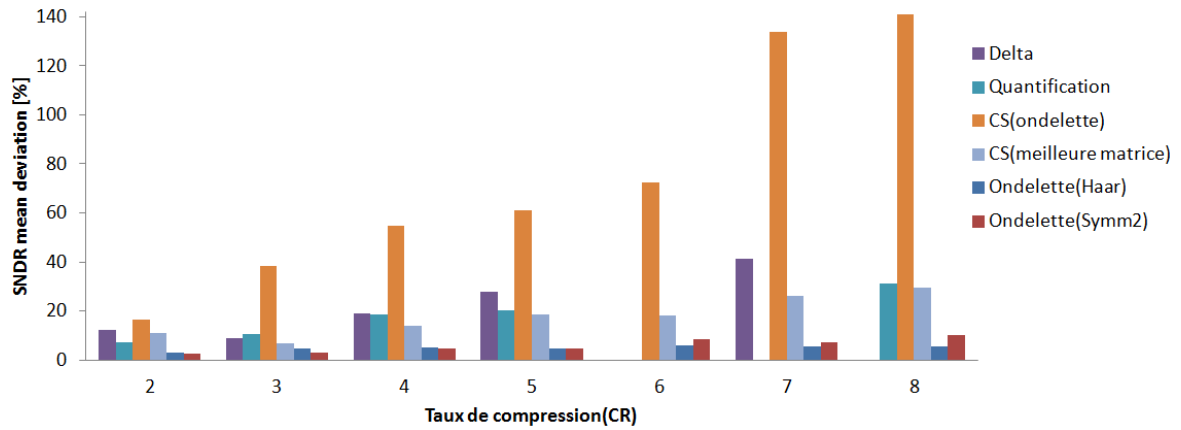


FIGURE 3.6 – Déviation moyenne du SNDR obtenu avec tous les algorithmes testés pour un CR de 2 à 8.

ondelettes subissent une grande variation dans les deux tests, idem pour ceux avec la matrice de parcimonie optimale. La grande variabilité des résultats obtenus par l’algorithme CS, avec les deux bases parcimonieuses, peut être expliquée par le fait que certains AP ne sont pas bien représentés par la base parcimonieuse utilisée, car certains AP sont reconstruits avec fidélité alors que d’autres ne le sont pas. Ces résultats laissent supposer que l’algorithme de compression CS n’est pas un algorithme de compression approprié pour compresser les AP et pour ensuite faire du *clustering*, ce qui concorde avec les conclusions présentées dans [86].

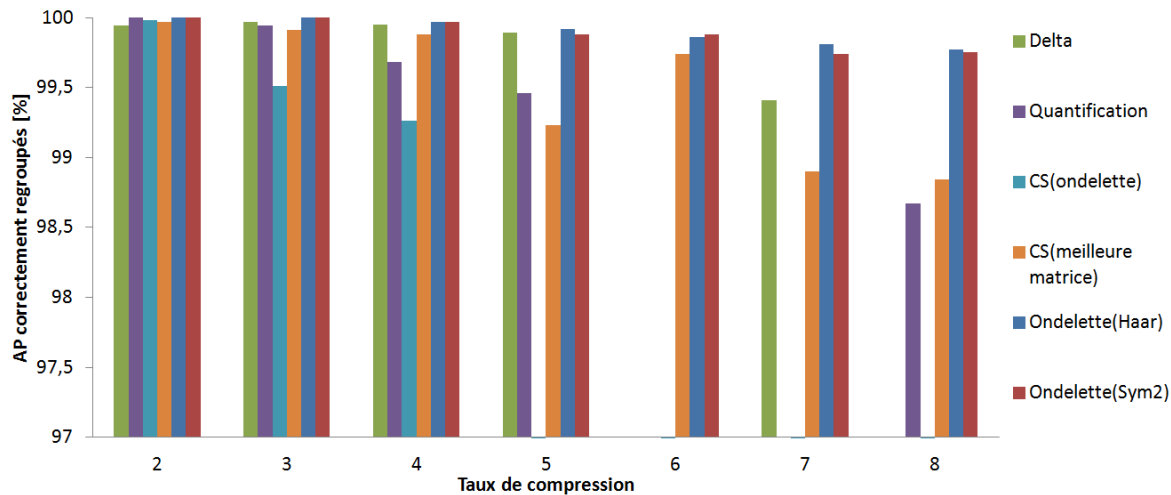


FIGURE 3.7 – Pourcentage des AP compressés dans le bon *cluster* avec PCA effectuée sur les AP compressés joint aux AP originaux.

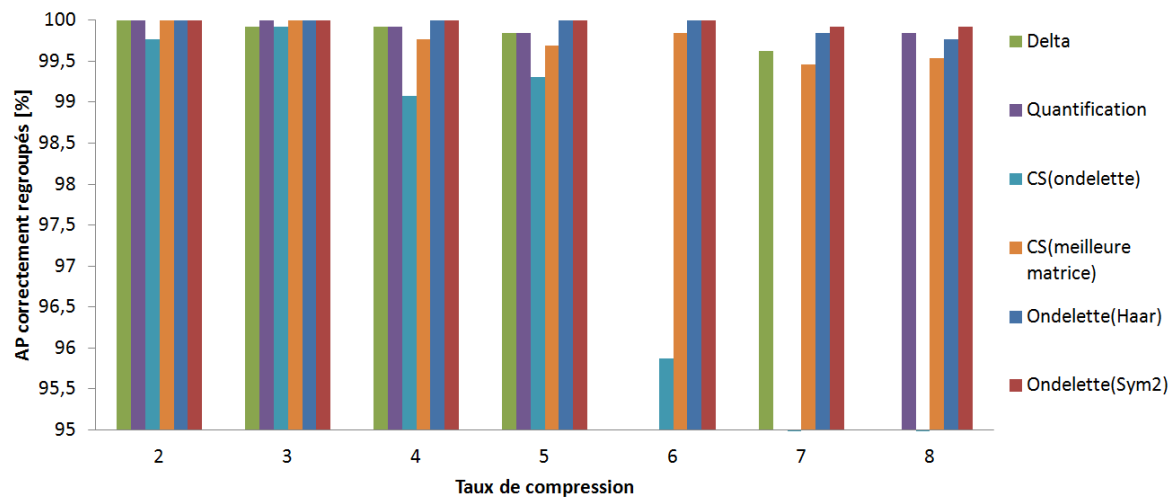


FIGURE 3.8 – Pourcentage des AP compressés dans le bon *cluster* avec PCA effectuée en ajoutant les AP compressés un à la fois aux AP originaux.

### 3.6.3 Variation du nombre de *clusters* détectés

Tous les algorithmes, à l'exception de CS, ont permis de détecter le bon nombre de *clusters* pour tous les CR. Lors des tests de *clustering* sur les AP compressés seulement (figure 3.9), l'algorithme CS avec parcimonie dans le domaine des ondelettes a généré des variations dans le nombre de *clusters* détectés pour un  $CR \geq 5$ , alors que l'algorithme CS avec matrice de parcimonie optimale présente des variations pour un  $CR \geq 6$ , allant même jusqu'à produire un *cluster* supplémentaire pour un CR de 8. Les résultats sont similaires pour les tests de *clustering* avec les AP compressés joints aux AP originaux (figure 3.10), à l'exception que des variations se produisent pour des  $CR \geq 7$ .

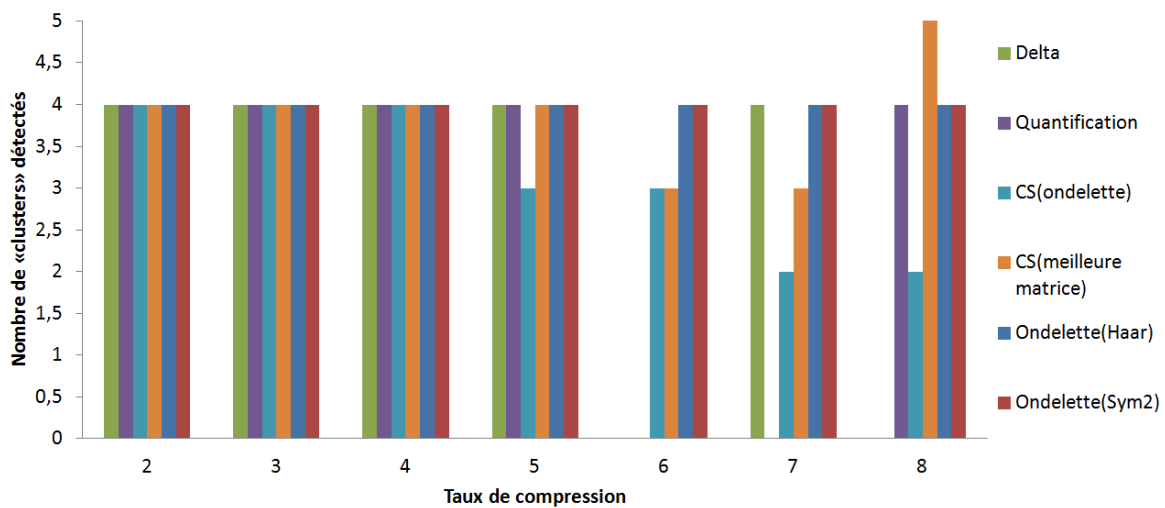


FIGURE 3.9 – Nombre de *clusters* trouvés après compression.

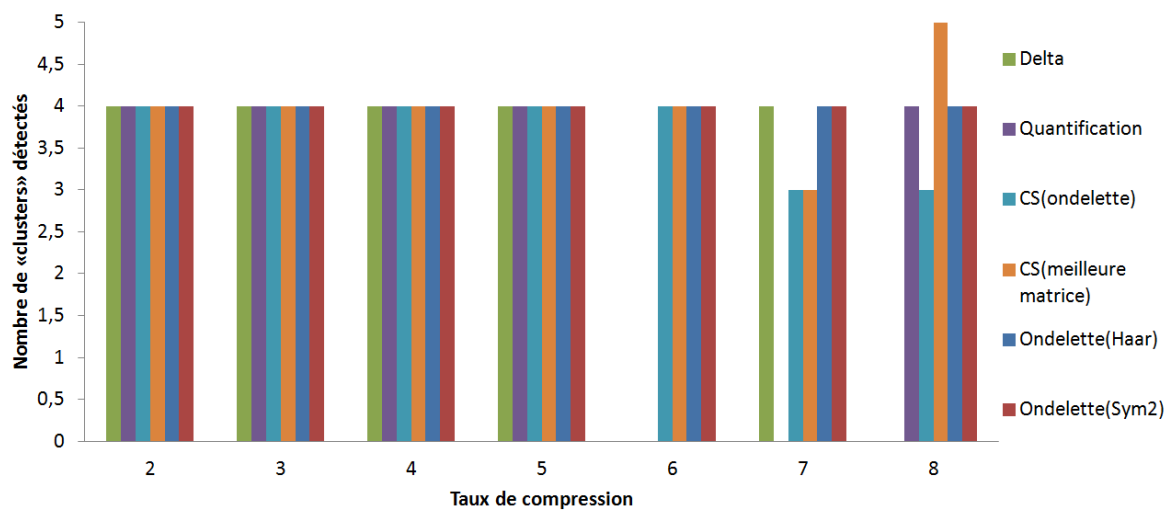


FIGURE 3.10 – Nombre de *clusters* trouvés avec les AP compressés joints aux AP originaux.

### 3.6.4 Fonction de coûts

Afin de déterminer l'algorithme de compression optimal, une fonction de coût a été créée. Premièrement, un score est calculé pour chacun des taux de compression. Ce score est pondéré à 50% en fonction des tests de SNDR et à 50% en fonction des résultats de *clustering*. Le score est construit en fonction des critères suivants :

#### 1. SNDR :

Les critères du SNDR moyen, minimum, maximum et la déviation moyenne du SNDR se voient attribuer 12.5% du score chacun. Les fonctions  $SNDR_{moy}/SNDR_{moy\_max}$ ,  $SNDR_{min}/SNDR_{min\_max}$  et  $SNDR_{max}/SNDR_{max\_max}$  sont conçus afin de donner une pondération de 1 aux algorithmes générant les meilleurs résultats alors que la fonction  $(1 - SNDR_{dev}/SNDR_{dev\_max})$  est conçue afin de donner une pondération de 0 à l'algorithme générant la plus grande la déviation moyenne. Dans ces formules, les termes  $SNDR_{xxx\_max}$  sont les valeurs maximales du critère  $SNDR_{xxx}$  obtenues parmi tous les algorithmes pour un même CR. Si la valeur de  $SNDR_{xxx}$  est négative, le critère correspondant se voit accorder une pondération de 0.

#### 2. Résultats de *clustering* :

Le ratio d'AP compressés et associés dans le bon *cluster* se voit attribuer 33.3% du score à partir de la fonction  $e^{-10 \times (1 - CL_{comp})} + e^{-10 \times (1 - CL_{joint})}$ , où  $CL_{comp}$  est le ratio des AP compressés dans le bon *cluster* (test de la figure 3.7),  $CL_{joint}^i$  est le ratio des AP compressés dans le bon *cluster* en les ajoutant un à la fois aux potentiels originaux (test de la figure 3.8). Ces fonctions sont exponentielles avec un facteur multiplicatif de 10 afin de pénaliser sévèrement les algorithmes ayant mené à de mauvaises associations et afin de donner une note de 1 lorsque toutes les associations sont bonnes.

Le critère du nombre de *clusters* trouvés se voit attribuer 16.7% du score à partir de la fonction  $(4 - |4 - NBCL_{comp}|)/8 + (4 - |4 - NBCL_{joint}|)/8$ , où  $NBCL_{comp}$  est le nombre de *clusters* trouvés après compression (test de la figure 3.9) et  $NBCL_{joint}$  est le nombre de *clusters* trouvés avec les AP compressés joints aux originaux (test de la figure 3.10). Cette fonction génère une pondération de 1 pour les algorithmes ayant trouvé 4 *clusters* dans les deux tests en plus de pénaliser de 0.125 point pour chaque *cluster* manquant ou supplémentaire trouvé.

Le score est défini par l'équation suivante :

$$\begin{aligned}
 Score_{CR} = & \left( SNDR_{min}/SNDR_{min\_max} + SNDR_{moy}/SNDR_{moy\_max} \right. \\
 & + SNDR_{max}/SNDR_{max\_max} + (1 - SNDR_{dev}/SNDR_{dev\_max}) + e^{-10 \times (1 - CL_{comp})} \\
 & \left. + e^{-10 \times (1 - CL_{joint})} + (4 - |4 - NBCL_{comp}|)/8 + (4 - |4 - NBCL_{joint}|)/8 \right) / 6
 \end{aligned} \quad (3.7)$$



Le tableau 3.1 présente les scores calculés pour tous les algorithmes. Puisque les algorithmes *Delta* et à faible quantification ne peuvent pas produire tous les CR, les valeurs impossibles à réaliser sont présentées en rouge dans le tableau et sont les mêmes que ceux obtenus avec le CR le plus petit et le plus proche réalisable.

La fonction de coût est calculée avec l'équation 3.8 à partir des scores du tableau 3.1 et donne une pondération plus grande aux CR plus élevés. Les résultats de la fonction de coût sont présentés dans le tableau 3.2.

$$\text{Coût} = (0.1 \times \text{Score}_2 + 0.2 \times \text{Score}_3 + 0.3 \times \text{Score}_4 + 0.4 \times \text{Score}_5 + 0.5 \times \text{Score}_6 + 0.6 \times \text{Score}_7 + 0.7 \times \text{Score}_8) / 2.8 \quad (3.8)$$

TABLE 3.1 – Score calculé pour chacun des algorithmes de compression en fonction d'un CR de 2 à 8.

CR	<i>Haar</i>	<i>Symmlet-2</i>	<i>Delta</i>	DNBQ	CSWD	CSOP
2	0.930	0.942	0.902	0.921	0.624	0.841
3	0.819	0.948	0.924	0.831	0.506	0.903
4	0.897	0.941	0.920	0.694	0.494	0.863
5	0.906	0.933	0.892	0.617	0.328	0.810
6	0.873	0.922	0.892	0.617	0.274	0.793
7	0.877	0.911	0.776	0.617	0.150	0.753
8	0.895	0.914	0.776	0.488	0.162	0.732

TABLE 3.2 – Fonction de coût calculée pour chacun des algorithmes de compression.

	<i>Haar</i>	<i>Symmlet-2</i>	<i>Delta</i>	DNBQ	CSWD	CSOP
<i>Coût</i>	0.890	0.924	0.844	0.619	0.280	0.789

### 3.6.5 Conclusion des simulations

Selon les résultats obtenus avec la fonction de coût, il est possible de conclure que l'algorithme de compression à base d'ondelettes de type *Symmlet-2* est le meilleur algorithme identifié afin de compresser les AP, suivi de l'algorithme à base d'ondelette de type *Haar*. De façon générale, l'algorithme *Delta* et à faible quantification ont généré des résultats moyens, presque toujours en deçà de ceux obtenus avec la transformée en ondelettes de type *Symmlet-2* et/ou de type *Haar*. Bien que l'algorithme de compression CS avec une base parcimonieuse optimale ait généré de bons résultats dans certaines circonstances, il s'agit d'un algorithme extrêmement gourmand en calcul et qui génère une grande variabilité des résultats i.e. on ne

peut pas prédire avec certitude la qualité de la compression pour tous les *AP*. C'est la même chose pour l'algorithme *CS* avec base parcimonieuse dans le domaine des ondelettes qui a généré les pires résultats dans presque tous les tests.

## Chapitre 4

# Systeme optogénétique sans fil avec détection en temps réel des AP

### 4.1 Présentation du projet

Ce chapitre présente la conception d'un *headstage* optogénétique miniature sans fil doté de deux canaux d'enregistrement, de deux canaux de stimulation optique et possédant la fonctionnalité de détection des AP en temps réel. Au moment de la rédaction de ce mémoire, il s'agit du premier *headstage* à offrir simultanément de la stimulation optique, de l'enregistrement de signaux électrophysiologiques ainsi que de la détection *in situ* d'AP. Le dispositif, dont les requis de conception sont présentés dans le tableau 4.1, a été réalisé avec la collaboration financière de l'entreprise *Doric Lenses, Canada*.

Le *headstage* décrit dans ce chapitre est conçu pour effectuer des expériences optogénétiques *in vivo* de longues durées sur des rongeurs transgéniques évoluant en toute liberté, tel que montré conceptuellement à la figure 4.1. Il peut transmettre les données neuronales brutes, ou bien optimiser la consommation énergétique en détectant et en transmettant seulement les formes d'ondes des AP détectés (méthode de compression présentée à la section 2.1). Le système innove en implémentant une nouvelle technique de seuillage adaptatif permettant une détection des AP robuste au bruit et fiable, tout en utilisant peu de ressources matérielles.

Le *headstage* est construit entièrement de composantes COTS et sa conception miniature, basée sur un PCB flexible rigide, résulte en un système léger et très compact. Le système a été validé *in vitro* (avec des signaux neuronaux réels et synthétiques) et *in vivo* avec une souris transgénique exprimant le gène Channelrhodopsin-2. Les expériences *in vivo* ont été réalisées au CRIUSMQ et les résultats obtenus, présentés à la fin du chapitre, témoignent de la grande qualité du système. Il est à noter que le *headstage* ainsi que les algorithmes et résultats présentés dans ce chapitre ont fait l'objet de plusieurs publications scientifiques [95],

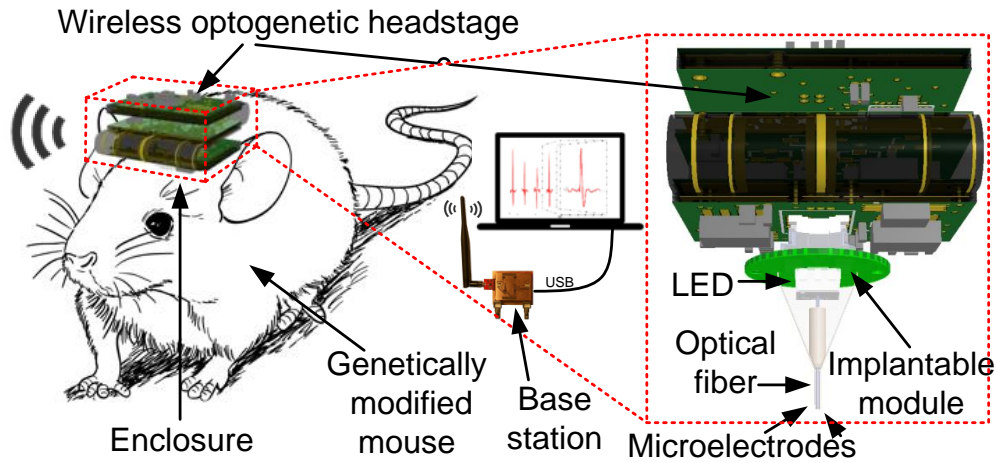


FIGURE 4.1 – Concept de la plateforme optogénétique sans fil à deux canaux.

[96], [97], [98], [99] et [100].

Dans un premier temps, les conceptions matérielle et logicielle du *headstage* sont présentées. Par la suite, des tests de performance comparant plusieurs algorithmes de détection implémentés et testés dans le *headstage* sont présentés et permettent d’isoler l’algorithme de détection des AP optimal. Ensuite, les aspects mathématiques ainsi que l’implémentation optimisée d’un nouvel algorithme de seuillage adaptatif sont présentés. Finalement, les caractéristiques physiques du *headstage* sont détaillées suivies des résultats *in vitro* et *in vivo*.

TABLE 4.1 – Requis de conception du *headstage* optogénétique à deux canaux exigés par le partenaire industriel *Doric Lenses*.

Requis	Valeur
# de canaux d’enregistrement	2
# de canaux de stimulation	2
Fréquence de quantification	$\geq 20$ kHz
Amplitude minimale des AP discernables	$40 \mu V$
Courant de stimulation	$\geq 150$ mA par DEL
Fréquence de stimulation	1 Hz à 100 Hz
<i>Duty cycle</i> de la stimulation	0.1 % à 10 %
Autonomie	$\geq 10$ min
Taille	$\leq 20 \times 20 \times 20$ mm <sup>3</sup>
Poids	$\leq 7$ g

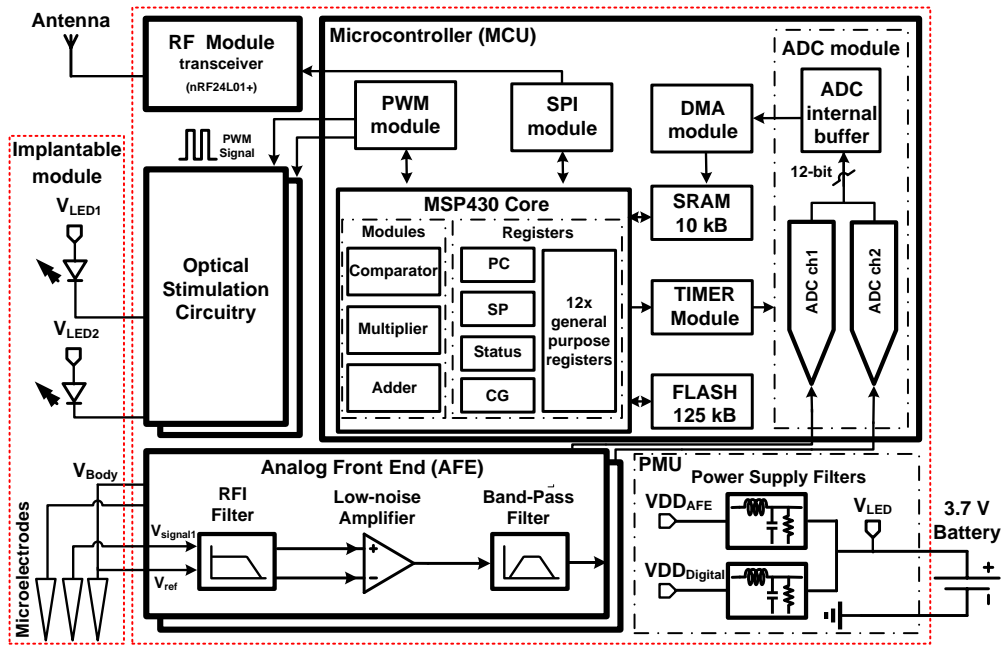


FIGURE 4.2 – Diagramme en bloc du *headstage* optogénétique à deux canaux.

## 4.2 Conception du *headstage*

Le système optogénétique est composé de deux parties reliées entre-elles par un connecteur. La première partie est le *headstage* lui-même, qui contient l'électronique pour traiter les données et contrôler le système, l'électronique de stimulation optique, l'électronique d'enregistrement et de conditionnement des signaux neuronaux ainsi que le système de transmission sans fil. La seconde partie est le module implantable, qui contient les DEL hautes puissances, les fibres optiques ainsi que les microélectrodes. Le module implantable est prévu pour être installé de façon permanente sur le crâne de l'animal, de façon à être réutilisé dans plusieurs expériences sur le même sujet. Le *headstage* peut-être déconnecté du module implantable entre les expériences, et ce, afin de recharger la batterie et/ou pour retirer du poids de sur la tête de l'animal.

La figure 4.2 montre le diagramme en bloc du *headstage*. Ce dernier est constitué de cinq parties distinctes : 1) un circuit d'amplification analogique (AFE), 2) un circuit de stimulation optique, 3) une unité de gestion d'énergie (PMU), 4) un transmetteur sans fil et 5) un microcontrôleur (MCU). La tâche de l'AFE consiste à amplifier et à filtrer avec précision les signaux neuronaux de faible amplitude présents aux bornes des microélectrodes tout en ajoutant un minimum de bruit électronique. Le rôle du circuit de stimulation optique consiste à appliquer un patron de tension déterminé sur les DEL. Cette tension, qui est traduite directement en courant dans les DEL, doit être précise afin de garantir que la bonne quantité de puissance optique est délivrée aux neurones ciblés. Le PMU est chargé de générer une tension

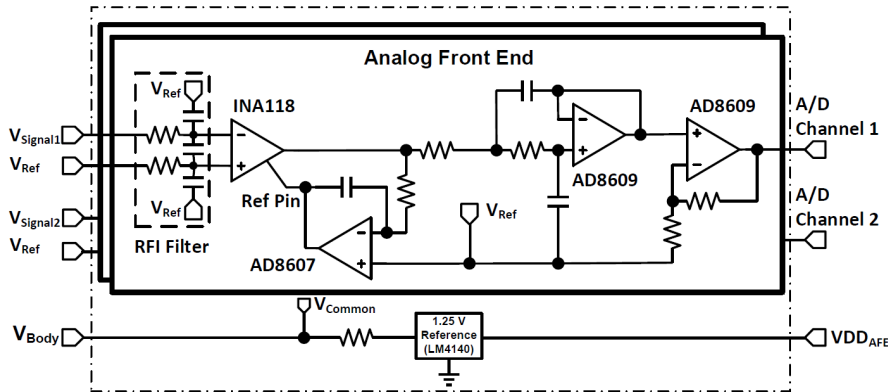


FIGURE 4.3 – Circuit de l’AFE.

d’alimentation stable de 3.3 V au MCU et au transmetteur sans fil ainsi qu’une tension stable de 3.0 V à l’AFE. Le transmetteur sans fil est utilisé afin de transmettre les données à une station de base et pour recevoir la configuration des expériences (temps, patron de stimulation, etc.). Le MCU est utilisé pour générer les patrons de stimulation optique, quantifier les signaux à la sortie de l’AFE, détecter les AP en temps réel, communiquer avec le transmetteur sans fil, et en général, pour commander le fonctionnement de l’ensemble du *headstage*. Chacun des sous-systèmes susmentionnés est passé en revue dans cette section.

#### 4.2.1 Circuit Analog Front End (AFE)

La figure 4.3 présente le circuit de l’AFE. Tel que stipulé précédemment, l’amplitude extracellulaire des AP peut varier entre  $50 \mu\text{V}$  et  $500 \mu\text{V}$  [17] [18]. Avec des signaux d’amplitude aussi faible, un taux de rejet du mode commun (CMRR) élevé et un taux de rejet de l’alimentation (PSRR) élevé sont nécessaires afin d’obtenir une acquisition fiable [18], [101], [48], [43]. L’AFE atteint un haut PSRR en incorporant des composantes ayant un haut PSRR et en possédant son propre filtre d’alimentation qui permet d’éliminer les fluctuations rapides de l’alimentation.

Le premier étage de l’AFE est constitué d’un filtre RFI, d’un amplificateur d’instrumentation haute performance INA118 de *Texas Instruments*, d’un filtre passe-haut du premier ordre en rétroaction avec l’amplificateur d’instrumentation ainsi que d’une référence de tension très précise de 1.25 V (LM4140 de *Texas Instruments*, 0.1 % de précision) possédant un haut PSRR afin de produire une référence du mode commun stable et à très faible bruit. Le filtre RFI permet d’éliminer le bruit haute fréquence pouvant être induit par des radiations électromagnétiques d’appareils électroniques se trouvant à proximité du *headstage*. L’amplificateur d’instrumentation INA118 a été sélectionné puisqu’il possède un très faible bruit référé à l’entrée (*input-referred noise* de  $10 \text{ nV}/\sqrt{\text{Hz}}$  avec un gain de 100 V/V @100Hz) en plus de posséder un CMRR et un PSRR élevé.

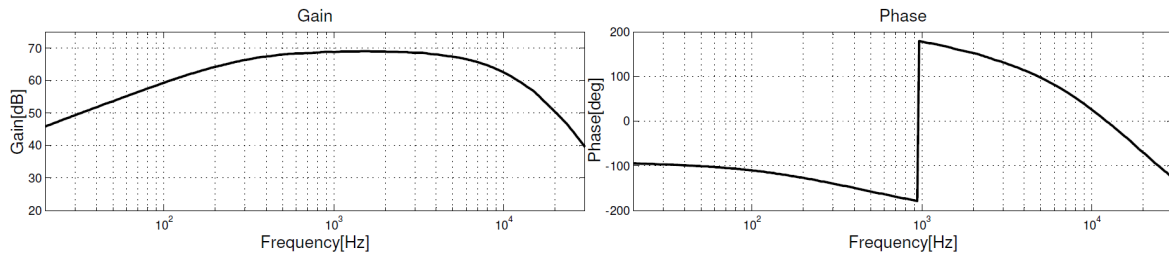


FIGURE 4.4 – Réponse en fréquence de l’AFE mesurée avec un analyseur de spectre Agilent 35670A.

Le deuxième étage d’amplification est constitué d’un filtre *Sallen-Key* passe-bande et le troisième étage d’un amplificateur non-inverseur. L’AFE est alimenté par son propre circuit d’alimentation (TLV70230 de *Texas Instruments*) de 3.0 V suivi d’un filtre passif permettant d’éliminer les fluctuations haute fréquence pouvant être induites par le circuit de stimulation optique lorsque les DEL de stimulation sont activées. L’alimentation de 3.0 V assure un écart de tension important avec la tension de la batterie LiPo de 3.7 V ce qui permet de maximiser le PSRR.

La réponse en fréquence de l’AFE, mesurée avec un analyseur de spectre 35670A de *Agilent*, est présentée à la figure 4.4 et ses caractéristiques sont décrites dans le tableau 4.2. La réponse en fréquence du circuit couvre de 285 Hz à 6580 Hz, permettant ainsi de capturer tout le spectre fréquentiel des AP. Le bruit référé à l’entrée du système, soit  $2.1 \mu V_{rms}$ , est de loin en deçà du  $40 \mu V$  exigé par les requis de conception. Toutes les mesures ont été effectuées alors que les DEL étaient activées avec une fréquence de 100 Hz et un *duty cycle* 10 % afin de reproduire des conditions de test réalistes. Les résultats obtenus démontrent qu’utiliser l’AFE en même temps que le circuit de stimulation optique n’affecte pas ses performances. En effet, on aurait pu s’attendre à ce que le courant élevé circulant dans les DEL (150 mA) impose des niveaux élevés de bruit aux entrées/sorties de l’AFE, ce qui n’est pas le cas grâce aux choix de conception cités plus haut.

TABLE 4.2 – Résumé des caractéristiques mesurées de l’AFE du *headstage* optogénétique à deux canaux.

Paramètre	Valeur
# de canaux	2
Gain	2851 V/V (69.09 dB)
Fréquence de coupure haute	285 Hz
Fréquence de coupure basse	6580 Hz
Bruit référé à l’entrée ( <i>Input-Referred Noise</i> )	$2.1 \mu V_{rms}$
CMRR	110dB @ 1KHz
Consommation électrique	1 mA @ 3.0 V (3 mW)

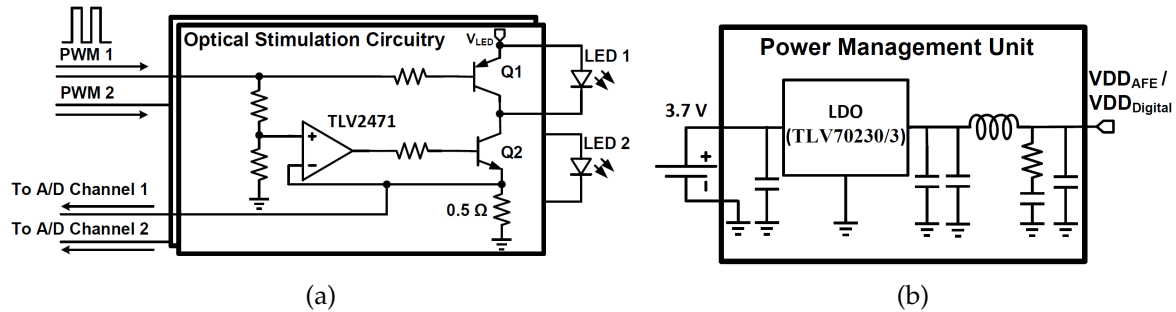


FIGURE 4.5 – a) Circuit de stimulation optique, b) circuit PMU.

#### 4.2.2 Circuit de stimulation optique

Les DEL de stimulation optique sont alimentées par le circuit de stimulation optique. Ce circuit est constitué d'une boucle de rétroaction à base d'amplificateurs opérationnels et de transistors afin de garantir que la bonne quantité de courant est passée à travers chaque DEL. La figure 4.5a présente le circuit de stimulation optique. Le transistor Q1 est utilisé pour s'assurer que la DEL est totalement fermée (non flottante), quand il n'y a pas de signal d'activation à la base de Q1, c'est-à-dire le signal PWM est à son niveau bas, tandis que le transistor Q2 sert à contrôler le courant. L'anode des DEL est directement connectée à la batterie afin de bénéficier d'une tension directe maximale. La tension à l'émetteur de Q2 est lue par un canal de l'ADC du MCU et elle est transmise à la station de base afin de fournir une rétroaction en temps réel de la quantité de courant s'écoulant dans chacune des DEL pendant l'expérience. Selon la courbe I-V de la DEL [102], une tension de 3.3 V correspond à un courant de 150 mA, soit le courant nominal avec lequel la DEL réalise ses performances optimales. Le tableau 4.3 présente les caractéristiques du circuit de stimulation optique.

TABLE 4.3 – Résumé des caractéristiques mesurées du circuit de stimulation optique.

Paramètre	Valeur
# de canaux	2
Tension nominale aux bornes de chaque DEL	3.275 V
Courant nominal dans chaque DEL	150 mA
Temps de monté	1.6 $\mu$ s
Temps de descente	5.1 $\mu$ s
Puissance de décharge dans la DEL	491 mW
Puissance optique à la sortie de la fibre implantée	8 mW (250 mW/mm <sup>2</sup> )



### 4.2.3 Circuit de transmission sans fil

Les communications sans fil sont assurées par une puce à faible consommation énergétique nRF24L01+ de *Nordic Semiconductor*, qui intègre un transmetteur 2.4-GHz de type *Gaussian frequency-shift keying* (GFSK) capable de transmettre à un taux maximal de 2 Mbps. Des tests effectués avec la puce ont démontré un taux de transmission maximal efficace mesuré de  $\sim 1.1$  Mbps, permettant de transmettre sans compression les signaux neuronaux d'un maximum de trois canaux échantillonnés à 20 kHz et quantifiés sur 12 bits (0.96 Mbps nécessaire). Le contrôle, le transfert de paquets et la configuration du transmetteur sont gérés par le MCU via un bus SPI fonctionnant à 8 MHz et les données sont transmises au transmetteur en paquets d'une taille de 32 octets maximum. En mode transmission (@0 dBm), la puce consomme 11.3 mA, contrairement à 13.5 mA en mode de réception. Cette puce est disponible en *package quad-flat no-leads* (QFN) de  $4 \times 4 \text{ mm}^2$  permettant une intégration sur PCB très compacte.

### 4.2.4 Circuit de gestion de l'énergie (PMU)

Le circuit du PMU est présenté à la figure 4.5b. Le rôle de ce circuit est de générer une tension stable de 3.3 V afin d'alimenter le MCU et le transmetteur sans fil. Pour ce faire, un régulateur *low-drop* (TLV70233 de *Texas Instrument*) possédant un PSRR de 51 dB est utilisé. Un réseau de filtres passifs passe-bas est situé à la sortie du régulateur afin de compenser pour le faible PSRR du régulateur et ainsi réduire les perturbations pouvant être engendrées par le circuit de stimulation optique. Un circuit similaire est utilisé afin de fournir une tension stable de 3.0 V à l'AFE et diffère seulement par l'utilisation d'un régulateur TLV70230 au lieu d'un TLV70233.

### 4.2.5 Microcontroller Unit (MCU)

La numérisation des signaux neuronaux, la détection *in situ* des AP, le contrôle des communications sans fil et la génération des patrons de stimulation optique sont effectués via un microcontrôleur MSP430F5328 de *Texas Instrument* [103] disponible en *package* miniature 80BGA de  $5 \times 5 \text{ mm}^2$ . Afin d'utiliser les ressources du microcontrôleur à leur maximum d'efficacité tout en consommant moins d'énergie, un système d'exploitation temps réel (RTOS) a été intégré dans le *firmware* du MCU. *FunkOS* [104], développé par *Funkenstein Software Consulting*, a été sélectionné puisqu'il supporte toutes les fonctionnalités typiques d'un RTOS tout en étant léger ( $< 8$  kbytes) et rapide.

Des RTOS sont utilisés dans d'autres systèmes d'acquisition neuronaux, comme *TinyOS* dans [105]. Contrairement à *TinyOS*, qui est programmé en langage nesC, *FunkOS* utilise le langage C qui est un langage de programmation courant et compatible avec la librairie *DriverLib* des pilotes du MSP430. De plus, *FunkOS* est entièrement compatible avec les interruptions natives du microcontrôleur. Une latence de changement de contexte de  $35 \mu\text{s}$  a été mesu-

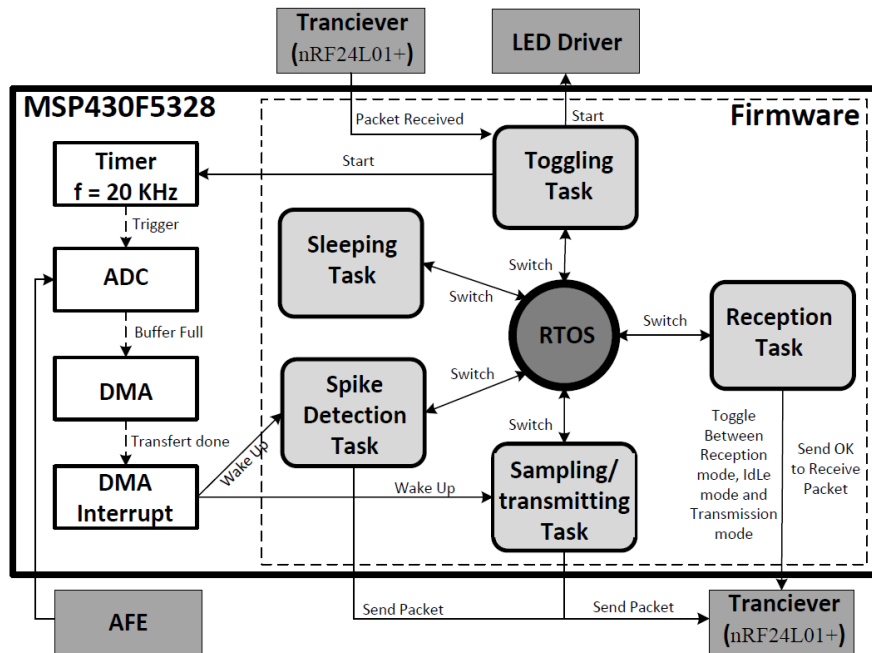


FIGURE 4.6 – Diagramme fonctionnel du *firmware* de contrôle utilisant un RTOS.

rée avec une fréquence de fonctionnement de 8 MHz. Pour comparer FunkOS avec d'autres RTOS courants, la même mesure a été effectuée avec *SYS/BIOS*, *FreeRTOS* et *BRTOS* qui ont généré des latences de changement de contexte plus élevé de  $94 \mu s$ ,  $137 \mu s$  et  $42 \mu s$  respectivement.

Tel que susmentionné, les tâches du RTOS peuvent être utilisées conjointement avec les interruptions natives du système, ce qui permet une réponse rapide aux événements. Cette propriété permet au *headstage* de posséder une fréquence d'échantillonnage élevée de 20 kHz sur deux canaux en utilisant conjointement les interruptions de l'ADC et du circuit de DMA en même temps que les tâches sont exécutées. Une fois que la mémoire tampon interne de l'ADC est remplie d'échantillons (16 échantillons), le DMA est déclenché automatiquement afin de transférer les données depuis le tampon de l'ADC vers un tampon circulaire de grande taille (384 octets) situé en mémoire RAM. Quand un nombre minimum d'échantillons sont dans le tampon circulaire (32 octets), l'interruption du DMA notifie, par l'intermédiaire d'un sémaphore du RTOS, soit la tâche de transmission des données brutes, ou la tâche de détection des AP selon le mode sélectionné par l'utilisateur (transfert des données brutes ou bien des AP détectés). Pendant ce temps, le MCU peut exécuter d'autres tâches ou passer en mode de veille pour économiser de l'énergie.

Le schéma fonctionnel du *firmware* de contrôle est illustré à la figure 4.6. Sans rentrer dans les détails, notons que le *firmware* de contrôle est constitué de cinq tâches : *Toggling Task*, *Reception Task*, *Sampling/transmitting Task*, *Spike Detection Task* et *Sleeping Task*. Avoir une tâche de

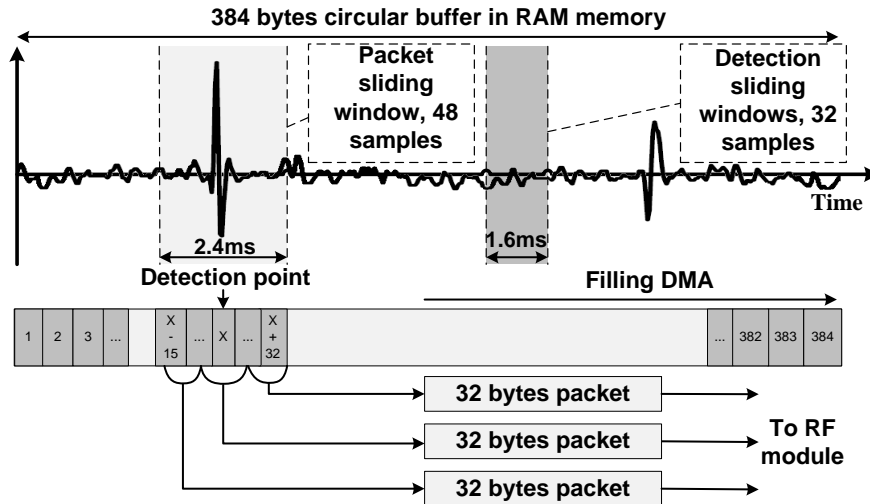


FIGURE 4.7 – Fenêtre glissante de détection et de transmission des AP.

sommeil permet d'économiser de l'énergie en mettant le MCU en mode veille lorsqu'il n'est plus utilisé. Le MCU est réveillé uniquement par les interruptions. En plein fonctionnement et en mode de transmission des signaux bruts, il a été observé que le MCU est dans la tâche de sommeil 73 % du temps, ce qui entraîne une économie d'énergie moyenne mesurée de 3.3 mW.

La tâche de détection des AP (*Spike Detection Task*) traite des fenêtres de 32 échantillons. Comme le montre la figure 4.7, lorsqu'un AP est détecté, une forme d'onde de 48 échantillons est récupérée à partir du tampon circulaire située dans la mémoire RAM, ce qui correspond à 15 échantillons avant et 32 échantillons après le point de détection. Ces échantillons sont ensuite transférés au transmetteur sans fil dans trois paquets de 32 octets chacun. Puisque les échantillons sont sur 12 bits, 8 octets supplémentaires sont disponibles dans chacun des paquets transférés. Ces octets sont utilisés afin de stocker les numéros de paquets, le numéro du canal de la détection ainsi qu'un *timestamp* afin de reconstruire temporellement le signal à la station de base.

#### 4.2.6 PCB

La taille et le poids sont des caractéristiques importantes dans cette application. Toutes les composantes du *headstage* sont assemblées sur un PCB flexible rigide possédant 3 sections rigides et 2 sections flexibles. La section supérieure du PCB est dédiée au transmetteur et au MCU, la section du milieu est dédiée au PMU numérique et au circuit de stimulation optique, alors que la section du bas supporte l'AFE, le PMU analogique et le connecteur *Molex* afin de connecter avec le module implantable. Le design du PCB est constitué de 6 couches avec une largeur minimum des connexions de 100  $\mu\text{m}$  et une taille minimale des

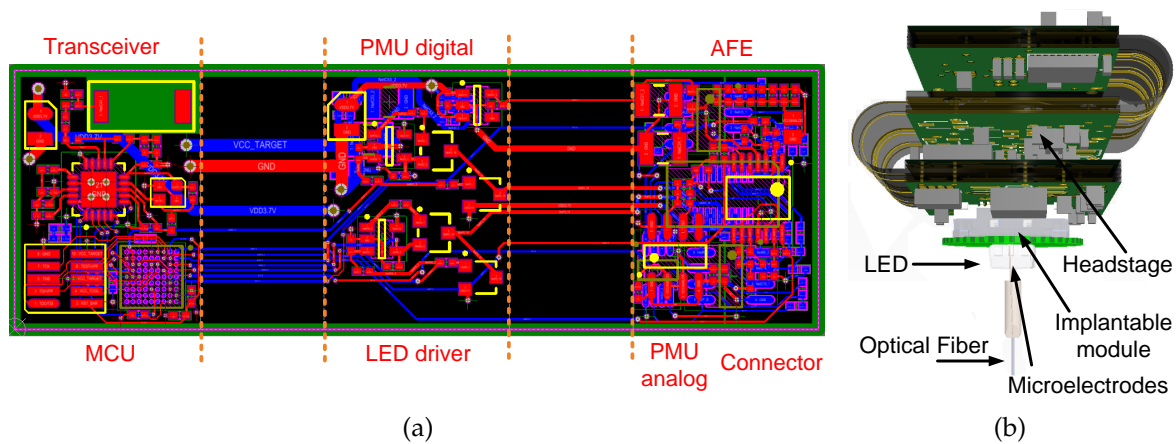


FIGURE 4.8 – a) Design du PCB flexible rigide, b) vue 3D du PCB flexible rigide replié et connecté à un module implantable.

vias de  $200 \mu\text{m}$ . Les sections rigides sont connectées entre-elles via des sections flexibles où il n’y a aucun *power planes*, résultant en une meilleure isolation entre les sections. La figure 4.8a présente le circuit du PCB, tandis qu’un modèle 3D du PCB replié et connecté à un module implantable est montré à la figure 4.8b.

## 4.3 Compression par détection des AP

### 4.3.1 Sélection de l’algorithme de détection optimal

Étant donné que les ressources disponibles dans un MCU de type MSP430 sont très limitées, l’algorithme de détection doit utiliser peu de ressources et être rapide à exécuter. Afin de choisir un algorithme de détection optimal, trois algorithmes de détection ont été implantés dans le *firmware* du *headstage* et testés. Les trois algorithmes retenus pour les tests sont l’algorithme de détection à valeur absolue, l’algorithme NEO et l’algorithme MTEO. Les algorithmes basés sur des convolutions ou sur des changements de base complexes ont été rejetés à cause de leur trop grande complexité. Il est à noter que les résultats présentés dans cette section ont fait l’objet d’une publication scientifique, voir [95].

### Signaux neuronaux synthétiques

Des signaux neuronaux synthétiques ont été utilisés pour tester les algorithmes de détection. Ces signaux ont été produits dans *Matlab* avec des formes d’ondes issues de la banque de signaux en ligne de l’Université Leicester [12]. Trois signaux d’une durée de 5 secondes chacun possédant une fréquence d’apparition des AP de 150 AP/s et un SNR de 20dB, 15dB et 10dB ont été construits puis téléchargés dans un générateur de fonction aléatoire AFG3101 de *Tektronix*. Le SNR des signaux neuronaux synthétiques est calculé avec la formule suivante :

$$SNR_{dB} = \log\left(\frac{P_{ap}}{P_n}\right) \quad (4.1)$$

Où  $P_{ap}$  est la puissance de tous les AP inclus dans le signal et  $P_n$  est la puissance du bruit sans les AP. Les signaux ont été atténués à la sortie du générateur de fonction en utilisant un diviseur de tension précis afin de produire une amplitude représentative des signaux neuronaux, de l'ordre du microvolt.

### Résultats de détection

Le tableau 4.4 présente les taux de détection de vrais positifs (VP) et de faux positif (FP) obtenus avec chacun des trois détecteurs. Les taux de VP ont été calculés en divisant le nombre de bonnes détections par le nombre total d'AP inclus dans le signal. Les taux de FP ont été calculés en divisant le nombre de mauvaises détections par le nombre total d'AP détectés. Un évènement de détection est considéré comme un VP si un AP est inclus à l'intérieur des 48 échantillons transmis par le *headstage* immédiatement après la détection. Les seuils de détection ont été ajustés manuellement afin d'obtenir les taux de détection les plus élevés pour chacun des algorithmes et ils ont été déterminés après plusieurs itérations successives.

TABLE 4.4 – Taux de détection du *headstage* à deux canaux avec les algorithmes à valeur absolue, NEO et MTEO selon différents SNR : 20dB, 15dB et 10dB.

SNR	Algorithme	VP [%]	FP [%]
20dB	Absolue	96.8	3.78
	NEO	95.9	9.00
	MTEO	96.3	7.75
15dB	Absolue	94.4	8.20
	NEO	93.0	14.1
	MTEO	92.7	14.3
10dB	Absolue	78.7	29.5
	NEO	82.5	31.0
	MTEO	76.6	30.4

L'analyse du tableau 4.4 permet de constater que l'algorithme de détection à valeur absolue génère les meilleurs résultats pour presque tous les SNR, à l'exception de l'algorithme NEO qui produit le meilleur résultat de VP pour un SNR de 10dB.

### Temps d'exécution des algorithmes

Le tableau 4.5 présente les temps d'exécution de chacun des algorithmes pour des fréquences de fonctionnement de 8MHz et de 16MHz et pour un nombre de canaux variant de 1 à 4. Puisque le *headstage* possède seulement deux canaux d'enregistrement, les deux canaux supplémentaires ont été obtenus par duplication logicielle afin de simuler quatre canaux.

Les temps affichés représentent le temps nécessaire au MCU pour traiter 32 échantillons par canal. Dans certains cas, le MCU n'a pas été assez rapide pour traiter toutes les données sans pertes. Ces résultats sont présentés en rouge à titre indicatif seulement.

TABLE 4.5 – Temps d'exécution du *headstage* à deux canaux pour traiter 32 échantillons par canal avec les algorithmes à valeur absolue, NEO et MTEO.

Algorithme	# de canaux	Temps d'exécution 8MHz[ms]	Temps d'exécution 16MHz[ms]
Absolue	1	0.0917	0.0458
NEO	1	0.562	0.280
MTEO	1	1.17	0.587
Absolue	2	0.164	0.0820
NEO	2	1.10	0.552
MTEO	2	2.33	1.16
Absolue	4	0.309	0.154
NEO	4	2.19	1.09
MTEO	4	4.63	2.32

L'analyse du tableau 4.5 permet de conclure que l'algorithme de détection à valeur absolue est celui qui nécessite le moins de temps de calcul. De plus, c'est le seul algorithme qui peut être utilisé sur quatre canaux d'enregistrement à une fréquence du 8MHz.

Puisque l'algorithme de détection à valeur absolue obtient majoritairement les meilleurs résultats de détection tout en nécessitant le moins de temps de calcul parmi tous les algorithmes testés, c'est cet algorithme qui est retenu pour être implémenté de façon permanente dans le *headstage*.

### 4.3.2 Algorithme de seuillage adaptatif

#### Aspects mathématiques

L'algorithme de détection des AP à valeur absolue nécessite l'utilisation d'un seuil qui doit être fixé selon l'amplitude du bruit. La nouvelle technique de seuillage automatique proposée s'inspire de [61] et consiste à estimer l'écart-type du bruit en exploitant certaines propriétés statistiques de la loi demi-normale. Puisque le bruit provenant de l'interface entre les microélectrodes à haute impédance et les neurones est considéré comme dominant [106] (en ne tenant pas compte du bruit neuronal provenant de neurones éloignés), le bruit peut être approximé par du bruit blanc gaussien possédant une distribution statistique normale [61]. Ce bruit, une fois replié par l'algorithme de détection à valeur absolue, suit une loi demi-normale. La densité de probabilité de la loi demi-normale est présentée à la figure 4.9 et est

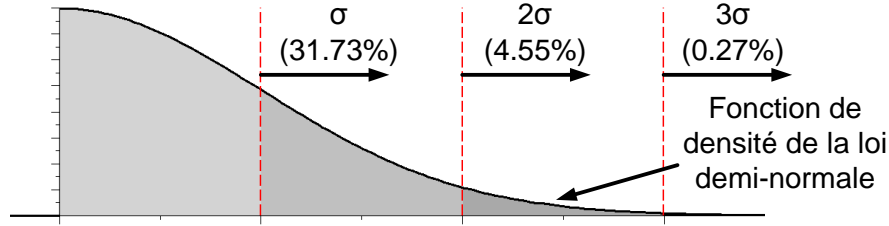


FIGURE 4.9 – Fonction de densité de la loi demi-normale présentée avec des multiples de son écart-type ( $\sigma$ ).

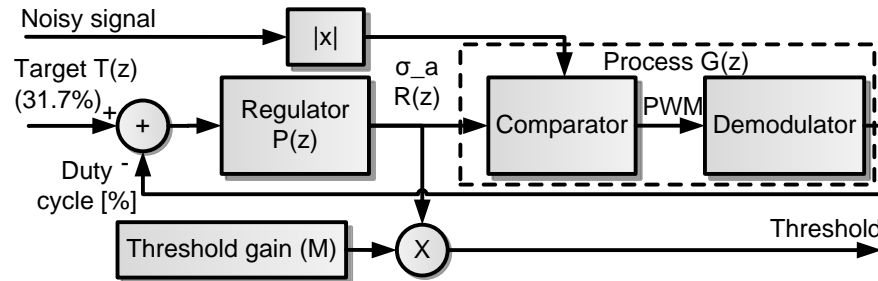


FIGURE 4.10 – Boucle de contrôle permettant de calculer le seuil de détection basé sur un multiple de l'écart-type estimé ( $\sigma_a$ ) du signal.

définie par l'équation suivante :

$$f(x) = \begin{cases} \text{Si } x > 0, & \frac{1}{\sigma} \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2\sigma^2}} \\ \text{Sinon,} & 0 \end{cases}$$

Où  $\sigma$  est l'écart-type du signal. Avec cette densité de probabilité, on sait que la probabilité que la valeur absolue d'un bruit gaussien excède l'écart-type du signal est de 31.7% :

$$31.7\% = 100\% - \int_0^\sigma \frac{1}{\sigma} \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2\sigma^2}} dx \times 100\%. \quad (4.2)$$

Afin d'exploiter cette propriété, l'algorithme proposé estime l'écart-type de la valeur absolue du bruit à l'aide d'une boucle de contrôle. Dans cette boucle, le procédé à contrôler estime le ratio du nombre d'échantillons supérieur à l'écart-type. Ce procédé numérique compare la valeur absolue du signal bruité avec la sortie du régulateur. Cette comparaison génère un signal de type PWM, 1 pour plus grand et 0 pour plus petit, possédant un *duty cycle* exprimant le nombre d'échantillons dont la valeur absolue est plus grande que la sortie du régulateur. Le procédé applique ensuite une démodulation afin de transformer le signal PWM en ratio pouvant être soustrait à la consigne dans la rétroaction de la boucle. À chaque nouvelle itération déclenchée par l'arrivée d'un nouvel échantillon, le régulateur tente de minimiser l'erreur entre la consigne, de valeur 31.7%, afin de trouver l'approximation de l'écart-type. Une fois que la boucle a atteint son régime permanent, le signal de contrôle du

régulateur représente l'écart-type estimé. Cette boucle de contrôle est présentée à la figure 4.10, où  $\sigma_a$  est l'écart-type estimé et  $M$  est un facteur multiplicatif permettant d'ajuster le seuil ( $Seuil = M \times \sigma_a$ ). La démonstration mathématique de cet algorithme est présentée ci-dessous. Dans cette démonstration,  $T(z)$  est la consigne,  $P(z)$  est le régulateur basé sur un intégrateur afin d'assurer une erreur nulle en régime permanent,  $G(z)$  est le procédé et  $R(z)$  est la sortie du régulateur. Premièrement, la fonction de transfert discrète du régulateur peut être exprimée comme suit :

$$R(z) = P(z) \cdot (T(z) - R(z) \cdot G(z)) \quad (4.3)$$

Ce qui devient :

$$R(z) = \frac{P(z) \cdot T(z)}{1 + P(z) \cdot G(z)} \quad (4.4)$$

Où  $T(z) = \frac{0.317z}{z-1}$  et  $P(z) = \frac{kz}{z-1}$  avec  $k$  le gain de l'intégrateur. Selon le théorème de la valeur finale, il est possible de calculer  $R(\infty)$  avec l'équation suivante :

$$R(\infty) = \lim_{z \rightarrow 1} R(z)(1 - z^{-1}) \quad (4.5)$$

$$\Rightarrow R(\infty) = \lim_{z \rightarrow 1} \frac{\frac{0.317z}{z-1} \cdot \frac{kz}{z-1}}{1 + \frac{kz}{z-1} \cdot G(z)} \cdot (1 - z^{-1}) \quad (4.6)$$

$$\Rightarrow \lim_{z \rightarrow 1} = \frac{0.317kz}{(z-1) + kzG(z)} = \frac{0.317}{G(1)} \quad (4.7)$$

$$\Rightarrow R(\infty) \cdot G(1) = 0.317 \quad (4.8)$$

Or,  $R(z) \cdot G(z)$  peut être exprimé par l'équation suivante :

$$R(z) \cdot G(z) = \frac{1}{N} \sum_{n=0}^{N-1} z^{-n} \begin{cases} 1 \text{ si } |x| \cdot z^{-n} > R \cdot z^{-n} \\ 0 \text{ sinon} \end{cases} \quad (4.9)$$

Ainsi,  $R(\infty) \cdot G(1)$  devient :

$$R(\infty) \cdot G(1) = \frac{1}{N} \sum_{n=0}^{N-1} \begin{cases} 1 \text{ si } |x| > R(\infty) \\ 0 \text{ sinon} \end{cases} = 0.317 \quad (4.10)$$

En prenant  $N \rightarrow \infty$ , l'on obtient le ratio d'échantillons supérieurs à  $R(\infty)$ , soit 0.317 ce qui correspond à l'écart-type ( $R(\infty) = \sigma$ ). En calculant l'algorithme sur une fenêtre de  $N < \infty$  échantillons, l'on obtient une approximation de l'écart-type ( $\sigma_a$ ).



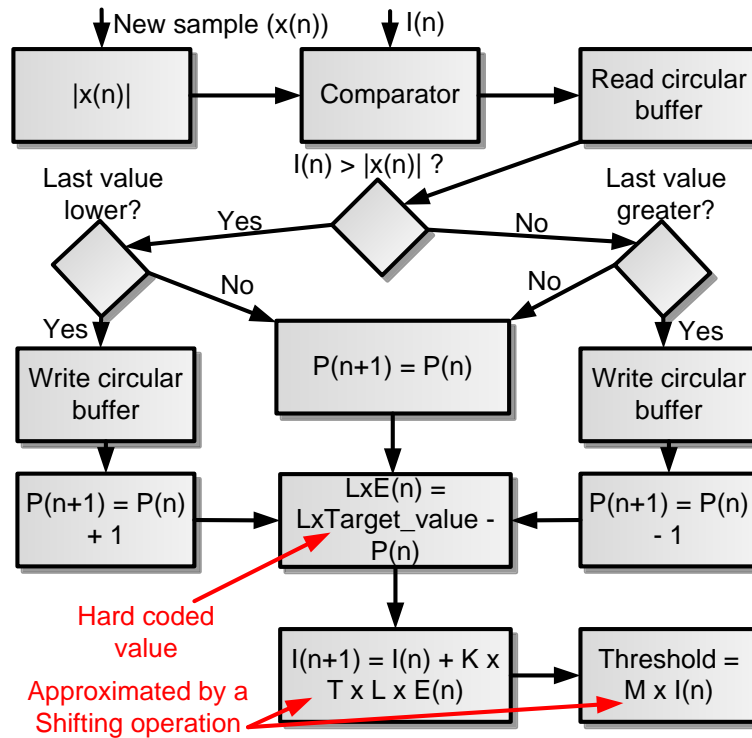


FIGURE 4.11 – Diagramme fonctionnel de l’algorithme permettant de calculer le seuil de détection.

### Implémentation optimisée pour MCU

L’algorithme de seuillage adaptatif doit être exécuté à chaque nouvel échantillon et nécessite, par conséquent, une implémentation numérique efficace. Avec deux canaux d’enregistrement et une fréquence d’échantillonnage de 20 kHz, le temps maximal d’exécution de l’algorithme est de  $\sim 25 \mu s$ . En notant qu’une multiplication de deux entiers de 16 bits sur MSP430 nécessite jusqu’à  $2.1 \mu s$  (17 cycles d’horloge avec le module matériel de multiplication activé) et qu’une division implémentée de façon logicielle nécessite au minimum  $2.6 \mu s$  (21 cycles d’horloge, dépend du compilateur, de l’algorithme de division et des opérandes) à une fréquence de fonctionnement de 8 MHz, ces opérations doivent nécessairement être évitées. L’approche logicielle proposée utilise uniquement des opérations de décalage de bits permettant d’effectuer des multiplications et des divisions par des puissances de 2 ainsi que des opérations d’additions et de soustractions.

Dans un premier temps, le régulateur  $P(z)$  est implémenté en utilisant un intégrateur afin d’assurer une erreur nulle sur  $\sigma$  en régime permanent. Cet intégrateur est construit en utilisant une somme de Riemann :

$$I(n+1) = I(n) + K \times T \times E(n) \quad (4.11)$$

Où  $K$  est le gain de l’intégrateur,  $T$  est la période d’échantillonnage et  $E(n)$  le signal d’erreur. La sortie du régulateur  $I(n)$ , qui est aussi  $\sigma_a$ , est comparée avec la valeur absolue de

l'échantillon courant. La sortie du comparateur est mise à 1 ou à 0 selon le résultat de la comparaison afin de générer le signal PWM. Un tableau en mémoire RAM est utilisé comme tampon circulaire afin de stocker 256 bits de comparaison par canal ( $L = 256$ , déterminé empiriquement). Avant de stocker le nouveau bit de comparaison, le dernier bit devant être écrasé est lu. Si le nouveau bit est à 1 et le dernier bit à 0, un compteur est incrémenté. Si le nouveau bit est à 0 et le dernier bit à 1, le même compteur est décrémenté. Dans tous les autres cas, la valeur du compteur reste inchangée. Ce compteur contient la somme de tous les bits en mémoire qui, lorsque divisé par la taille du tampon circulaire  $L$ , représente le pourcentage d'échantillons dont la valeur était supérieure à  $\sigma_a$  et peut être représenté mathématiquement par  $P(n)$ . Puisque la taille du tampon circulaire est connue à l'avance, et pour éviter d'utiliser une opération de division, la consigne de la boucle de contrôle est ajustée à 31.7% multipliée par la taille du tampon circulaire. L'opération  $E(n) = \text{consigne} - P(n)/L$  devient  $L \times E(n) = L \times \text{consigne} - P(n)$ . Le facteur  $L$  qui multiplie l'erreur  $E(n)$  peut être joint avec le gain  $K$  de l'intégrateur et avec la période d'échantillonnage  $T$ . Puisque  $K$ ,  $L$  et  $T$  sont connus à l'avance, la multiplication  $K \times T \times L \times E(n)$  est approximée par une simple opération de décalage de bits de l'erreur  $E(n)$ . Idem pour la multiplication de  $\sigma_a$  déterminant le seuil qui est aussi approximée par une simple opération de décalage de bits. Les étapes de l'algorithme sont présentées dans le diagramme de la figure 4.11.

### Innovations de l'algorithme de seuillage adaptatif

La méthode de seuillage adaptative proposée ne nécessite aucun circuit dédié et est entièrement numérique, soit deux avantages de taille pour diminuer les coûts, augmenter la fiabilité et accélérer le développement. Cette caractéristique avantage l'algorithme proposé, contrairement au système précédent présenté dans [61], qui utilise une implémentation analogique nécessitant un circuit dédié. Un circuit ASIC numérique exploitant certaines propriétés de la loi normale afin de seuiller les coefficients d'une DWT est présenté dans [107]. L'algorithme présenté dans [107] diffère de celui présenté dans cette section selon plusieurs points. Notamment, cet algorithme est exécuté sur une fenêtre fixe de coefficients, ce qui l'empêche d'être exécuté sur un signal continu i.e. il ne possède aucune mémoire des événements précédant la fenêtre, contrairement à l'algorithme proposé qui utilise un démodulateur avec mémoire interne, lui permettant d'être exécuté en continu avec un minimum d'opérations. De plus, les systèmes présentés dans [61] et [107] exploitent les propriétés de la loi normale du bruit gaussien, alors que le nouvel algorithme exploite une loi demi-normale, ce qui constitue un net avantage étant donné que le bruit n'est jamais parfaitement gaussien en pratique et souvent non symétrique autour de 0 volt. Avec la méthode proposée, c'est l'intégralité du bruit qui est utilisé pour déterminer le seuil de détection, alors que les méthodes précédentes [61] et [107] n'utilisent que les échantillons positifs supérieurs au seuil estimé dans ses calculs. Cette amélioration confère beaucoup plus de précision au nouvel algorithme lorsque le seuil est combiné à un algorithme de détection en valeur absolue dont

le but est de détecter la présence d'AP dans un signal bruité. Il est à noter que cet algorithme et son implémentation ont fait l'objet de quatre publications scientifiques [96], [108], [98], [100] et d'une déclaration d'invention (# de divulgation 01691-DIV-02) auprès de l'agent de brevet de l'Université Laval ayant mené à l'obtention d'un brevet provisoire aux États-Unis (# de référence : 000819-0303).

### 4.3.3 Reconstruction temporelle des signaux neuronaux

Les AP détectés doivent être replacés temporellement afin de reconstruire le signal original. Pour ce faire, la valeur d'un compteur de 32 bits synchronisé avec l'échantillonnage des canaux est mémorisée lors de la détection d'un AP et ajoutée à chacun des paquets transmis. La valeur de ce compteur correspond au moment de la détection et il permet de reconstruire les signaux neuronaux pendant une durée maximale de 60 heures avant de produire un *overflow*. De plus, ce compteur est utilisé afin de déterminer s'il y a eu des pertes de paquets durant la transmission. Un AP est transmis en trois paquets de 32 bits. Si l'un des trois paquets reçus ne possède par le même *time stamp* que les deux autres, alors une perte est détectée et les paquets sont rejetés.

## 4.4 Caractéristiques du *headstage*

### 4.4.1 Caractéristiques physiques

La figure 4.12a présente le PCB flexible rigide final avec chacun des circuits et composants identifiés. Les dimensions, une fois le PCB replié, sont de 1.6 x 2.0 x 2.0 cm et le poids du système sans la batterie est de 4.9 g. La figure 4.12b présente le système complet avec un module implantable installé et la batterie sur le côté. Les figures 4.12c et 4.12d présentent le PCB flexible rigide replié sous deux angles différents.

### 4.4.2 Autonomie et consommation électrique

La figure 4.13a présente la consommation de chacun des modules du *headstage* en transmettant les signaux neuronaux bruts pendant une stimulation optique de fréquence 100 Hz avec un *duty cycle* de 10%. La consommation totale est de 113 mW générant une autonomie de 3h avec une batterie LiPo de 100 mAh.

La figure 4.13b présente la consommation de chacun des modules du *headstage* ainsi que les gains engendrés par la détection *in situ* des AP. Les gains de consommation du système en fonction de *firing rate* typiques sont présentés dans le tableau 4.6 et peuvent atteindre jusqu'à 9.82% de la consommation totale, générant une autonomie de 3h30 avec une batterie LiPo de 100 mAh.

Il est à noter que sans stimulation optique, l'autonomie du *headstage* peut atteindre 8h avec une batterie LiPo de 100 mAh.

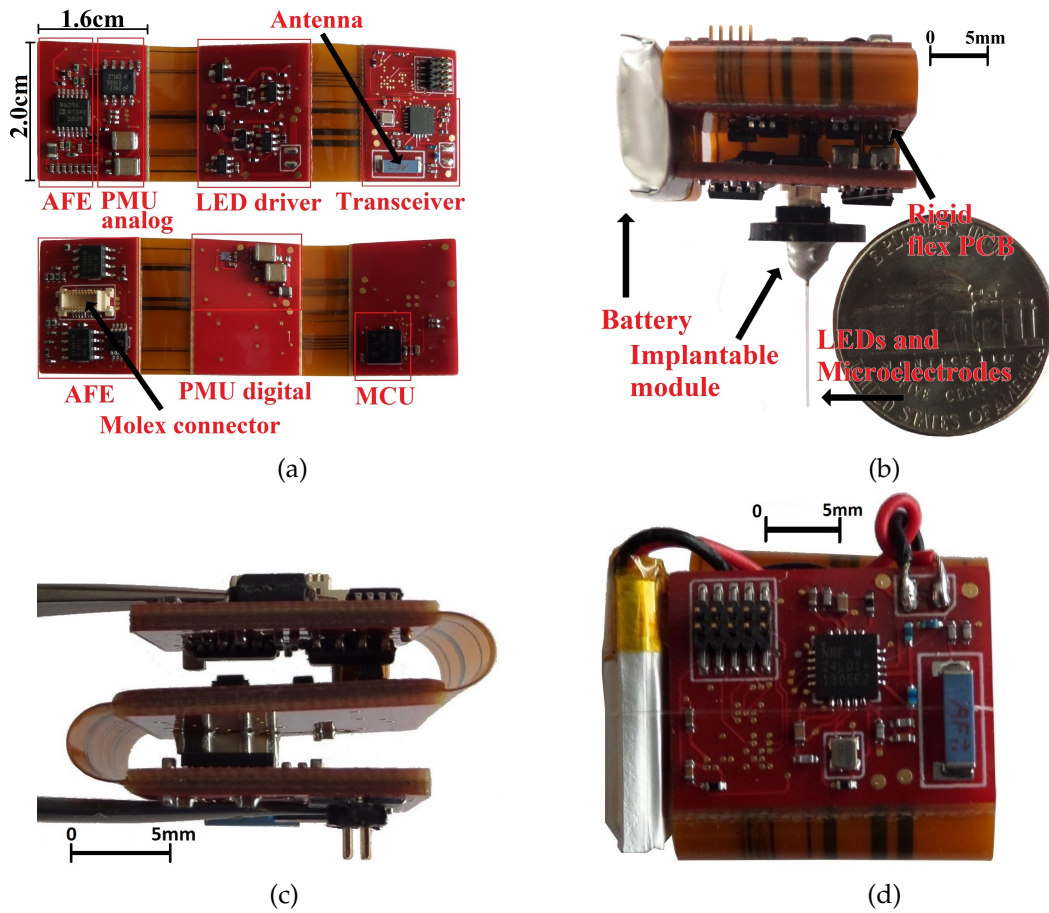


FIGURE 4.12 – a) Vue non repliée du PCB flexible rigide, b) PCB flexible rigide avec la batterie sur le côté et le module implantable avec une DEL installée, c) vue repliée sur le côté, d) vue repliée sur le dessus.

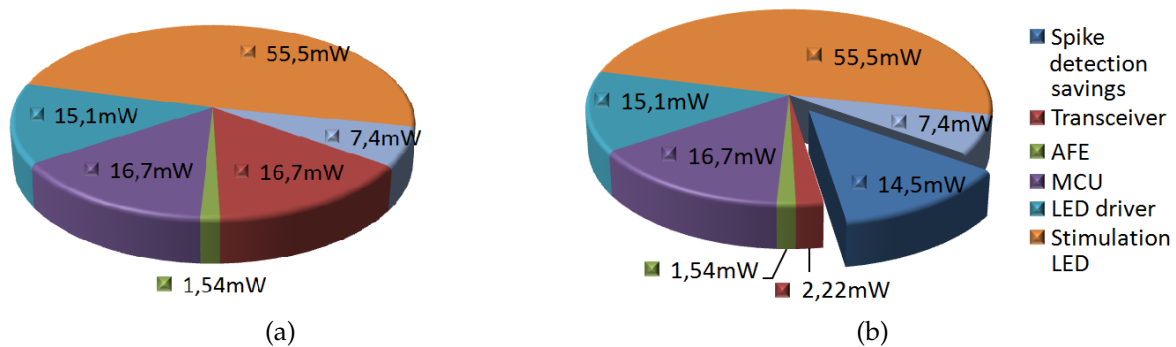


FIGURE 4.13 – Consommation électrique des sous-systèmes du *headstage*. a) Consommation avec transmission des signaux bruts, b) Consommation avec détection et transmission des AP avec les gains de consommation pour un *firing rate* de 50 AP/s.

TABLE 4.6 – Gains en consommation du *headstage* à deux canaux engendré par la détection des AP en fonction du *firing rate*.

Détection des AP	Firing rate [AP/s]	Puissance [mW]	Économie d'énergie [%]
Désactivé	—	113	—
Activé	0	96.6	14.5
Activé	50	98.5	12.8
Activé	100	100.4	11.4
Activé	150	101.9	9.82

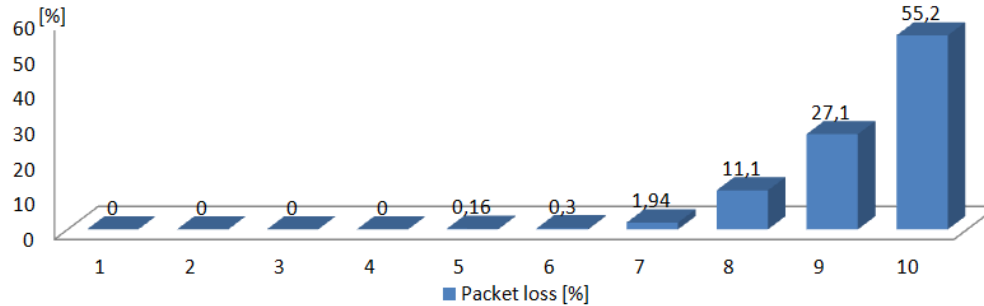


FIGURE 4.14 – Pertes de paquets mesurés en fonction de la distance entre l'émetteur (*headstage*) et le récepteur (station de base).

#### 4.4.3 Portée du transmetteur

Afin de quantifier les performances de portées du transmetteur, la figure 4.14 présente la quantité de perte de paquets en fonction de la distance entre le *headstage* et la station de base. Les tests ont été effectués en éloignant progressivement le *headstage* de la station de base. Pour chacun des essais, un million de paquets ont été transmis d'une distance variant de 1 m à 10 m. Les tests ont été répétés quatre fois pour chaque distance avec l'antenne du *headstage* orientée dans des directions différentes (suivant des rotations successives de 90 degrés) et les résultats présentés sont la moyenne des quatre essais. On peut constater que les pertes de paquets sont négligeables pour des distances  $\leq 5$  m, ce qui est une portée suffisante pour l'application visée.

## 4.5 Validation *in vitro*

Cette section présente les performances du *headstage* optogénétique à deux canaux mesurées lors d'expériences *in vitro*. Le montage de test est présenté à la figure 4.15, dans lequel le *headstage* transmet les données sans fil à une station de base. Les signaux utilisés dans les tests *in vitro* ont été synthétisés suivant la même procédure que celle présentée à la section 4.3.1.

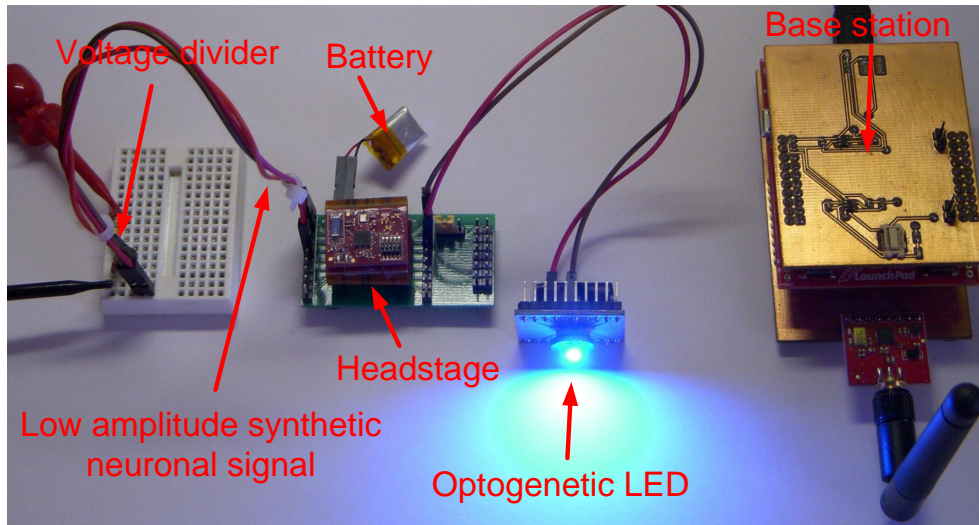


FIGURE 4.15 – Montage de test *in vitro* du *headstage* optogénétique à deux canaux.

#### 4.5.1 Résultats d'enregistrements

Cette section présente des résultats d'enregistrement de signaux neuronaux synthétiques de faible amplitude. Pour reproduire des conditions expérimentales réalistes, toutes les mesures ont été effectuées avec la stimulation optique activée avec une fréquence 100 Hz et un *duty cycle* de 10%. La figure 4.16 montre une capture d'oscilloscope de la sortie d'un canal de l'AFE et de la tension aux bornes d'une des DEL. Cette figure permet de constater que le courant important circulant dans la DEL lorsque le PWM est activé (150 mA) n'induit pas de distorsions visibles à la sortie de l'AFE. La figure 4.17a présente une sortie typique du système dans lequel les signaux d'entrée ont des AP d'amplitude inférieure à 150  $\mu\text{V}$  ainsi que le signal de stimulation optique affiché sur le même graphique. La figure 4.17b montre un grossissement où des potentiels d'action d'amplitudes différentes sont traités avec succès par le *headstage*. On peut constater que le niveau de bruit inhérent au système est nettement inférieur aux AP, qui sont capturés sans distorsion malgré leur faible amplitude.

#### 4.5.2 Performances de détection des AP

##### Paramètres d'implémentation

L'algorithme de détection des AP est implémenté avec les paramètres suivants : le paramètre  $L \times \text{SetPoint}$  est approximé par l'entier 81 ( $256 \times 0.317 \approx 81$ ). Le paramètre  $L \times K \times T$  est approximé par un décalage des bits de onze positions vers la droite, ce qui est équivalent à un gain d'intégrateur  $K = 4 \times 10^{-4}$ , puisque  $256 \times 4 \times 10^{-4} \times 1/20000 = 2^{-11}$ . Finalement, le gain  $M$  du seuil est approximé par une opération de décalage de deux positions vers la gauche, soit  $M = 4$ .

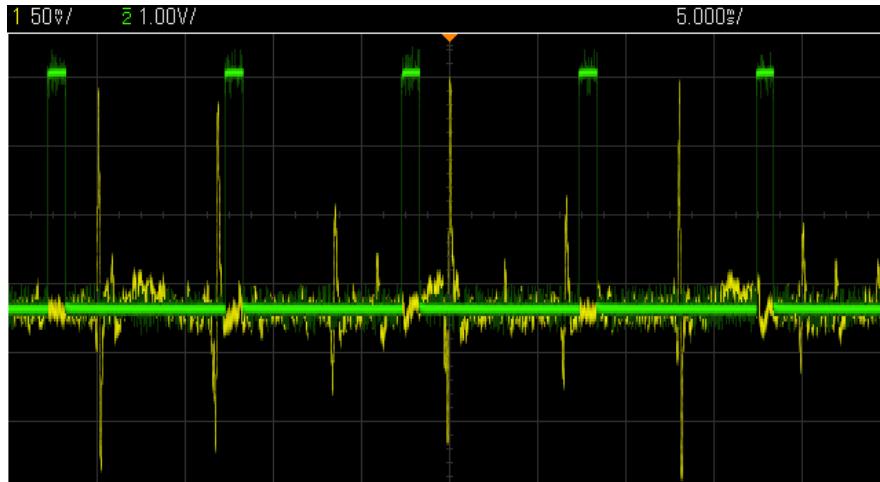
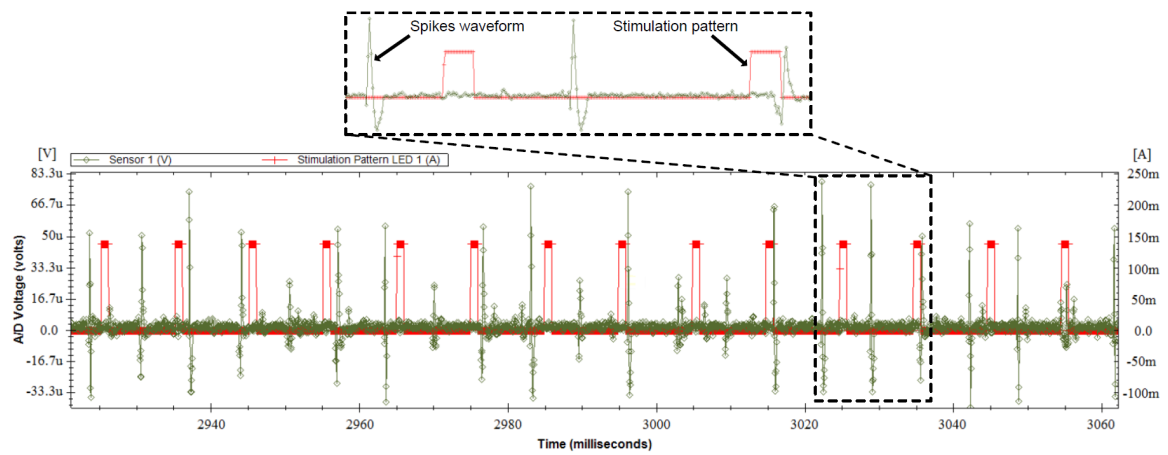
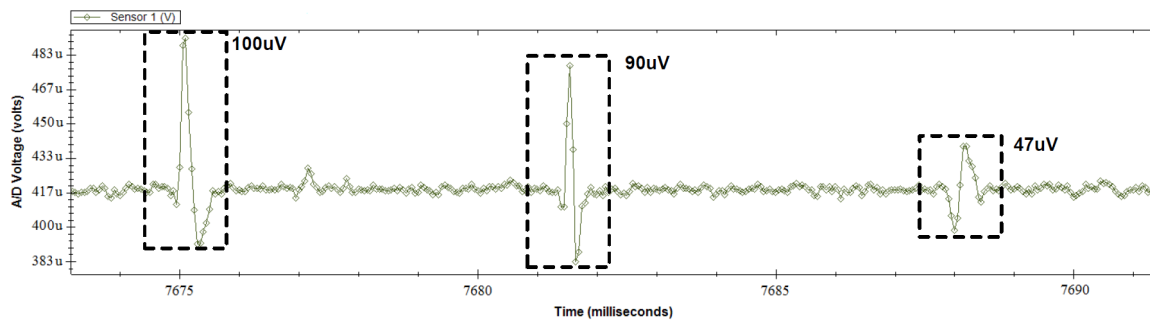


FIGURE 4.16 – Signal neuronal constitué d'AP d'amplitude *peak-to-peak*  $<150 \mu\text{V}$  capté à la sortie de l'AFE en même temps que le patron de stimulation aux bornes de la DEL, acquisitionnés par un oscilloscope MSO-X 2024A d'Agilent.



(a)



(b)

FIGURE 4.17 – Signaux neuronaux acquisitionnés en utilisant l'ensemble de la chaîne d'acquisition de données, du *headstage* à la station de base. a) Patron de stimulation et signal neuronal captés par le *headstage* sur le même graphique, b) grossissement sur des AP de faible amplitude.

TABLE 4.7 – Résultats comparatifs de détection entre la méthode de seuillage adaptative proposée exécutée par le *headstage* à deux canaux versus d’autres méthodes exécutés sur les signaux bruts importés dans *Matlab* et possédant un *firing rate* variant de 25, 75 à 125 AP/s.

Signal		Méthode proposée			Méthode proposée (non optimisée)			RMS			MAD (Wave Clus)		
SNR (dB)	AP	Total	VP	FP	Total	VP	FP	Total	VP	FP	Total	VP	FP
0	3844	2753	2189 (59.9%)	564 (20.5%)	2679	2162 (56.2%)	517 (19.3%)	2662	2164 (56.3%)	498 (18.7%)	2796	2245 (58.4%)	551 (19.7%)
5	3844	3801	3690 (96.0%)	111 (2.92%)	3716	3657 (95.1%)	59 (1.59%)	3747	3681 (95.8%)	66 (1.76%)	3787	3683 (95.8%)	104 (2.75%)
10	3844	3847	3818 (99.3%)	29 (0.75%)	3827	3807 (99.0%)	20 (0.52%)	3841	3815 (99.2%)	26 (0.68%)	3838	3814 (99.2%)	24 (0.63%)
15	3844	3880	3844 (100%)	36 (0.93%)	3871	3844 (100%)	27 (0.70%)	3860	3844 (100%)	16 (0.41%)	3870	3844 (100%)	26 (0.67%)

## Résultats de détection

Dans cette section, les résultats ont été obtenus en transmettant simultanément les AP détectés ainsi que le signal complet, et ce, afin de déterminer le nombre de VP et de FP à partir du signal original. La méthode de seuillage automatique proposée est comparée avec d’autres méthodes de seuillage adaptatif implémentées dans *Matlab* et exécutées sur le signal brut acquisitionné i.e. sur le même signal ayant été traité par le *headstage*. Les algorithmes implémentés dans *Matlab* n’ont fait l’objet d’aucune optimisation (nombres à virgules flottantes, multiplications, divisions, etc.) afin de comparer l’algorithme proposé avec les meilleurs résultats pouvant être obtenus avec d’autres algorithmes. Les méthodes de seuillage implémentées dans *Matlab* sont les suivantes :

- Méthode proposée non optimisée afin de quantifier de la perte de précision engendrée par les optimisations de l’algorithme.
- Méthode basée sur la valeur RMS du signal et qui est présentée à l’équation 2.2. Le seuil est évalué comme un multiple de  $X_{rms}$ , fixé à un multiple de 4 afin d’obtenir le même gain que pour la méthode proposée. La taille de la fenêtre de calcul a été fixée à 256 points.
- Méthode basée sur l’opérateur MAD (*Wave Clus*) et qui est présentée à l’équation 2.1. Le seuil est déterminé comme un multiple de  $\delta$ , fixé à un multiple de 4 afin d’obtenir le même gain que pour la méthode proposée et que dans le logiciel *Wave Clus*. Il est important de noter que cette méthode doit être exécutée sur le signal en entier i.e. ne peut pas être exécutée *in situ* et sert de point de comparaison seulement.

Il est à noter que les résultats reportés dans cette section ont été présentés dans deux publications, voir [96] et [98].

Le tableau 4.7 présente les résultats de détection obtenus avec différents signaux synthétiques de 60 secondes chacun. Chaque signal a été généré avec un SNR différent, soit avec un SNR de 15 dB, 10 dB, 5 dB ou de 0 dB, ainsi qu’avec un *firing rate* variant de 25 à 125 AP/s. Afin de faciliter l’interprétation des résultats, les augmentations des VP et des FP ob-



TABLE 4.8 – Variations des résultats de détection présentés au tableau 4.7 entre la méthode de seuillage adaptative proposée et les autres méthodes.

Signal	Méthode proposée (non optimisée)		RMS		MAD (Wave Clus)	
SNR (dB)	VP (%)	FP (%)	VP (%)	FP (%)	VP (%)	FP (%)
0	+3.7	+1.2	+3.6	+1.8	+1.5	+0.8
5	+0.9	+1.33	+0.2	+1.16	+0.2	+0.17
10	+0.3	+0.23	+0.1	+0.07	+0.1	+0.12
15	0	+0.23	0	+0.52	0	+0.26

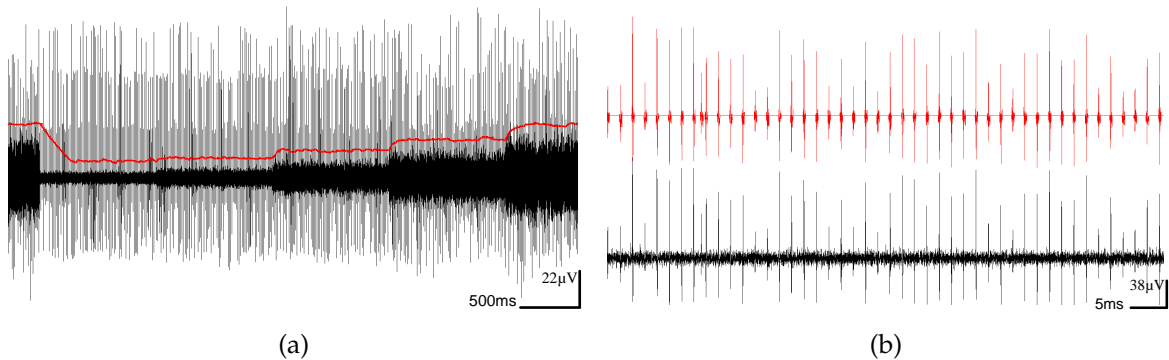


FIGURE 4.18 – a) Variation automatique du seuil pour un SNR variant de 2dB, 20dB, 15dB, 10dB à 5dB avec un *firing rate* constant de 100 AP/s, b) signal reconstruit (en rouge) à partir des AP détectés dans un signal (en noir) ayant un SNR de 20 dB.

tenus entre la méthode proposée et les autres méthodes sont présentées dans le tableau 4.8. On peut constater à l’analyse des résultats du tableau 4.7 et 4.8 que la méthode développée rivalise avantageusement avec les autres méthodes exigeant beaucoup plus de ressources. En fait, la méthode proposée obtient les meilleurs résultats de VP dans tous les tests effectués, au prix d’une augmentation négligeable du nombre de FP (1.8% dans le pire cas). Ces résultats permettent de conclure que les optimisations apportées à l’algorithme n’impactent pas significativement ses performances et que la méthode proposée est aussi performante, sinon meilleure, que les méthodes RMS et MAD beaucoup plus exigeantes en calculs.

Afin d’obtenir un aperçu de la robustesse de la méthode proposée face à une variation brusque du SNR, la figure 4.18a présente les variations du seuil calculé par le *headstage* en fonction d’un SNR variant successivement de 2dB, 20db, 15dB, 10dB à 5db avec un *firing rate* constant de 100 AP/s. La figure 4.18b présente un signal reconstruit à partir des AP transmis par le *headstage* ainsi que le signal synthétique original ayant un SNR de 20 dB. On peut voir que le signal reconstruit est identique à l’original à l’exception du bruit entre les AP qui a été perdu lors de la compression.

### Temps d’exécution du détecteur

Le tableau 4.9 présente les temps d’exécution nécessaires afin de calculer le seuil d’un canal pour différents algorithmes exécutés dans le MSP430 fonctionnant à 8 MHz. Afin de ten-

TABLE 4.9 – Comparaison du temps d’exécution du MCU du *headstage* à deux canaux avec la méthode de détection proposée versus d’autres méthodes pour une fréquence de 8 MHz.

Méthode	Temps d’exécution [ $\mu$ s]
Méthode proposée (optimisée)	11
Méthode proposée (non optimisée)	440
RMS (optimisée)	20

nir compte d’éventuelles optimisations pouvant être apportées à l’algorithme de seuillage à base de la valeur RMS du signal, une version optimisée utilisant uniquement une multiplication, des additions et des opérations de décalage a été implémentée puis testée. Une version non optimisée de l’algorithme proposé qui utilise des nombres à virgules flottantes (*float*), des multiplications et des divisions a aussi été implémentée et testée. On peut constater que la méthode proposée optimisée réduit considérablement le temps d’exécution de 440% par rapport à sa version non optimisée, en plus de nécessiter deux fois moins de calcul que la méthode RMS optimisée. En fait, l’algorithme de seuillage proposé est le seul pouvant être exécutée parallèlement sur les deux canaux d’enregistrements du *headstage*, car chaque canal alloue un temps d’exécution maximal à l’algorithme de  $\sim 12.5 \mu$ s à une fréquence d’échantillonnage de 20 kHz.

## 4.6 Validation *in vivo*

Le système a été validé *in vivo* avec une souris transgénique exprimant le gène Channelrhodopsin-2 le 20 mars 2015 au CRIUSMQ. Le *headstage* connecté au module implantable utilisé dans les tests est présenté à la figure 4.19a. Le module implantable est constitué d’une DEL reliée à une fibre optique de 200  $\mu$ m de diamètre et de deux microélectrodes en Tungstène de 75  $\mu$ m (impédance de 1 M $\Omega$ ) provenant de chez *MicroProbes*. Ce module, conçu par *Doric Lenses* afin de valider le système, n’est pas destiné à une implantation permanente sur la souris. Le montage expérimental est présenté aux figures 4.19b, 4.19c et 4.19d. Il est à noter que les résultats reportés dans cette section ont été présentés dans une publication, voir [99].

### 4.6.1 Préparation de l’animal

La préparation de l’animal a été réalisée avec une souris adulte (50 g) anesthésiée avec de la kétamine/xylazine (10 :10 mg/kg) et placée dans un appareil stéréotaxique. La ligne de souris utilisée (ligne Thy1 : :ChR2-YFP 4) [109] a été achetée auprès de *The Jackson Laboratory*. Tout au long de l’expérience, la souris a été maintenue en anesthésie profonde et sa température corporelle a été maintenue à 37, 5°C. Pour que la sonde ait accès aux tissus du cerveau, une craniotomie a été effectuée et la *dura* a été retirée. La sonde a été abaissée à l’aide d’un micromanipulateur dans le cortex frontal, au-dessus de l’Hypocampe. Tous les protocoles

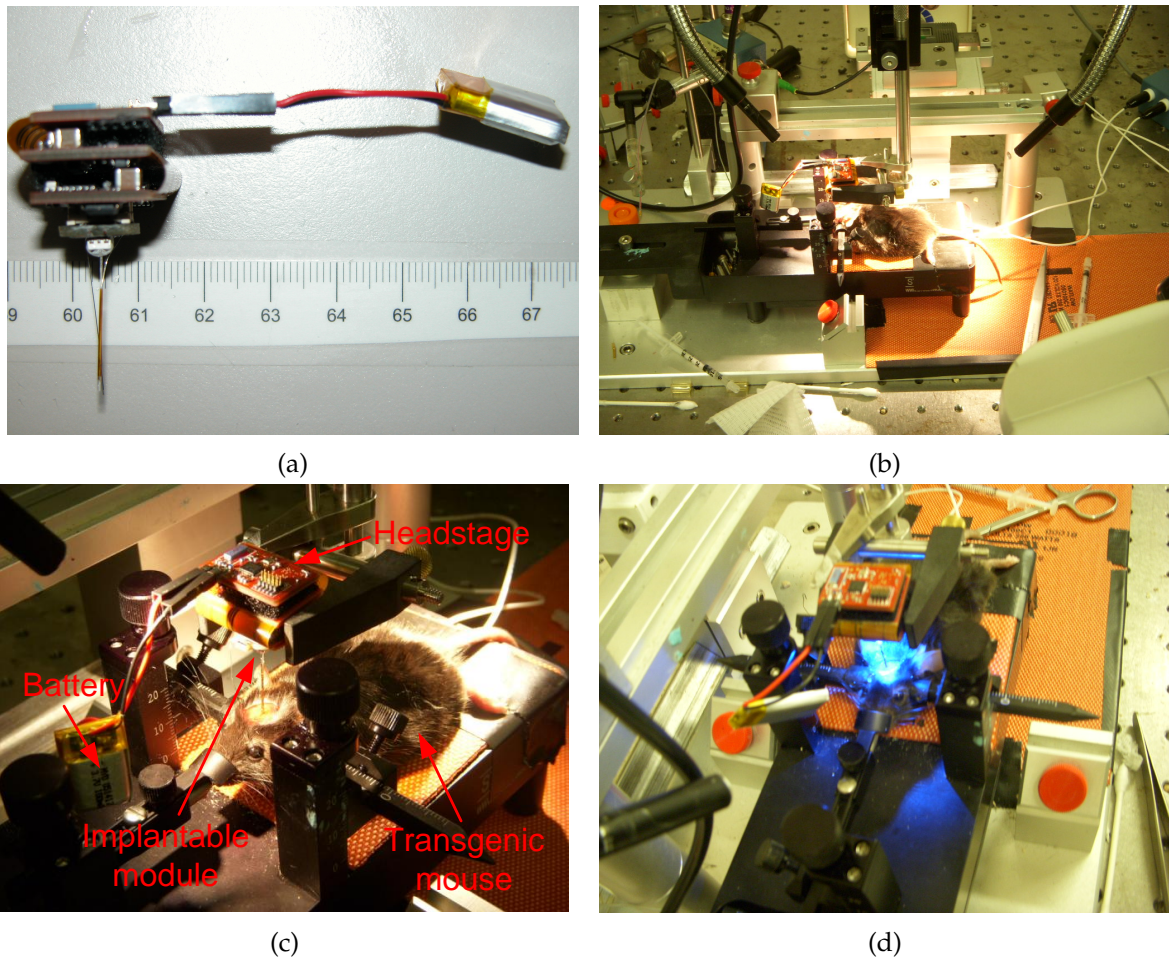


FIGURE 4.19 – a) *Headstage* avec le module implantable , b) montage expérimental, c) *headstage* installé sur la souris en fonctionnement, d) *headstage* installé sur la souris avec la DEL en fonctionnement.

ont été réalisés conformément aux lignes directrices du Conseil canadien de protection des animaux.

#### 4.6.2 Résultats

Il a été démontré que la *headstage* est capable de détecter les AP évoqués par activation lumineuse et ceux issus de décharges spontanées (figure 4.20 et 4.21). Le SNR du signal ainsi que la résolution d'enregistrement (12 bits/échantillons à 20 kHz) étaient suffisants pour obtenir des identifications unitaires claires. La figure 4.20a démontre que des AP ont été évoqués systématiquement par stimulation optique. Afin de démontrer que l'enregistrement n'est pas le résultat d'artefacts induits par l'exposition de l'électrode à la lumière [110], [111], différentes durées d'illumination (largeur d'impulsions entre 5 ms à 20 ms) ont été utilisées pour montrer que le signal ne varie pas en fonction du temps d'exposition à la lumière (figure 4.20c); les AP ont la même forme, ont été déclenchés après le même délai et possèdent

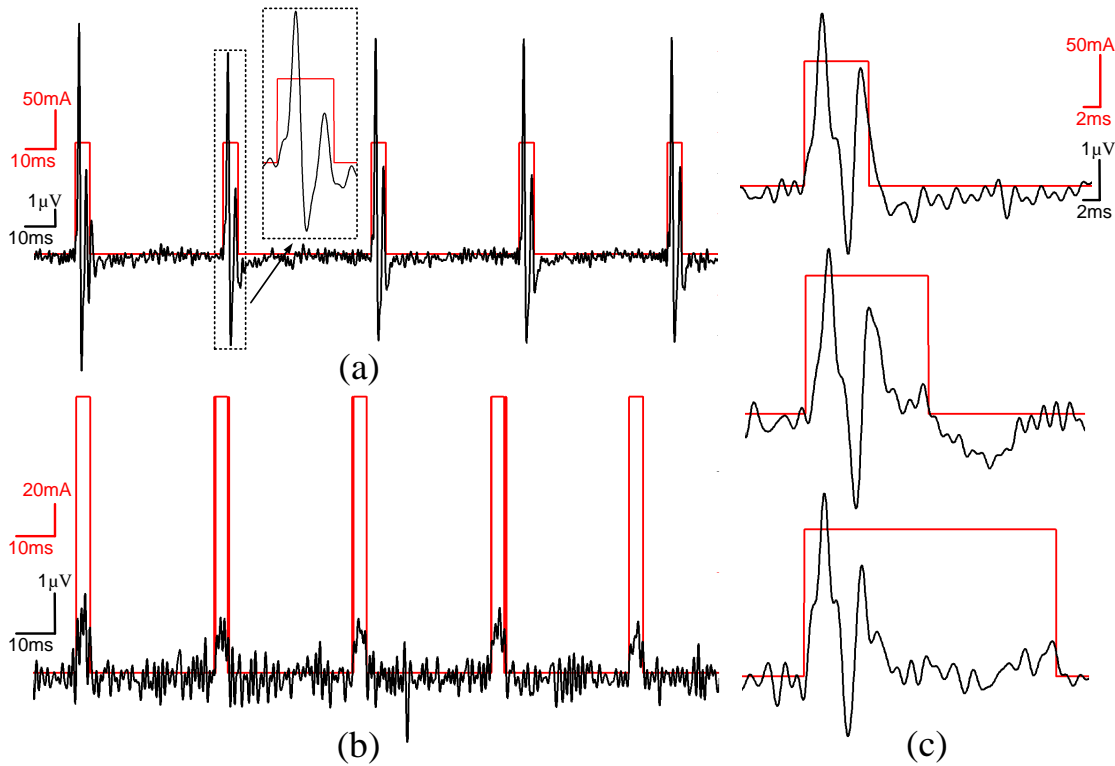


FIGURE 4.20 – a) Résultats expérimentaux montrant un enregistrement extracellulaire (en noir) en réponse à une activation lumineuse locale (courant dans de DEL en rouge), les signaux montrés proviennent du *headstage*, b) exemple d’enregistrement *in vivo* sans réponse électrophysiologique, c) exemple de réponse électrophysiologique en fonction de trois temps d’activation lumineux différents (5, 10, 20 ms).

tous des formes similaires, indépendantes de la stimulation lumineuse. Il est aussi démontré à la figure 4.20b qu’aucun AP ne se produit dans les régions du cerveau connues pour ne pas avoir de protéines de Channelrhodopsin. Cependant, il est possible de voir de petite bosse ( $1.5 \mu\text{V}$ ) corrélée aux séances d’éclairage. Ces fluctuations sont interprétées comme des artefacts induits par la lumière sur les électrodes et sont décrits dans la littérature [110], [111].

Le *headstage* permet de personnaliser les modes de stimulation optique afin d’être adaptable à différents systèmes biologiques, régions locales du cerveau, ou à des populations de cellules. La figure 4.21 montre un exemple où un patron de stimulation influence la réponse physiologique d’une cellule spécifique. Un train de cinq impulsions de 5 ms induit systématiquement des AP d’amplitude constante (figure 4.21b) par rapport à un train de cinq impulsions de 20 ms qui induit des AP dont l’amplitude décroît suite à un blocage de la dépolarisation [112] (figure 4.21c). Ce phénomène illustre bien une question biologique spécifique qui peut être abordée avec le système proposé.

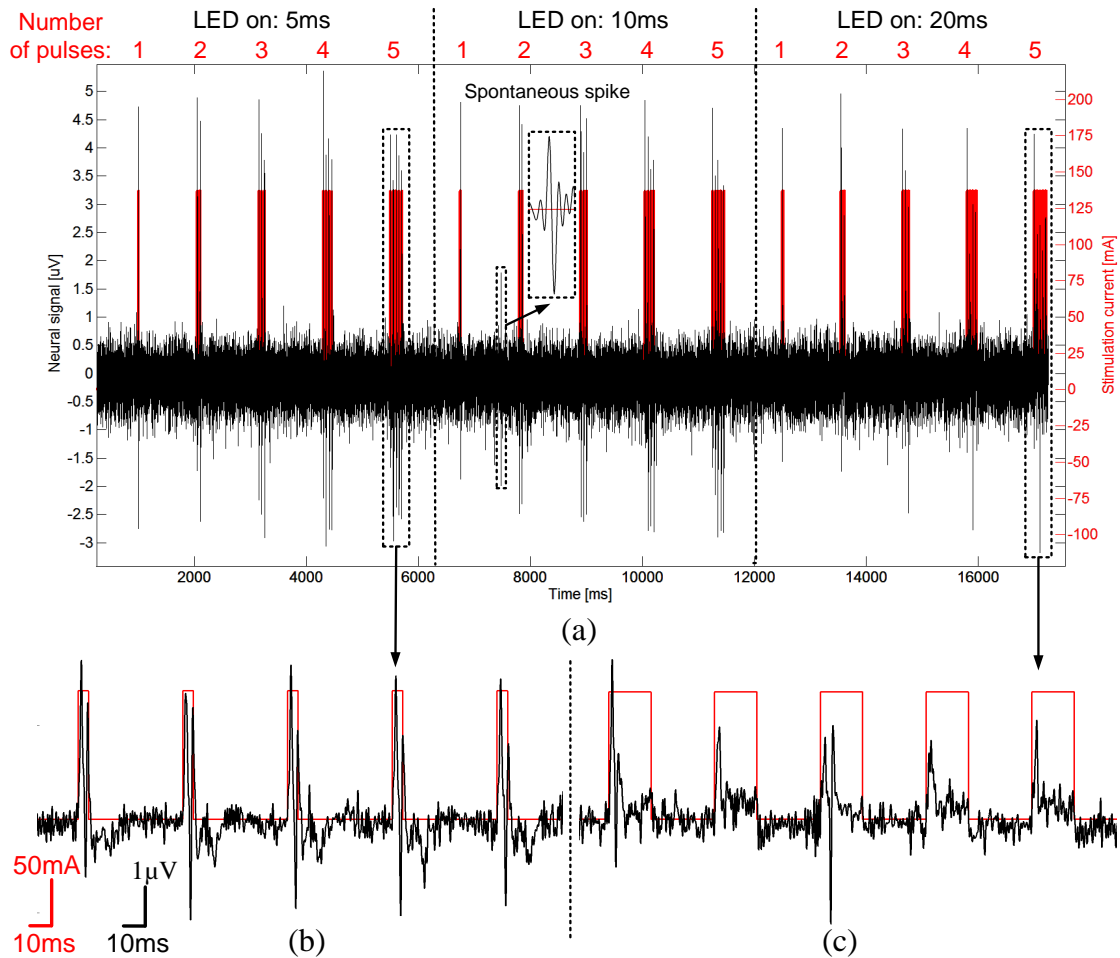


FIGURE 4.21 – Résultats expérimentaux montrant la réponse extracellulaire d’AP à trois séquences de stimulation (1 à 5 pulses) et à des temps d’exposition lumineux différents (5, 10, 20 ms), b) grossissement sur l’enregistrement extracellulaire face à 5 pulses lumineux de 5 ms de long chacun, c) grossissement sur l’enregistrement extracellulaire face à 5 pulses lumineux de 20 ms de long chacun, montrant des *AP failures*.

## 4.7 Système amélioré pour commercialisation

Grâce aux excellentes performances obtenues avec le *headstage* présenté dans de ce chapitre, une version pré-commerciale a été réalisée dans le cadre d’une entente Mitacs<sup>1</sup> [55] avec le partenaire industriel *Doric Lenses* et dont la commercialisation est prévue pour octobre 2015.

### 4.7.1 Améliorations

La version pré-commerciale du *headstage* est présentée à la figure 4.22 et diffère de la version présentée précédemment selon les points suivants :

- Système beaucoup plus léger, 1.1 g versus 4.9 g.

1. Le Mitacs s’est échelonné de février à juin 2015.

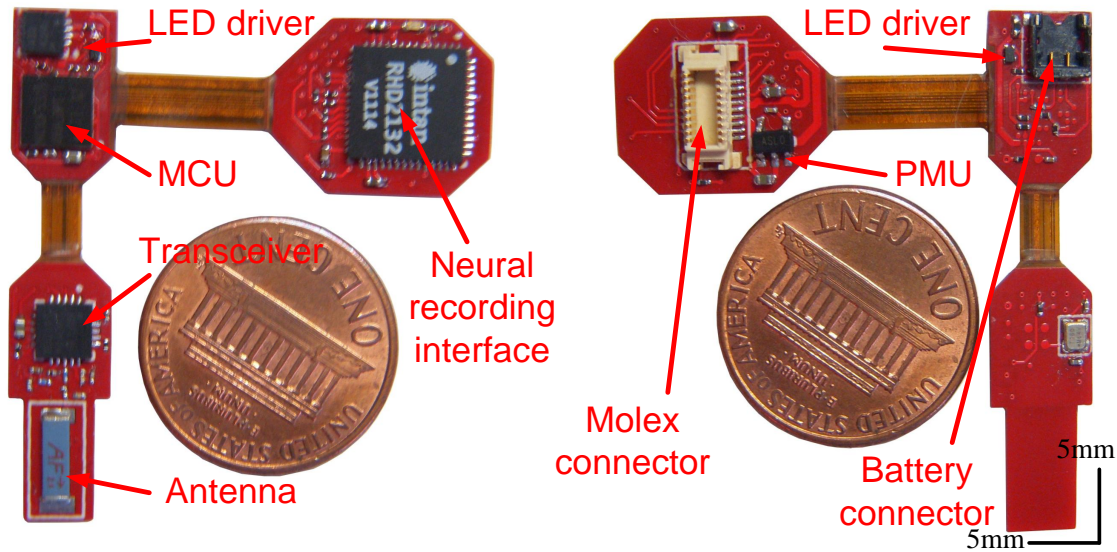


FIGURE 4.22 – Vue de chaque côté et non repliée du PCB flexible rigide optimisé en taille et en poids (1.1 g) avec chacune des parties électroniques clairement identifiées.

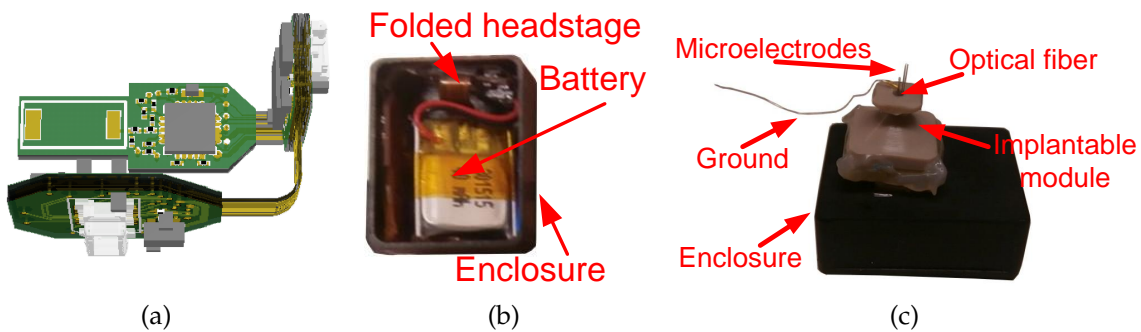


FIGURE 4.23 – a) Vue 3D du PCB flexible rigide replié dans sa position finale afin de laisser un espace pour la batterie au centre, b) *headstage* replié avec la batterie installée dans une enceinte protectrice, c) système complet avec le module implantable installé.

- Design optimisé permettant de replier le *headstage* autour de la batterie afin que le système soit plus compact.
- Utilisation d'un circuit intégré RHD2132 de *Intan Technologies* comme AFE afin de rendre le *headstage* plus compact et léger (cette puce sera décrite en détail à la section 5.2.1 du chapitre 5).
- Ajustabilité des fréquences de coupure des filtres analogiques (grâce à la puce RHD2132) à partir d'une interface utilisateur.

Une vue 3D du PCB flexible rigide replié est présentée à la figure 4.23a. Le système replié, connecté à la batterie et installé dans une enceinte protectrice est présenté à la figure 4.23b.



FIGURE 4.24 – a) Souris transgénique anesthésiée avec le module implantable (installé de façon permanente sur le crâne de l’animal) connecté au *headstage*, b) souris transgénique éveillée en pleine expérience d’enregistrement électrophysiologique avec le *headstage*.

#### 4.7.2 Validation *in vivo*

Le système commercialisable a été validé *in vivo* avec une souris transgénique exprimant le gène Channelrhodopsin-2 le 22 juillet 2015 au CRIUSMQ. Le *headstage* installé dans son enceinte protectrice et connecté au module implantable est présenté à la figure 4.23c. Le module implantable, conçu par *Doric Lenses*, est constitué d’une DEL reliée à une fibre optique de 200  $\mu\text{m}$  de diamètre ainsi que de deux microélectrodes en Tungstène de 75  $\mu\text{m}$  (impédance de 1 M $\Omega$ ) provenant de chez *MicroProbes*. La préparation de l’animal a été effectuée conformément à la procédure décrite à la section 4.6.1, à l’exception que le module a été installé de façon permanente sur le crâne souris avec de la colle *LOCTITE 454* et que les électrodes ont été insérées dans le cortex moteur du cerveau. Une fois le module fixé, la souris a été réveillée progressivement afin que l’expérience ait lieu sur un spécimen éveillé. Le système installé sur la souris est présenté aux figures 4.24a et 4.24b. Il est à noter que les résultats reportés dans cette section ont été présentés dans deux publications, voir [98] et [100].

L’algorithme de seuillage adaptatif présenté à la section 4.3.2 a été validé *in vivo*, en même temps que le système lui-même, en transmettant simultanément le seuil calculé par le *headstage* et le signal d’un canal spécifique. La figure 4.25 montre une séquence d’enregistrement de 50 secondes où l’on peut voir le seuil calculé par le *headstage* (en rouge) superposé au signal brut (en noir). Dans cette expérience, l’animal bougeait les pattes avant par lui-même en séquence de quelques secondes. Puisque les électrodes étaient insérées dans le cortex moteur du cerveau, des rafales d’AP additionnées à du bruit neuronal et à du bruit engendré par le mouvement de l’animal (léger mouvement des électrodes) se sont produites à chaque

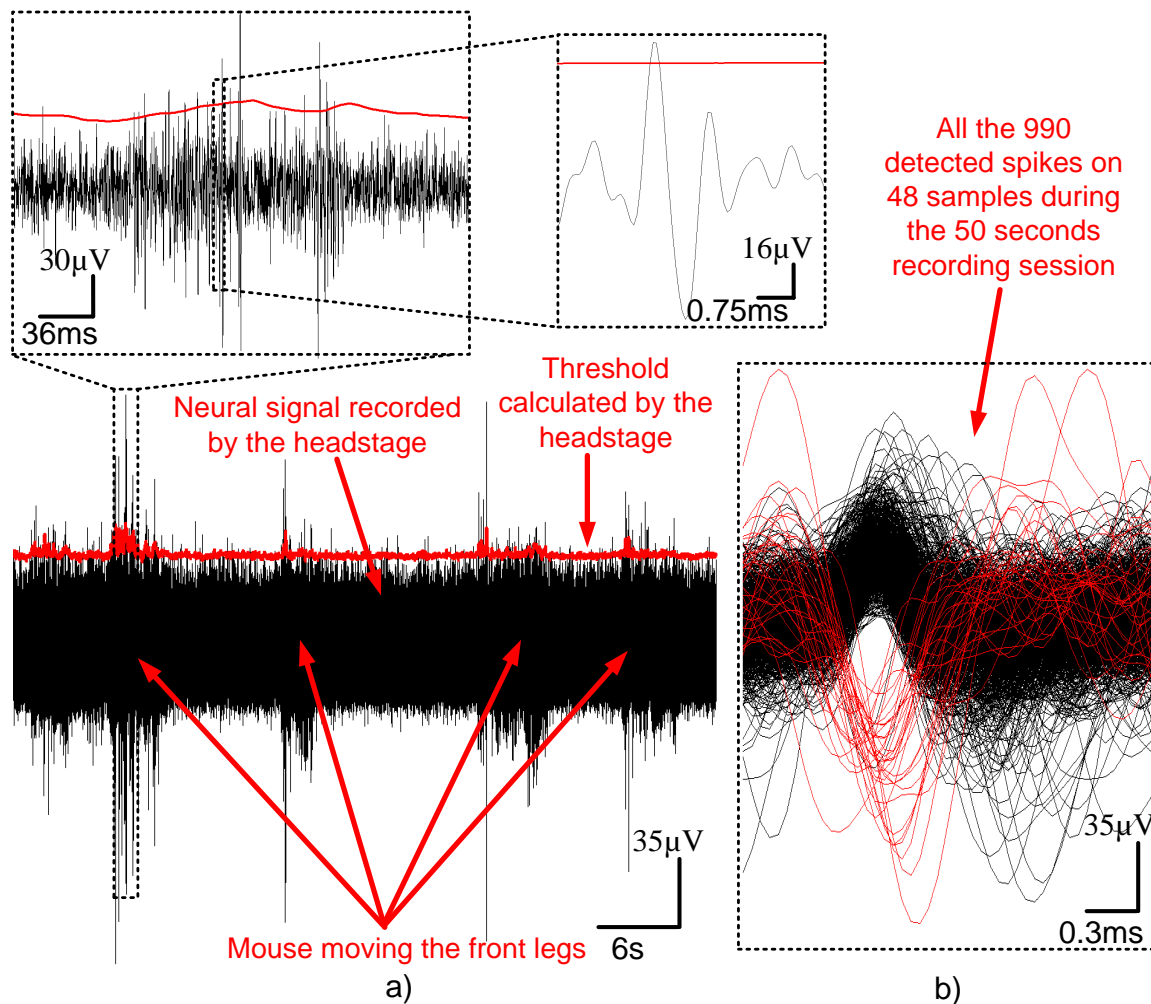


FIGURE 4.25 – a) Seuil calculé par le *headstage* (en rouge) superposé au signal brut (en noir) lors d’une expérience d’enregistrement électrophysiologique d’une durée de 50 secondes, b) AP détectés par le *headstage*, classés puis superposés.

mouvement des pattes avant. Comme la figure 4.25 le montre, le seuil s’est adapté à ces variations brusques du niveau de bruit tout en détectant avec précision les AP produits par les neurones proches de la microélectrode.

Afin de quantifier les performances de détection du *headstage* en condition *in vivo* versus d’autres méthodes, les signaux bruts ont été importés dans *Matlab* pour les utiliser avec l’algorithme **RMS** et **MAD**. Ces deux algorithmes ont été implémentés suivant les mêmes paramètres que ceux présentés à la section 4.5.2. Les résultats de 8 expériences sont présentés dans le tableau 4.10. On peut voir que l’algorithme **RMS** a détecté beaucoup moins d’AP que la méthode proposée dans toutes les expériences. En fait, ces résultats concordent avec les résultats *in vitro* présentés à la section 4.5.2 pour un signal neuronal ayant un **SNR** faible. L’algorithme **MAD** a généré des résultats très similaires à la méthode proposée lorsque l’ani-



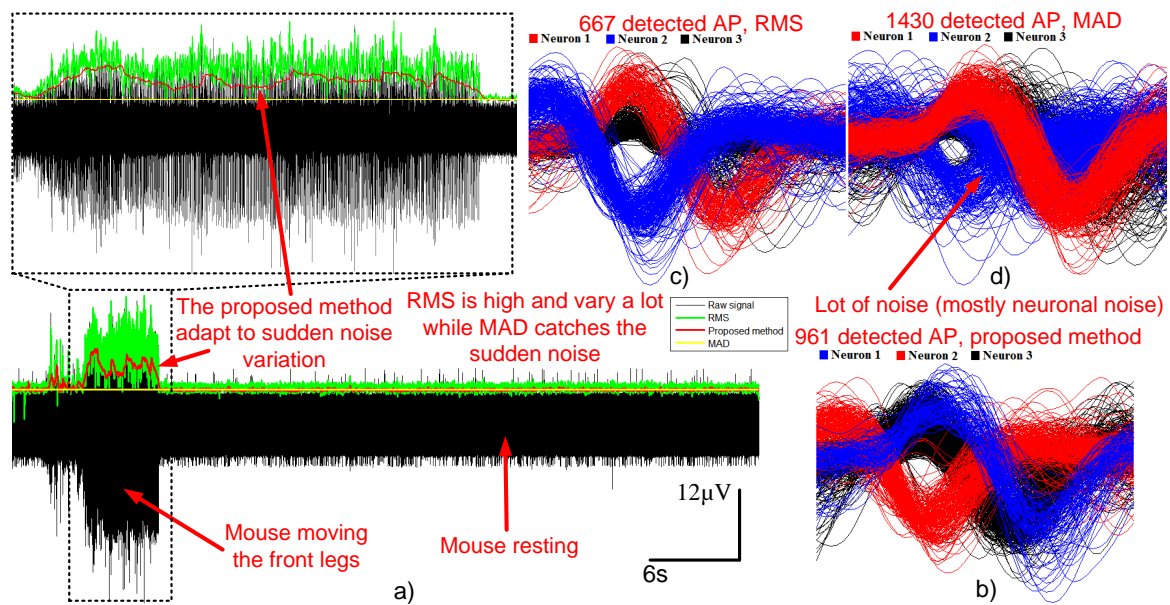


FIGURE 4.26 – a) Seuil calculé par le *headstage* (en rouge), seuil calculé par la méthode RMS (en vert) et MAD (en jaune) superposé au signal brut (en noir), b) AP détectés par le *headstage*, c) AP détectés par seuillage RMS, d) AP détectés par seuillage MAD

TABLE 4.10 – Résultats de détection du *headstage* à deux canaux commercial pour 8 séances d’enregistrement électrophysiologique *in vivo*, comparés aux résultats obtenus avec les méthodes de seuillage RMS et MAD exécutées sur les signaux bruts importés dans *Matlab*.

Durée de l’expérience [s]	Micro-électrode	Comportement de l’animal	AP détectés par le <i>headstage</i>	AP détectés par la méthode RMS	AP détectés par la méthode MAD
50	1	Pattes avant en mouvement	1243	841	1912
50	1	Pattes avant en mouvement	990	461	1304
50	1	Pattes avant en mouvement	961	667	1430
30	1	Animal au repos	542	274	570
50	2	Pattes avant en mouvement	3063	927	4972
50	2	Pattes avant en mouvement	3008	1557	3447
50	2	Animal au repos	2136	754	2358
50	2	Animal au repos	1980	757	2069

mal était en repos. Les résultats divergent lors des expériences où l’animal bougeait les pattes avant. Dans cette situation, la méthode MAD a détecté beaucoup plus d’AP que la méthode proposée, et ce, puisque la méthode MAD n’est pas adaptative localement. Tel qu’il a été expliqué précédemment, l’algorithme MAD estime l’écart-type du signal en entier, ce qui l’empêche de s’adapter aux variations brusques du bruit sur une courte période. La majeure partie des formes d’ondes détectées lorsque l’animal bougeait les pattes avant sont du bruit (bruit neuronal ou autre) et non pas des AP. Il est à noter que la détection des AP par le *headstage* a permis d’obtenir un CR moyen de 11.4 lors les expériences présentées dans le tableau 4.10.

La figure 4.26 montre le comportement du seuil calculé par le *headstage* (en rouge), calculé par la méthode **RMS** (en vert) et la méthode **MAD** (en jaune). On voit clairement dans cette figure que le seuil calculé par la méthode **RMS** est trop haut et varie trop brusquement pour être précis. Le seuil calculé par la méthode proposée s'ajuste parfaitement au bruit comparativement au seuil calculé par la méthode **MAD** qui est constant. Il est important de noter que le seuil calculé par la méthode proposée est complètement superposé au seuil calculé par la méthode **MAD** lorsque l'animal est au repos, ce qui démontre que l'écart-type estimé par le *headstage* est équivalent à celui calculé par la méthode **MAD** sur le signal en entier.

## Chapitre 5

# Systeme optogénétique sans fil à 32 canaux avec compression et détection en temps réel des AP

### 5.1 Présentation du projet

Ce chapitre présente la conception d'un *headstage* optogénétique miniature sans fil doté de 32 canaux d'enregistrement et de 32 canaux de stimulation optique. Le *headstage* peut détecter, compresser et transmettre les signaux électrophysiologiques de 32 canaux en parallèle grâce à de nouveaux algorithmes implémentés dans un FPGA intégrant un processeur *Microblaze*.

Le *headstage* décrit dans ce chapitre est conçu pour effectuer des expériences optogénétiques *in vivo* de longues durées sur des rongeurs transgéniques évoluant en toute liberté. Il diffère des deux systèmes présentés au chapitre 4 en possédant plus de canaux d'enregistrement/stimulation et en étant plus petit et plus léger que la version non commerciale. Or, ce nouveau système augmente significativement le nombre de canaux pouvant être transmis ( $\times 16$ ) en détectant en temps réel les AP (méthode de compression présentée à la section 2.1) et en compressant les AP détectés via une méthode de compression par ondelettes de type *Symmlet-2* (méthode de compression présentée à la section 2.2). De plus, le *headstage* ainsi que les algorithmes et résultats présentés ont fait l'objet de plusieurs publications scientifiques, voir [113], [108], et [114].

Le système innove en exécutant parallèlement sur 32 canaux la méthode de seuillage adaptative décrite à la section 4.3.2 et en utilisant une nouvelle méthode de compression basée sur le seuillage adaptatif des coefficients de la DWT et sur une maximisation de la plage dynamique des coefficients conservés (taux de compression de 508 atteints lors des tests *in vivo*).

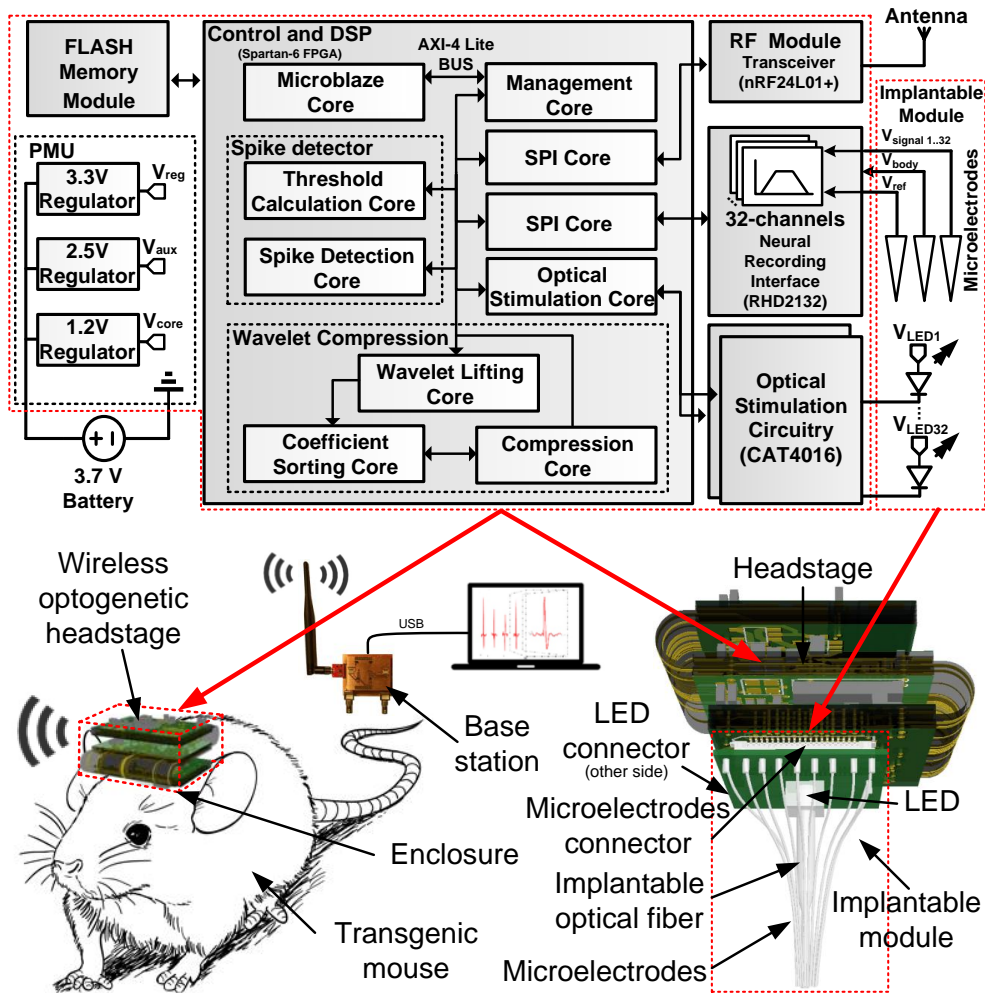


FIGURE 5.1 – Concept et diagramme en bloc du *headstage* optogénétique à 32 canaux.

Le *headstage* est construit entièrement de composants COTS et sa conception miniature, basée sur un PCB flexible rigide, produit un système léger (2.8 g) et très compact ( $17 \times 18 \times 10 \text{ mm}^3$ ). Le système a été validé *in vitro*, avec des signaux neuronaux réels et synthétiques, et *in vivo* avec une souris transgénique exprimant le gène Channelrhodopsin-2.

D'abord, la conception matérielle du *headstage* est présentée suivi de la description de l'algorithme de compression et de son implémentation optimisée. Ensuite, l'algorithme de détection des AP et son implémentation optimisée sont présentés. Les résultats *in vitro* et *in vivo* obtenus avec le système font l'objet du chapitre 6.

## 5.2 Conception du *headstage*

Le système optogénétique est constitué de deux parties reliées entre elles via deux connecteurs, un pour les DEL et un pour les microélectrodes. La première partie est le *headstage* lui-

même, qui inclut l'électronique nécessaire à l'acquisition des signaux neuronaux, au contrôle du système, à la détection des AP, à la compression des AP, à la stimulation optique et à la transmission sans fil des données. La seconde partie est le module implantable, qui contient les DEL haute puissance, les fibres optiques ainsi que les microélectrodes. La figure 5.1 montre le concept et un schéma-bloc du système.

Le *headstage* est constitué de cinq parties distinctes : 1) une interface d'acquisition neuronale, 2) un circuit de stimulation optique, 3) une unité de gestion de l'énergie (PMU), 4) un transmetteur sans fil et 5) un puissant processeur FPGA à faible consommation énergétique. La tâche de l'interface d'acquisition neuronale est de conditionner et de quantifier les signaux neuronaux de faible amplitude présents aux bornes des microélectrodes. Le rôle du circuit de stimulation optique consiste à appliquer un patron de tension déterminé sur les DEL. Le PMU est chargé de générer une tension stable de 3.3 V à l'interface d'acquisition neuronale, au FPGA, au transmetteur sans fil ainsi qu'à la mémoire non volatile externe du FPGA. Le PMU génère aussi une tension stable de 2.5 V aux broches auxiliaires du FPGA et une tension stable de 1.2 V au *core* du FPGA. Un transmetteur sans fil de 2.4 GHz est utilisé pour transmettre les données à une station de base et pour recevoir les paramètres de configuration de l'expérience. Le contrôle du système, la détection des AP, la compression par ondelettes des AP, la génération des patrons de stimulation et la communication entre tous les modules sont assurés par un FPGA Spartan-6 de *Xilinx* qui est programmé au démarrage du système par un module de mémoire non volatile externe de 16 Moctets. Chacun des sous-systèmes susmentionnés est passé en revue dans cette section.

### 5.2.1 Circuit d'acquisition neuronal

L'interface d'acquisition neuronale est responsable du conditionnement et de la quantification des signaux électrophysiologiques de 32 microélectrodes ayant une amplitude très faible. Extraire un si grand nombre de signaux de faible amplitude nécessite l'utilisation d'un circuit intégré. Pour ces raisons, le *headstage* utilise la puce d'enregistrement électrophysiologique RHD2132 de *Intan Technologies* [115], disponible dans un *package* QFM très compact (8 mm x 8 mm) et possédant un faible bruit référé à l'entrée et une grande linéarité. La puce RHD2132 peut échantillonner les signaux neuronaux de 32 microélectrodes avec une précision de 16 bits à 30 kHz maximum, dont l'amplitude peut varier entre -5 mV et 5 mV et possède un bruit référé à l'entrée de  $2.4 \mu V_{rms}$ .

La puce RHD2132 est totalement reconfigurable par lien SPI, ce qui permet d'ajuster les fréquences de coupures des filtres passe-bande en temps réel selon le niveau de bruit et le type de signal à enregistrer (AP ou LFP). Le filtre passe-bande de la puce RHD2132 est constitué d'un filtre passe-haut du premier ordre, dont la fréquence de coupure est configurable entre 0.1 Hz et 500 Hz et d'un filtre passe-bas de type Butterworth du 3<sup>ème</sup> ordre dont la fréquence de coupure est configurable entre 0.1 kHz et 20 kHz.

### 5.2.2 Circuit de stimulation optique

Le circuit de stimulation optique est conçu pour fournir en courant 32 DEL. Une DEL optogénétique conventionnelle doit délivrer une puissance optique de  $10 \text{ mW/mm}^2$  à proximité des neurones sensibles à la lumière afin de générer une réaction, ce qui nécessite une grande quantité de courant [116] [102]. Un courant si élevé produit une chute de tension aux bornes de la DEL proche de la tension d'une batterie LiPo de 3.7 V (3.3V @150mA [102]).

Pour cette raison, le circuit de stimulation optique a été conçu à partir de deux puces CAT4016 de *ON Semiconductor* possédant une faible chute de tension interne (0.4 V pour des courants variant de 2 mA à 150 mA) et disponible dans un *package* miniature TQFN (4 mm x 4 mm). Le courant fourni à chacune des DEL peut être ajusté en modifiant deux résistances, une résistance pour chaque puce, ce qui permet des patrons de stimulation d'intensité variable. Des mesures empiriques ont démontré que le circuit de stimulation optique peut délivrer 150 mA par puce sans hausse significative de température ( $\uparrow 2^\circ\text{C}$  @100 Hz, 10%) ce qui permet, par exemple, de délivrer 150 mA à deux DEL hautes puissances ou bien 9.4 mA simultanément à 32 DEL. Un courant de 150 mA permet de générer une puissance de  $250 \text{ mW/mm}^2$  à la sortie d'une fibre optique parfaitement couplée à une DEL [102], ce qui est de loin supérieur au seuil d'activation des cellules sensibles à la lumière [117].

Les patrons de stimulation optique (PWM) sont configurables via un protocole sériel (protocole *custom* de la puce CAT4016) contrôlé par le *Microblaze*, permettant de configurer à la volée la fréquence et le *duty cycle* des pulses de lumière.

### 5.2.3 Unité de gestion de l'énergie

Le *headstage* est alimenté par une batterie 100 mAh LiPo de 3.7 V. Le PMU doit générer une tension stable de 3.3 V pour le FPGA, pour l'interface d'enregistrement neuronal, pour le transmetteur sans fil et pour la mémoire externe. De plus, le PMU doit générer une tension stable de 2.5 V aux broches auxiliaires du FPGA et une tension stable de 1.2 V au *core* du FPGA. Un régulateur *low-drop* TLV70233 de *Texas Instruments* est utilisé afin de générer la tension de 3.3 V. Ce régulateur possède un haut PSRR de 68 dB @ 1 kHz afin de diminuer les fluctuations de tensions produites par le circuit de stimulation optique sur l'interface d'enregistrement neuronal. Les tensions d'alimentation de 2.5 V et de 1.2 V sont assurées par un régulateur de tension haute précision LP3990 de *Texas Instruments*. Tous les circuits intégrés du PMU utilisent un *package* SOT-23 (3 mm x 3 mm) afin de minimiser la taille et le poids.

### 5.2.4 Processeur de traitement de signal

Le contrôle du système, la détection des AP, la compression par ondelettes des AP, la génération des patrons de stimulation et les communications entre tous les modules sont effectués

par un FPGA Spartan-6 LX16 de *Xilinx*. Le FPGA LX16 possède les ressources logiques nécessaires pour cette application (14,579 cellules logiques et 32 blocs de RAM) et son petit *package* CPG196 (8 mm x 8 mm) de type *ball grid array* (BGA) permet une miniaturisation importante. De plus, ce FPGA possède une faible consommation énergétique grâce à sa gravure de 45 nm optimisée pour consommer peu. Un processeur logiciel *Microblaze* de *Xilinx* est implémenté dans le *Spartan-6* afin de contrôler les modules du *headstage* (voir la figure 5.1) et de faciliter le développement du *firmware* de contrôle.

Le circuit du FPGA est constitué de plusieurs modules : 1) un module *Microblaze* 2) un module de détection des AP, 3) un module de seuillage automatique, 4) un module de calcul de la DWT par *lifting*, 5) un module de tri des coefficients de la DWT, 6) un module de compression, 7) un module de génération des patrons de stimulation optique, 8) deux modules SPI pour communiquer avec l'interface d'enregistrement neuronal et avec le transmetteur sans fil et 9) un module de gestion des données qui a été ajouté pour faciliter les communications entre le *Microblaze* et les autres modules via un BUS AXI-4 Lite de *Xilinx*. En plus de contrôler tous les modules, le *Microblaze* gère les communications avec le transmetteur sans fil, en plus de fournir la configuration des registres et du filtre passe-bande de l'interface d'enregistrement neuronal. Le programme du *Microblaze* est développé en langage C alors que tous les modules sont développés en langage VHDL.

### 5.2.5 Système de transmission dans fil

Le transmetteur sans fil est le même que celui utilisé dans les *headstages* à deux canaux dont la description est disponible à la section 4.2.3 du chapitre 4.

### 5.2.6 PCB

La taille et le poids sont des caractéristiques importantes du *headstage*. Toutes les composantes du *headstage* sont assemblées sur un PCB flexible rigide possédant 3 sections rigides et 2 sections flexibles. La section supérieure du PCB est dédiée au transmetteur et au PMU, la section centrale est dédiée au FPGA et au circuit de stimulation optique et la section inférieure supporte l'interface d'enregistrement neuronal et les deux connecteurs *Molex* afin de connecter le *headstage* avec le module implantable. Les sections rigides sont connectées entre elles via des sections flexibles où il n'y a aucun *power planes*, résultant en une meilleure isolation entre les sections. Le design du PCB est constitué de 6 couches avec une largeur minimale des connexions de 100  $\mu\text{m}$  et une taille minimale des vias de 200  $\mu\text{m}$ . La figure 5.2a présente le circuit de conception du PCB, témoignant de sa grande complexité, tandis qu'un modèle 3D du PCB replié et connecté à un module implantable est montré à la figure 5.2b.

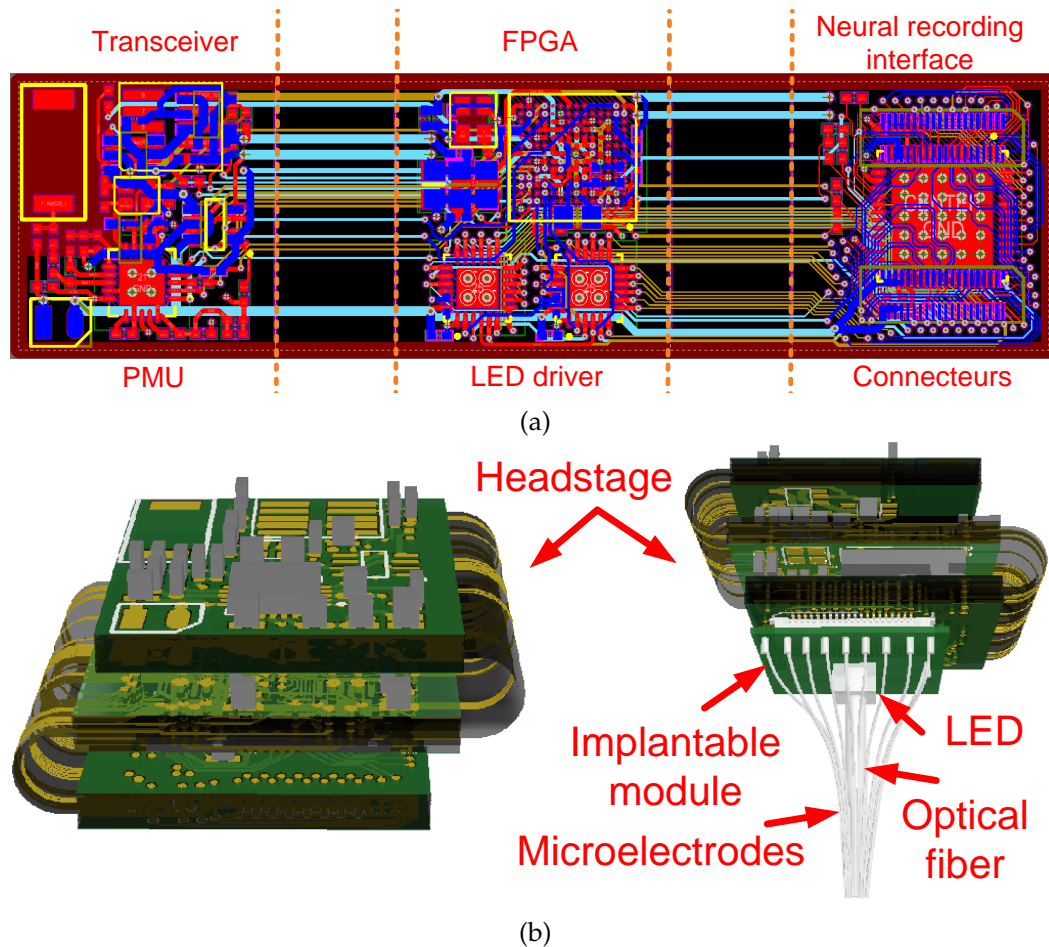


FIGURE 5.2 – a) Design du PCB flexible rigide, b) vue 3D du PCB flexible rigide déplié, c) vue 3D du PCB flexible rigide replié et attaché à un module implantable montrant une DEL couplée à une fibre optique et 9 microélectrodes.

### 5.3 Algorithme modifié de compression basé sur les ondelettes

Afin d'augmenter significativement le nombre de canaux d'enregistrement, le *headstage* comprime les AP détectés en calculant une DWT de 4 niveaux sur les 48 échantillons de chaque AP. La compression est effectuée en seuillant les coefficients générés par la DWT, de façon à conserver seulement ceux possédant la plus grande énergie et en requantifiant dynamiquement les coefficients conservés de façon à maximiser la plage binaire utilisée. La nouvelle technique de compression développée permet une reconstruction des AP avec la même résolution que la forme d'onde originale, et ce, même pour un taux de compression élevé.

La sélection de la *mother wavelet* a un impact majeur sur la qualité des signaux reconstruits et sur les CR pouvant être atteints [75]. Tel qu'il a été démontré au chapitre 3, l'ondelette de type *Symmlet-2* a produit les meilleurs résultats parmi toutes les méthodes de compression testées. Ce résultat concorde avec ceux présentés dans [75], dans lequel il est démontré que



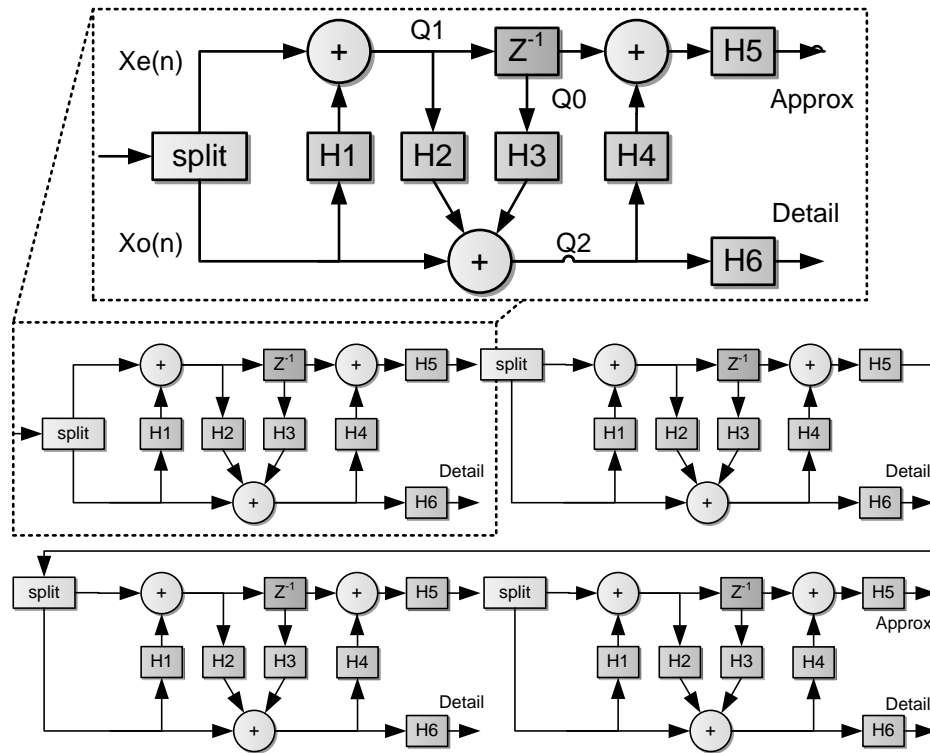


FIGURE 5.3 – Représentation schématique d’une transformée en ondelettes de type *Symmlet-2* de 4 niveaux de profondeur par la méthode de *lifting*.

les ondelettes de type *Symmlet* permettent de générer une compression presque optimale des signaux neuronaux. En fait, une *mother wavelet* possédant une forme similaire aux AP génère moins de coefficients significatifs, ce qui est le cas de l’ondelette *Symmlet-2*. Dans le but de réduire l’empreinte du circuit calculant la DWT dans le FPGA, les AP détectés par le module de détection sont multiplexés vers un seul module de calcul séquentiel de la DWT par *lifting*.

Avec une fréquence d’échantillonnage de 20 kHz et la possibilité de détecter 32 AP au même moment, le temps de compression de chacun de AP doit être  $\leq 75 \mu s$  ( $48 \text{ échantillons} / (20000 \text{ échantillons/s} \times 32 \text{ canaux})$ ) afin d’éviter tout débordement de mémoire. Une fréquence de 20 MHz a été retenue pour alimenter le module de calcul de la DWT par *lifting*, le module de tri des coefficients et le module de compression. Cette fréquence est un compromis entre la vitesse de compression et la consommation énergétique, générant un maximum de 1500 cycles d’horloges par compression d’AP.

### 5.3.1 Implémentation séquentielle de l’ondelette *Symmlet-2* par *lifting*

La figure 5.3 présente le schéma d’une transformée en ondelettes *Symmlet-2* de 4 niveaux par *lifting*.  $X_e(n)$  et  $X_o(n)$  sont les échantillons d’indices pairs et impairs respectivement du signal d’entrée, tandis que  $Q_0$ ,  $Q_1$  et  $Q_2$  sont des résultats intermédiaires. Les étapes de *lifting*

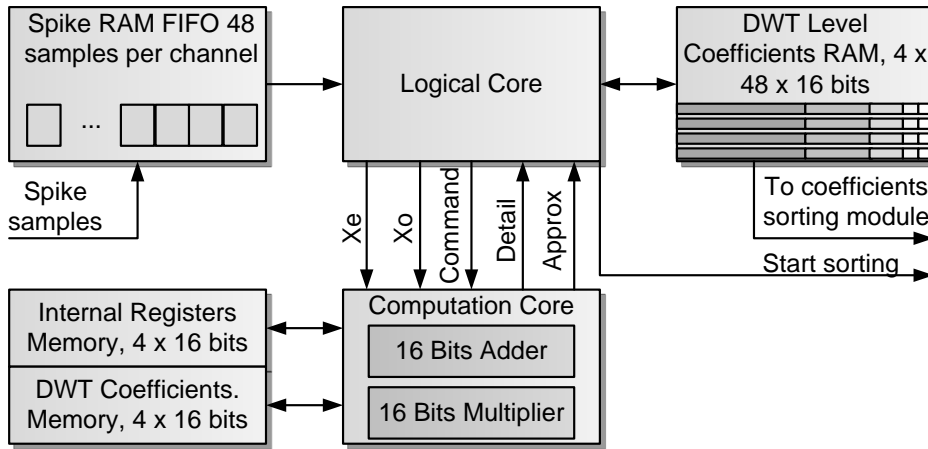


FIGURE 5.4 – Diagramme système de l’implantation séquentielle de la transformée en ondelettes de type *Symmlet-2* par *lifting*.

sont les suivantes :

- $Q_0 = X_0(n - 1) + H1 \times X_e(n - 1)$
- $Q_1 = X_0(n) + H1 \times X_e(n)$
- $Q_2 = X_e(n) + H3 \times Q_0 + H2 \times Q_1$
- $Approx = H5 \times (Q_0 + H4 \times Q_2)$
- $Detail = H6 \times Q_2$

Où  $H1$  à  $H4$  sont les quatre coefficients constants du schéma de *lifting* de type *Symmlet-2*,  $\sqrt{3}$ ,  $\frac{\sqrt{3}}{4}$ ,  $\frac{\sqrt{3}-2}{4}$  et  $-1$  respectivement, tandis que  $H5$  et  $H6$  sont les facteurs d’échelle de valeur  $\frac{\sqrt{3}-1}{\sqrt{2}}$  et  $\frac{\sqrt{3}+1}{\sqrt{2}}$  respectivement.

Le diagramme système de l’implémentation séquentielle de la transformée en ondelettes *Symmlet-2* est présenté à la figure 5.4. Pour chacun des niveaux de la transformée, 7 multiplications sont nécessaires. Avec une décomposition à 4 niveaux et une implémentation de type pipeline, il en résulte l’utilisation de 28 circuits de multiplication dédiée, ce qui est très prohibitif lors de l’utilisation d’un FPGA Spartan-6 à ressources limitées. Pour cette raison, l’implémentation séquentielle proposée utilise un seul circuit de multiplication qui tire avantage de la vitesse modérée d’acquisition des données neuronales. Toutes les opérations de la DWT sont contrôlées par un module logique qui gère les échantillons entrants, les étapes de calcul et le stockage des coefficients calculés.

Dans un premier temps, les 48 échantillons provenant du module de détection sont stockés en mémoire dans une structure *RAM FIFO IP Core* de *Xilinx*. Lorsque plus de deux échantillons sont disponibles dans la *FIFO*, le module logique les récupère sous forme d’échantillons d’index pair et impair. Les deux échantillons d’index pair et impair sont ensuite transmis à un module de calcul avec une commande indiquant l’opération à effectuer. C’est le

module logique qui gère ensuite toutes les étapes de calcul du *lifting* en transmettant des commandes au module de calcul. Le module de calcul détient dans ses registres internes les constantes du *lifting* *Symmlet-2* ainsi que le facteur d'échelle (4 valeurs sur 16 bits, à l'exception du coefficient  $-1$ ) et possède des registres internes pour stocker les résultats intermédiaires  $Q1$ ,  $Q2$ ,  $Q3$  et  $Q4$ . Une fois toutes les opérations terminées, le module logique récupère les coefficients approximatifs et détaillés produits par le module de calcul et les stocke dans 4 mémoires RAM (voir section 5.3.2). Si plus de 2 coefficients approximatifs sont disponibles dans la mémoire RAM pour le niveau courant, les mêmes opérations sont répétées par le module logique, mais en utilisant les coefficients approximatifs d'index pairs et impairs pour calculer le prochain niveau jusqu'au niveau 4. La latence totale du module de calcul de la DWT est de 456 cycles d'horloge.

### 5.3.2 Calcul du seuil des coefficients

Une fois que les 48 coefficients de la transformée sont calculés et stockés en mémoire RAM, ils sont traités par un module de tri. La tâche de ce module consiste à classer les coefficients par amplitude afin de trouver l'amplitude du  $x^{ieme}$  coefficient trié qui sera utilisé pour seuiller les coefficients dans le module de compression, où  $x$  est le nombre de coefficients à conserver. Au lieu de trier les coefficients un à un, la stratégie adoptée consiste à compter le nombre de coefficients possédant une amplitude plus grande que le coefficient courant et à répéter cette opération pour tous les coefficients. Lorsque le compte égal  $x$ , le coefficient courant est identifié comme étant le seuil des coefficients, soit le  $x^{ieme}$  coefficient trié. Afin d'augmenter la vitesse d'exécution de l'algorithme, le coefficient courant est comparé simultanément à 4 coefficients à chaque cycle d'horloge. Ces comparaisons simultanées sont possibles par l'utilisation de 4 mémoires RAM distinctes stockant les coefficients de la transformée et permettant d'effectuer 4 lectures en mémoires RAM par cycle d'horloge. Pendant l'exécution de l'algorithme, le coefficient possédant la plus grande amplitude est trouvé et sa valeur est transférée au module de compression en même temps que la valeur du seuil des coefficients. La latence totale du module de tri est de 576 cycles d'horloges, soit 48 coefficients comparés à 12 blocs de 4 coefficients.

### 5.3.3 Compression des données

Une fois que le coefficient trié possédant la  $x^{ieme}$  amplitude et le coefficient d'amplitude maximale sont trouvés, la compression est effectuée par le module de compression. La stratégie de compression des AP consiste à seuiller les coefficients de la DWT pour ensuite requantifier dynamiquement les coefficients conservés sur un nombre de bits moindre. Premièrement, à l'étape de seuillage, l'amplitude de chacun des coefficients est comparée avec l'amplitude du  $x^{ieme}$  coefficient. Si l'amplitude du coefficient courant est supérieure à l'amplitude du  $x^{ieme}$  coefficient, un tableau de 48 bits est mis à jour avec un 1 à la position du coefficient courant, sinon le tableau est mis à jour avec un 0 (voir la figure 5.5). Une fois que

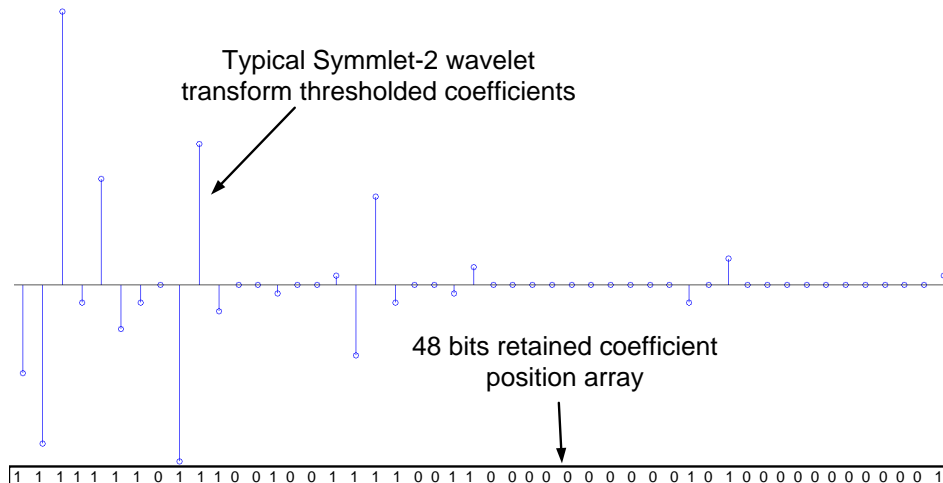


FIGURE 5.5 – Coefficients obtenus après une transformée en ondelettes *Symmlet-2* de niveau 4 et seuillés afin de conserver 20 coefficients sur 48.

tous les coefficients ont été traités, le tableau contient la position ordonnée des coefficients conservés et doit absolument être transmis avec les données afin de repositionner les coefficients à leur position initiale avant la reconstruction. Autrement, même si les coefficients sont transmis dans le bon ordre, leur position exacte est incertaine.

Un taux de compression plus élevé est obtenu en réduisant le nombre de bits de quantification des coefficients conservés. L'approche proposée est plus complexe que de seulement conserver les bits d'ordre plus élevé, ce qui résulterait en une perte significative d'information. Par exemple, si un coefficient possède la valeur  $0x00FF$  (255) et qu'il est requantifié sur 8 bits, toute l'information relative à ce coefficient est perdue. La solution proposée consiste à étendre dynamiquement la valeur des coefficients de façon à ce qu'ils utilisent le maximum de bits avant de les requantifier. Une façon simple d'y arriver est de décaler les bits de tous les coefficients vers la gauche jusqu'à ce que le **MSB** du coefficient d'amplitude maximale atteigne la  $16^{ième}$  position. Cette technique à l'avantage d'être simple, mais elle n'est pas optimale, surtout lorsqu'un taux de compression élevé est désiré. Par exemple, avec une amplitude maximale des coefficients de  $0x1000$  (4096), un coefficient possédant la valeur  $0x0010$  (16) va perdre toute son information lorsque requantifié sur 8 bits. Ce problème peut être résolu avec quelques calculs supplémentaires. Une façon d'y arriver consiste à diviser chacun des coefficients par l'amplitude maximale de tous les coefficients et de requantifier sur la mantisse du résultat. Dans un *FPGA Spartan-6*, des valeurs fractionnaires peuvent être calculées en représentant chaque coefficient de 16 bits sur 32 bits, avec les 16 bits originaux aux positions **MSB** et en effectuant une division entière avec un module de division IP *core* de *Xilinx*. Les 16 **LSB** résultants sont la partie fractionnaire sur laquelle la requantification est effectuée. En réutilisant les valeurs de l'exemple précédent et en appliquant cette méthode,

avec une amplitude maximale de 0x1000, le coefficient 0x0010 devient 0x0100 (256) et son information ne sera pas perdue lorsque requantifiée sur 8 bits, donnant 0x01. Ainsi, la méthode proposée permet de conserver 1 bit supplémentaire de précision comparé à une opération de décalage de bits, ce qui est un avantage majeur lorsque les coefficients sont requantifiés sur un nombre de bits très petits, comme 5 ou 6 bits.

Dans une transformée par ondelettes, les coefficients possédant une amplitude supérieure contiennent plus d'information sur la forme d'onde originale que les coefficients de faible amplitude [118]. Une caractéristique de l'approche proposée consiste à transmettre le coefficient d'amplitude maximale en pleine résolution (16 bits), ce qui permet à tous les coefficients de retrouver leur résolution originale lors de l'opération de normalisation inverse, au prix de devoir transmettre 16 bits supplémentaires. Pour chacun des coefficients reçus à la station de base, l'opération suivante est effectuée :

$$ReconstructCoef = MaxAmplitudeCoef \times LowQuantizeCoef \quad (5.1)$$

Où *LowQuantizeCoef* est le coefficient dont la plage dynamique a été maximisée et qui a été requantifié sur un nombre de bits moindre, *MaxAmplitudeCoef* est l'amplitude originale du coefficient de plus grande amplitude en pleine résolution (16 bits) et *ReconstructCoef* est le coefficient reconstruit sur 16 bits. Le CR atteint par la méthode décrite est exprimé par l'équation suivante :

$$CR = \frac{48samples \times 16bits}{48bits + 16bits + N \times Q} \quad (5.2)$$

Où *N* est le nombre de coefficients conservés et *Q* est le nombre de bits employé dans l'opération de compression par requantification.

Le module de compression nécessite 430 cycles d'horloge par AP, soit huit cycles d'horloge par coefficient plus 46 cycles d'horloge de latence du module de division. Au total, la latence nécessaire afin de compresser un AP avec une horloge de 20 MHz est de 73.1  $\mu$ s (1462 cycles d'horloge), ce qui est en dessous du maximum de 75  $\mu$ s alloué.

### 5.3.4 Innovations

Les méthodes conventionnelles de compression par ondelettes effectuent une compression simplement en conservant les coefficients de la transformée qui sont supérieurs à un seuil [47], [77], [78], ou qui contiennent une certaine quantité d'énergie [78], [79], [80]. La méthode développée se distingue des méthodes de compression par ondelettes conventionnelles en permettant d'atteindre de meilleurs taux de compression grâce à une diminution du nombre de bits de quantification des coefficients conservés tout en minimisant la perte de précision grâce à une maximisation de la plage dynamique de ces derniers. La méthode permet également de reconstruire les signaux compressés avec la même résolution que les données d'origine. Cet avantage essentiel survient grâce à la transmission en pleine résolution du

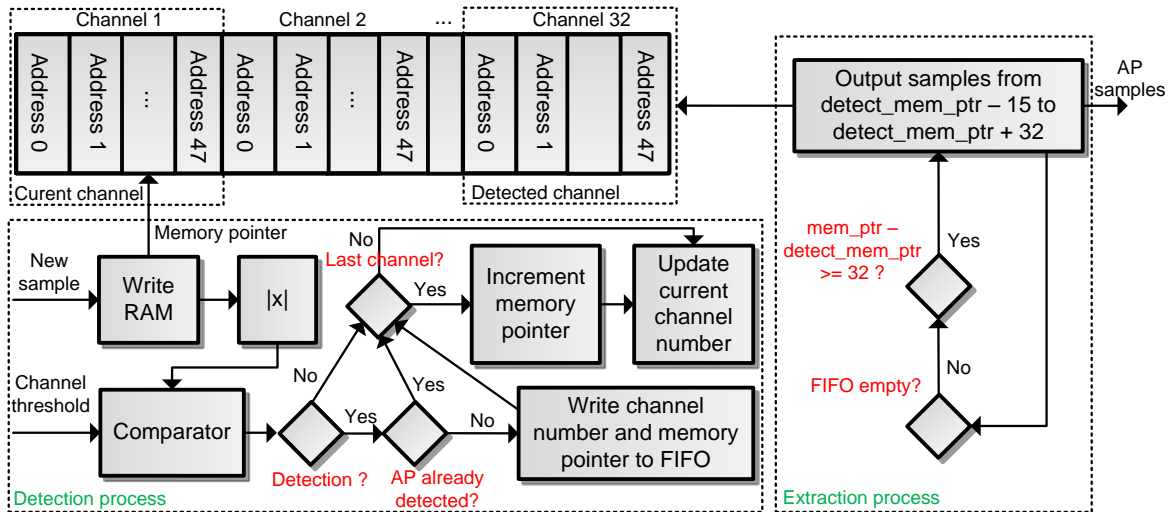


FIGURE 5.6 – Algorithme de détection et de stockage en mémoire des AP.

coefficient de la transformée d’amplitude maximale (coefficient contenant le plus d’information sur la forme d’onde initiale) et qui est utilisée lors de l’opération de normalisation inverse pendant la reconstruction. Normalement, diminuer le nombre de bits de quantification des coefficients afin de compresser davantage ne permet pas de recouvrer la résolution d’origine lors de l’étape de reconstruction des signaux. Ainsi, la méthode proposée innove en permettant de reconstruire le signal en pleine résolution. Il est à noter que cet algorithme et son implémentation ont fait l’objet de deux publications scientifiques [113], [114] et d’une déclaration d’invention (# de divulgation 01736-DIV-01) auprès de l’agent de brevet de l’Université Laval.

## 5.4 Algorithme de détection des AP

L’algorithme de détection des AP est exécuté indépendamment sur 32 canaux en parallèle et nécessite, par conséquent, une implantation optimisée, rapide et efficace. Pour sa simplicité et ses bonnes performances de détection, l’algorithme de détection à valeur absolue a été retenu pour être implémenté dans le *headstage*. Les raisons guidant la sélection de cet algorithme sont détaillées à la section 4.3.1.

### 5.4.1 Aspects mathématiques

Les aspects mathématiques de l’algorithme de détection des AP à valeur absolue sont présentés en détail à la section 4.3.1.

## 5.4.2 Implémentation optimisée pour FPGA

La puce *Intan* est configurée pour conditionner et échantillonner à une fréquence de 20 kHz des échantillons de 16 bits. Pour chacun des canaux, un tampon de 2.4 ms autour du point de détection est stocké en mémoire, ce qui correspond à une fenêtre de 48 échantillons (15 échantillons avant et 32 échantillons après le point de détection). De façon à couvrir le pire cas, qui consiste à capturer 32 AP simultanément (1 AP par canal), un tampon de 3072 octets (24,576 bits) doit être alloué en mémoire. Une telle quantité de données doit être implémentée par des blocs de mémoire RAM interne au FPGA. La stratégie adoptée consiste à diviser un bloc de mémoire RAM en 32 sections, où chaque section est dédiée à un canal spécifique et agit en tant que tampon de mémoire circulaire. Étant donné que la puce *Intan* produit une séquence ordonnée d'échantillons, du canal 1 au canal 32, chaque nouvel échantillon est stocké à l'adresse d'un pointeur de mémoire unique qui est additionnée à l'adresse de base de la section de l'échantillon courant. Lorsque l'échantillon du 32<sup>ème</sup> canal est traité, le pointeur de mémoire unique est incrémenté d'une position. Entre temps, la valeur absolue de chaque nouvel échantillon est comparée à un seuil, dont les valeurs sont stockées dans un tableau logique contenant les 32 seuils sur 16 bits (voir la section 5.5).

Lorsqu'une détection d'AP se produit, un tableau de 32 bits (1 bit par canal) est mis à jour à la position du canal où il y a eu détection afin de garder en mémoire l'état détecté. Simultanément, le pointeur de mémoire unique est écrit, ainsi que le numéro de canal, dans une mémoire *FIFO IP core* de *Xilinx*. Si le tableau de détection indique qu'une détection s'est déjà produite pour ce canal, la détection est ignorée, car l'échantillon détecté appartient à un AP dont les 32 échantillons suivant le point de détection sont en attente d'acquisition. L'utilisation d'une mémoire de type *FIFO* garantit que les AP détectés sont traités dans le bon ordre de détection. Lorsque la mémoire *FIFO* n'est pas vide, un processus parallèle extrait les échantillons de l'AP en comparant le pointeur de mémoire du premier AP détecté (élément à la tête de la *FIFO*) avec le pointeur de mémoire unique courant. Si la différence est supérieure ou égale à 32 échantillons, 15 échantillons avant et 32 échantillons après le pointeur de mémoire sont séquentiellement émis vers le module de calcul de la DWT par *lifting* à partir des échantillons stockés en mémoire RAM. Lorsque le dernier échantillon est émis, le tableau de détection est mis à jour afin de réactiver la détection pour ce canal. Les étapes de détection susmentionnées sont présentées schématiquement à la figure 5.6.

## 5.5 Seuillage adaptatif

### 5.5.1 Aspects mathématiques

Les aspects mathématiques de l'algorithme de seuillage adaptatif basé sur une boucle de rétroaction estimant l'écart-type du bruit dans le signal sont présentés en détail à la section 4.3.1. La boucle de rétroaction est présentée à la figure 5.7.

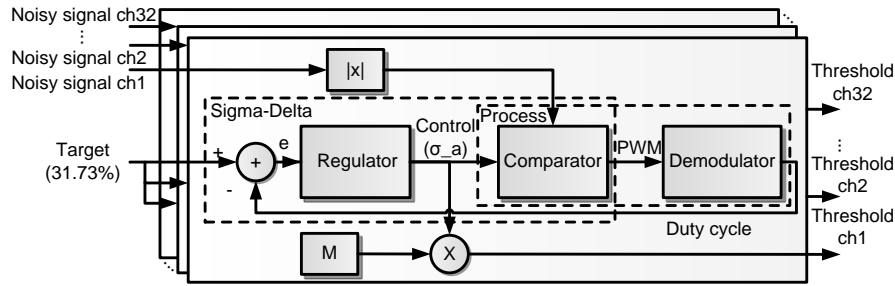


FIGURE 5.7 – Boucle de rétroaction permettant de calculer le seuil de détection pour chacun des 32 canaux en parallèle.

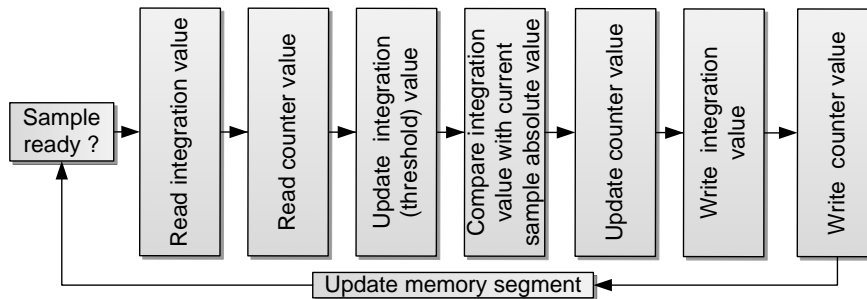


FIGURE 5.8 – Algorithme de calcul du seuil de détection.

### 5.5.2 Implémentation optimisée pour FPGA

L'implémentation de l'algorithme est optimisée pour utiliser seulement des ressources logiques de base comme des blocs de mémoire RAM, des circuits d'addition, des comparateurs et des registres de décalages. Cette implémentation est similaire à celle présentée à la section 4.3.2 du chapitre 4 à l'exception qu'elle calcule le seuil pour 32 canaux en parallèle et qu'elle utilise des ressources logiques disponibles dans un FPGA. Premièrement, l'intégrateur est basé sur une somme de Riemann :

$$I(n + 1) = I(n) + K \times T \times E(n), \quad (5.3)$$

Où  $K$  est le gain de l'intégrateur,  $T$  est la période d'échantillonnage et  $E(n)$  l'erreur relative à la référence (31.73%). La sortie  $I(n)$  est l'approximation de l'écart-type du bruit,  $\sigma_n$ . Cette valeur est comparée avec la valeur absolue de l'échantillon courant et la sortie d'un comparateur est mise à 1 si la valeur absolue de l'échantillon courant est supérieure à  $\sigma_n$  et à 0 autrement. Un bloc de mémoire RAM, divisé en 32 segments, est utilisé pour stocker 32 tampons circulaires (1 tampon circulaire par canal), afin de garder une certaine quantité de bits de comparaison en mémoire. Avant d'inscrire le bit de comparaison courant, le dernier bit à la fin du tampon circulaire est récupéré. Si le bit de comparaison courant est 1 et le dernier bit est 0, un compteur est incrémenté. Si le bit de comparaison courant est 0 et le dernier bit est 1, un compteur est décrémenté. Dans les deux autres cas, la valeur du



compteur reste inchangée. Ce compteur contient la somme de tous les bits en mémoire pour ce canal qui, lorsque divisé par la taille du tampon circulaire, représente le ratio d'échantillons dont l'amplitude est supérieure à  $\sigma_a$ , valeur qui est notée  $P(n)$ . Puisque la taille du tampon circulaire est connue à l'avance, et pour éviter l'utilisation d'un circuit de division, la référence est ajustée à 31.73% multipliée par la taille du tampon circulaire, noté  $L$ . Ainsi, l'opération  $E(n) = Reference - P(n)/L$  devient  $L \times E(n) = L \times Reference - P(n)$ . Le facteur  $L$  qui multiplie l'erreur  $E(n)$  peut être joint avec le gain  $K$  de l'intégrateur et avec la période d'échantillonnage  $T$ . Puisque  $K$ ,  $L$  et  $T$  sont connus à l'avance, la multiplication  $K \times T \times L \times E(n)$  est approximée par une simple opération de décalage de bits de l'erreur  $E(n)$ . Le compteur de bits et la valeur de l'intégrateur sont stockés en mémoire RAM aux côtés du tampon circulaire, et ce, pour tous les canaux. Les étapes susmentionnées sont effectuées de manière séquentielle et sont résumées à la figure 5.8. Chacune des mises à jour du seuil nécessite seulement 9 cycles d'horloge par canal, ce qui permet au module de calcul du seuil d'être alimenté par fréquence d'horloge aussi faible que 5.76 MHz (9 cycles  $\times$  32 canaux  $\times$  20 kHz).



## Chapitre 6

# Résultats expérimentaux

Ce chapitre présente les caractéristiques et les performances du système optogénétique à 32 canaux mesurés lors d'expériences *in vitro* et *in vivo*. Les expériences *in vivo* ont été effectuées avec une souris transgénique exprimant le gène Channelrhodopsin-2 et les signaux utilisés dans les tests *in vitro* ont été synthétisés en suivant la même procédure que celle présentée à la section 4.3.1.

### 6.1 Caractéristiques du système

La figure 6.1a présente le PCB flexible rigide final avec chacun des circuits et composants identifiés. Les dimensions, une fois le PCB replié, sont de  $17 \times 18 \times 10 \text{ mm}^3$  et le poids du système, sans la batterie, est de 2.8 g. La figure 6.1b présente le système avec un module implantable installé, tandis que les figures 6.1c et 6.1d présentent le PCB flexible rigide replié

TABLE 6.1 – Caractéristiques du *headstage* optogénétique à 32 canaux.

Paramètre	Valeur
# de canaux d'enregistrement	32
# de canaux de stimulation	32
Résolution de l'ADC	16 bits
Fréquence d'échantillonnage par canal	20,000 échantillons/s
Bande passante du filtre passe-bas	0.1 kHz - 20 kHz
Bande passante du filtre passe-haut	0.1 Hz - 500 Hz
Bruit référé à l'entrée	$2.4 \mu V_{rms}$
Bande de fréquence du transmetteur sans fil	2.4 GHz
Débit maximal effectif du transmetteur sans fil	1.1 Mbps
Dimensions (L x D x H)	$17 \times 18 \times 10 \text{ mm}^3$
Poids ( <i>headstage</i> seulement)	2.8 g
Consommation en courant	52 mA
Autonomie (avec une batterie 100 mAh)	105 minutes

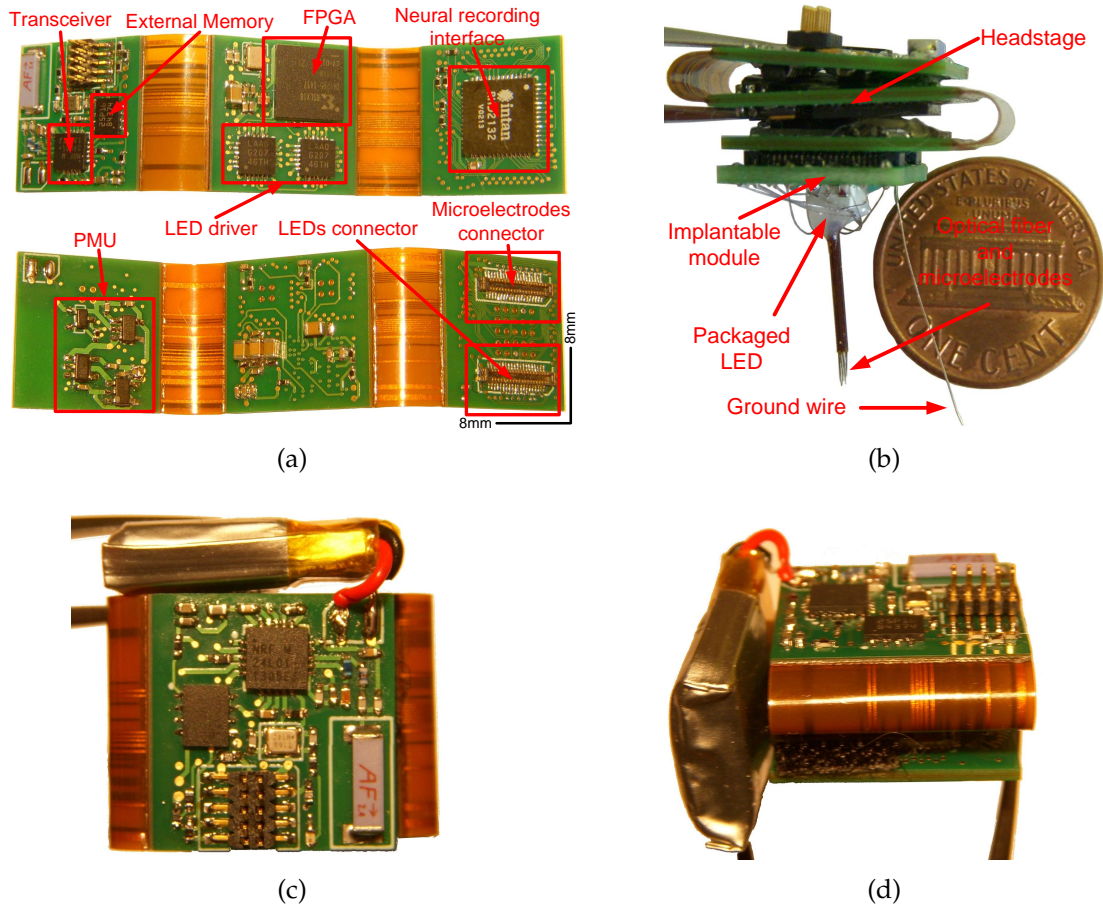


FIGURE 6.1 – a) Vue recto verso du PCB flexible rigide non replié, b) vue du *headstage* avec un module implantable installé, b) vue du dessus du *headstage* replié avec la batterie sur le côté, d) vue de devant du *headstage* replié avec la batterie sur le côté.

sous deux angles différents avec la batterie sur le côté. Le *headstage* possède une autonomie mesurée, sans stimulation optique, de 105 minutes avec une batterie LiPo de 100 mAh. Les caractéristiques du *headstage* sont résumées dans le tableau 6.1.

## 6.2 Résultats expérimentaux *in vitro*

Cette section présente les performances du *headstage* optogénétique à 32 canaux mesurés lors d'expériences *in vitro*. Le montage de test est présenté à la figure 6.2, dans lequel le *headstage* transmet les données sans fil à une station de base. Il est à noter que les résultats reportés dans cette section ont été présentés dans une publication scientifique, voir [114].

### 6.2.1 Résultats de détection des AP

Dans cette section, les résultats de détection ont été obtenus en transmettant simultanément les AP détectés ainsi que le signal original pour un canal spécifique, et ce, afin de déterminer

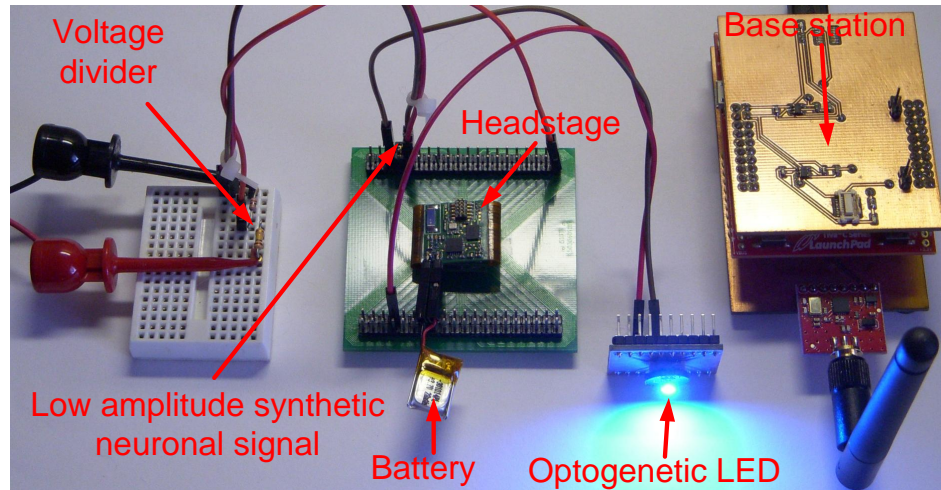


FIGURE 6.2 – Montage de test *in vitro* du headstage optogénétique à 32 canaux.

le nombre de VP et de FP à partir du signal original.

### Paramètres d'implémentation

Il existe un compromis entre la taille du tampon circulaire et la réactivité temporelle de la boucle de contrôle qui calcule le seuil. Peu importe la taille du tampon, la valeur calculée du seuil sera la même une fois que le régime permanent de la boucle est atteint. La taille du tampon circulaire a été fixée empiriquement à 256 bits, ou 32 octets. Le gain total de l'intégrateur, constitué du gain  $K$ , de la période d'échantillonnage  $T$  et de la taille du tampon circulaire  $L$ , a été fixé empiriquement à  $4 \times 10^{-4} \times 1/20000 \times 128$  (voir section 5.5), ce qui génère un gain total de  $2^{-11}$ . Ce gain est implémenté par une opération de décalage de 11 bits vers la droite. Finalement, le paramètre  $L \times SetPoint$  est approximé par l'entier 81 ( $256 \times 0.317 \approx 81$ ) et le gain  $M$  du seuil a été fixé à 4 et il est implémenté par une opération de décalage de deux bits vers la gauche.

### Résultats de détection

La méthode proposée est comparée avec d'autres méthodes de seuillage adaptatif implémenté dans *Matlab* et exécuté sur le signal brut i.e sur le même signal ayant été traité par le *headstage*. Les algorithmes implémentés dans *Matlab* n'ont fait l'objet d'aucune optimisation (nombres à virgules flottantes, multiplications, divisions, etc.) afin de comparer l'algorithme proposé avec les meilleurs résultats pouvant être obtenus avec d'autres algorithmes. Les deux méthodes de seuillage implémentées dans *Matlab* sont celle à base de l'opérateur MAD (*Wave Clus*) et celle à base de la valeur RMS du signal. Ces deux algorithmes sont décrits par les équations 2.1 et 2.2 respectivement et ont été implémentés avec les mêmes paramètres que dans les tests effectués à la section 4.5.2.

TABLE 6.2 – Comparaison des performances de détection entre la méthode proposée exécutée par le *headstage* à 32 canaux versus d’autres méthodes exécutées sur les signaux bruts importés dans *Matlab* et possédant un *firing rate* variant de 25 à 125 AP/s.

Signal		Méthode proposée			RMS			Wave Clus (MAD)		
SNR	AP	Total	VP	FP	Total	VP	FP	Total	VP	FP
0	3900	2879	2739 (71.5%)	140 (4.86%)	2641	2571 (65.9%)	70 (2.65%)	2840	2718 (69.7%)	122 (4.30%)
5	3900	3861	3794 (97.3%)	67 (1.74%)	3694	3652 (93.6%)	42 (1.14%)	3797	3765 (96.5%)	32 (0.84%)
10	3901	3936	3897 (99.9%)	39 (0.99%)	3920	3900 (100%)	20 (0.51%)	3928	3900 (100%)	28 (0.71%)
15	3900	3917	3900 (100%)	17 (0.43%)	3895	3890 (99.7%)	5 (0.13%)	3913	3900 (100%)	13 (0.33%)

Le tableau 6.2 montre les performances de détection pour des signaux synthétiques de 60 secondes chacun. Chaque signal a été généré avec un SNR distinct, soit avec un SNR de 15 dB, 10 dB, 5 dB ou de 0 dB et avec un *firing rate* variant de 25 à 125 AP/s.

Comme on peut le constater, la méthode proposée compétitionne avantageusement avec les deux autres méthodes plus gourmandes en calculs. Pour les SNR de 0 dB et de 5 dB, la méthode proposée génère les meilleurs résultats de VP, au coût d’une légère augmentation des FP (seulement 4.86 % à 0 dB). En fait, à 0 dB, la méthode proposée détecte 5.6% plus de VP que la méthode RMS au coût de seulement 2.21% de plus de FP et 1.8% plus de VP que la méthode MAD au coût d’une augmentation de 0.56% des FP. Ces résultats sont excellents, compte tenu que les VP sont plus important que les FP, car ces derniers peuvent être éliminés lors d’un traitement subséquent sur ordinateur (*clustering* par exemple). Il est à noter que ces résultats concordent avec ceux obtenus avec le *headstage* à deux canaux et qui sont présentés à la section 4.5.2.

Afin de donner un aperçu de la robustesse du seuil face à des variations instantanées du bruit, la figure 6.3 montre le seuil (en rouge) en réponse à une variation du SNR passant de 15 dB, 0 dB, 5 dB à 10 dB avec un *firing rate* constant de 75 AP/s. Comme prévu, le seuil s’ajuste dynamiquement en fonction du bruit selon une dynamique qui peut être modifiée en changeant certains paramètres d’implémentations décrits à la section 5.5, comme le gain de l’intégrateur ou bien la taille du tampon circulaire.

## 6.2.2 Résultats de compression du *headstage* avec différents SNR

### Méthodologie

Dans cette section, le *headstage* a transmis les AP détectés et compressés en même temps que le signal original pour un canal spécifique, et ce, afin d’associer chacun des AP compressés avec sa version originale. Il est à noter que les résultats reportés dans cette section ont été présentés dans une publication scientifique, voir [114].

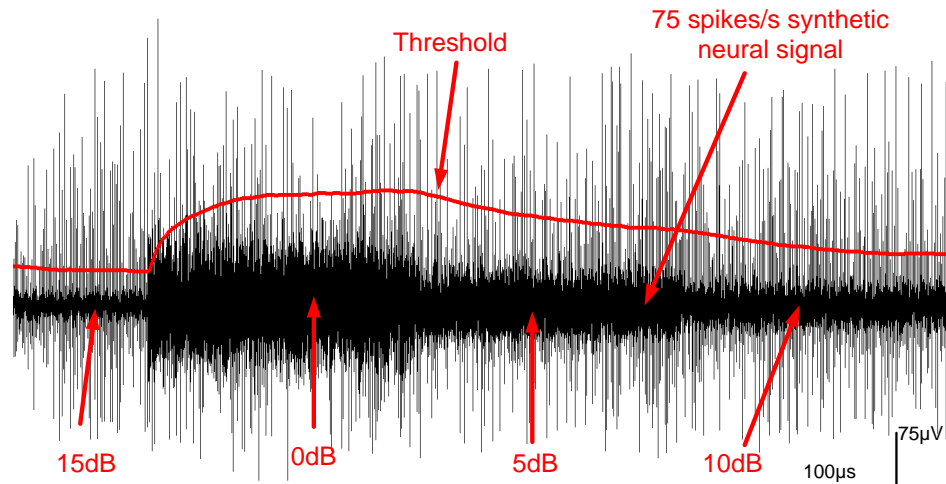


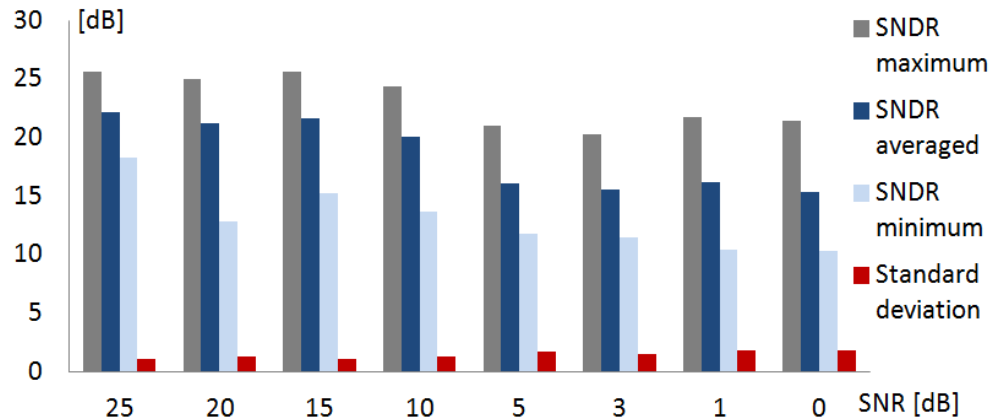
FIGURE 6.3 – Variation du seuil pour un signal ayant un SNR passant de 15 dB, 0 dB, 5 dB à 10 dB et un *firing rate* constant de 75 AP/s.

Les performances de compression ont été quantifiées pour des signaux neuronaux ayant différents SNR afin d'évaluer l'impact du bruit sur la compression. À cette fin, des signaux neuronaux synthétiques possédant des SNR différents ont été synthétisés avec un SNR variant de 25 dB à 0 dB, et ce, afin de couvrir une plage de signaux neuronaux faiblement bruités à très bruités. À partir de ces signaux, 1500 AP ont été détectés, compressés et transmis par le *headstage* pour chaque SNR. Pour ces tests, le taux de compression du *headstage* a été fixé à 4.17, soit 20 coefficients retenus avec une précision de 6 bits chacun (même CR que lors des tests *in vivo* présentés à la section 6.3.2).

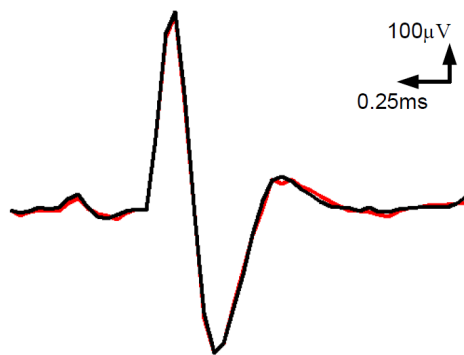
## Résultats

Les résultats de SNDR sont présentés à la figure 6.4a et suggèrent que la qualité de la reconstruction décroît en fonction de la diminution du SNR du signal. Les SNDR maximum, minimum et moyen suivent la même tendance tandis que l'écart-type du SNDR augmente de façon inversement proportionnelle au SNR du signal. Cette tendance indique que la qualité de la reconstruction tend à diminuer légèrement plus un signal est bruité. Ce phénomène est attribuable à l'énergie des coefficients issus d'une DWT d'un signal bruité qui est distribué sur un plus grand nombre de coefficients, dont certains sont éliminés lors de la compression. Le SNDR moyen varie de seulement 6.82 dB (30.7%) entre un signal possédant un SNR de 25 dB et un signal possédant un SNR de 0 dB, ce qui démontre que l'algorithme de compression du *headstage* compresse avec succès les AP, même lorsque le SNR du signal est très faible.

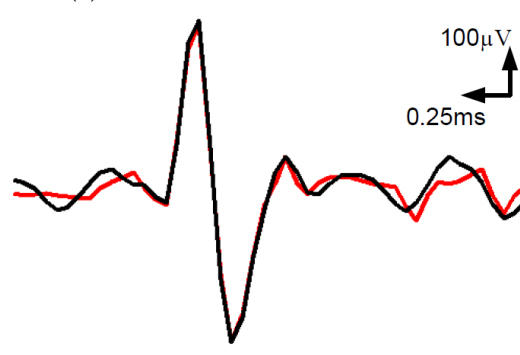
Afin d'avoir un indicatif visuel de la qualité de la reconstruction, les figures 6.4b et 6.4c montrent la meilleure et la pire reconstruction en fonction du SNDR pour un signal possédant un SNR de 10 dB. La figure 6.4c montre que même la pire reconstruction reproduit



(a)



(b)



(c)

FIGURE 6.4 – a) Résultats du SNDR pour un SNR variant de 25 dB à 0 dB, b) meilleure reconstruction suivant le SNDR pour un signal ayant un SNR de 10 dB, c) pire reconstruction suivant le SNDR pour un signal ayant un SNR de 10 dB

fidèlement la forme d'onde originale tout en éliminant le bruit, puisque la forme de l'AP demeure intacte.

Il est à noter que les résultats de la figure 6.4a sont détaillés dans le tableau B.1 à l'annexe B.

## 6.2.3 Résultats comparatifs de compression pour différents CR

### Méthodologie

Dans cette section, le *headstage* a transmis les signaux neuronaux synthétiques acquis pour un canal spécifique, et ce, afin d'importer les formes d'ondes dans *Matlab*. Les signaux neuronaux synthétiques utilisés sont issus du fichier *C\_Easy1\_noise01.mat* de la banque en ligne [12] et sont des signaux complets déjà bruités avec du bruit neuronal et électronique, en plus du bruit ajouté par l'électronique du *headstage*, par le générateur de fonction aléatoire et par les interférences électromagnétiques ambiantes (60 Hz par exemple). À partir de ces formes d'ondes, 2890 AP ont été détectés dans *Matlab* et sont utilisés pour comparer les



TABLE 6.3 – Paramètres de compression utilisés afin de comparer les performances des algorithmes de compression pour différent CR.

CR	<i>Headstage</i>		<i>Ondelette (sans maximisation)</i>		Delta Bits/diff	CS	
	Coeffs	Bits/Coeffs	Coeffs	Bits/Coeffs		Coeffs	Bits/Coeffs
3	29	7	29	7	6	30	8
4	20	6	20	6	4	22	8
5	18	5	18	5	3	17	8
6	13	5	13	5	N/A	14	8
7	11	4	11	4	2	11	8
8	8	4	8	4	N/A	10	8

SNDR produits pour six différents CR : 3, 4, 5, 6, 7 et 8.

De façon à pouvoir apprécier les gains engendrés par la maximisation de la plage dynamique des coefficients, les tests ont été exécutés avec et sans cette caractéristique. L’algorithme de compression du *headstage* est aussi comparé avec les algorithmes de compression *Delta* (voir section 2.3) et *CS* avec base de parcimonie optimale (voir section 2.5). Le logiciel *Matlab* est utilisé afin de faciliter les tests selon plusieurs CR et pour faciliter l’implémentation des autres algorithmes de compression. Les tests présentés dans cette section diffèrent de ceux présentés au chapitre 3 en fonction de l’utilisation de signaux neuronaux synthétiques ayant été traités par le *headstage* en suivant toute la chaîne d’acquisition sans fil. Il est important de noter que les résultats reportés dans cette section ont été présentés dans une publication scientifique, voir [114].

Les algorithmes *Delta* et *CS* sont implémentés avec les paramètres décrits aux sections 3.3 et 3.5 respectivement, à l’exception que la matrice de parcimonie utilisée dans l’algorithme *CS* est entraînée sur 10% des 2890 AP détectés, au lieu de 5%, afin d’optimiser ses résultats. Puisque l’algorithme *Delta* ne peut pas produire les taux de compression 6 et 8 à cause de limitations dans la réduction du pas de quantification, les résultats de SNDR correspondants sont extrapolés à partir des résultats obtenus avec les CR de 5 et de 7. Les paramètres de compression des algorithmes testés sont présentés dans le tableau 6.3.

## Résultats

Les SNDR moyens ainsi que la plage couverte par les SNDR en fonction du CR sont présentés à la figure 6.5a. Ces résultats démontrent que la maximisation de la plage dynamique des coefficients augmente le SNDR moyen de plus de 10 dB pour tous les CR. L’algorithme de compression du *headstage* et l’algorithme *CS* génèrent sensiblement les mêmes résultats de SNDR moyen alors que l’algorithme *Delta* est en dessous de 5 dB de façon presque constante. La variabilité des résultats produits par les algorithmes *CS* et *Delta* est perceptible en regardant leurs plages de SNDR couverte dans la figure 6.5a. En fait, l’algorithme *CS* a généré des AP compressés/décompressés avec des SNDR en dessous de 0 dB, alors que l’algorithme de

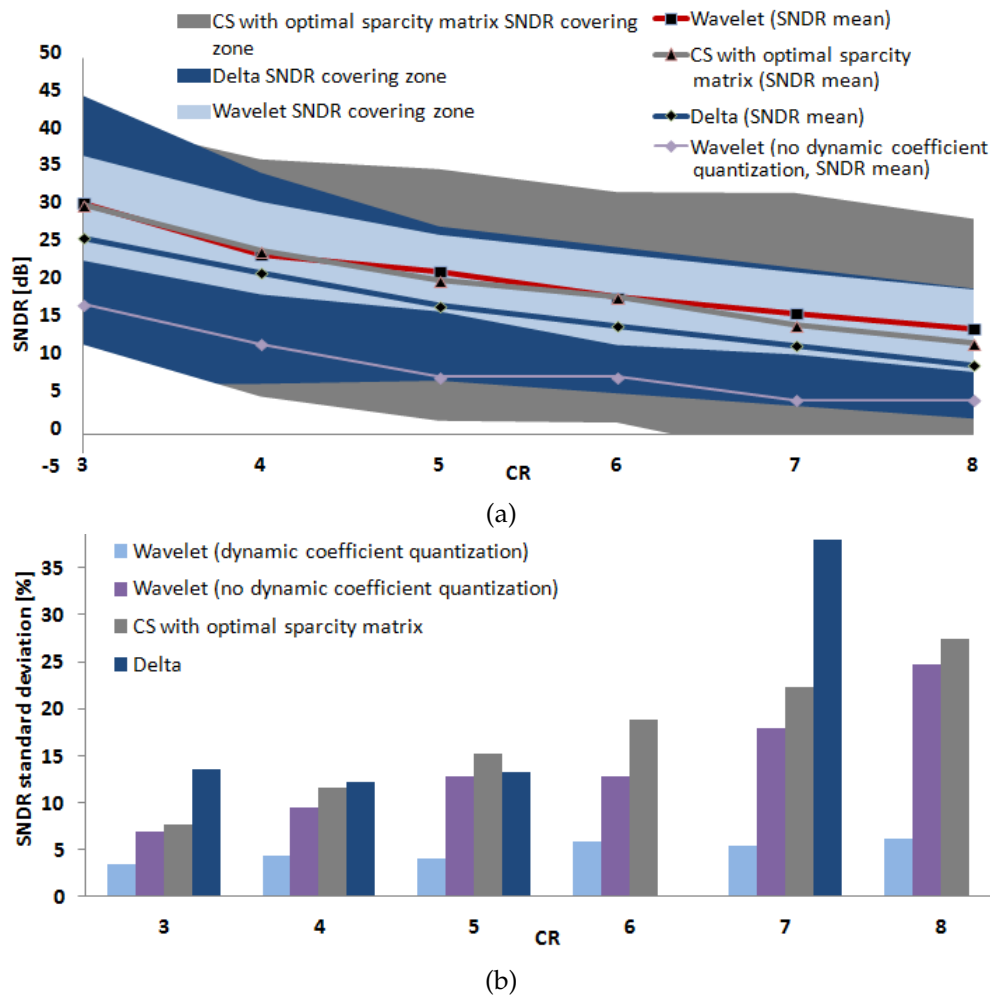


FIGURE 6.5 – Comparaison des performances de compression du *headstage* versus celles obtenues avec l’algorithme *Delta*, l’algorithme CS avec base parcimonieuse optimale et l’algorithme du *headstage* sans maximisation la plage dynamique des coefficients, a) SNDR moyen et plage couverte par le SNDR, b) écart-type du SNDR relatif au SNDR moyen.

compression du *headstage* n’a jamais produit de SNDR en dessous de 8.4 dB.

La figure 6.5b montre un histogramme de l’écart-type des SNDR relatif au SNDR moyen pour chaque algorithme. Comme l’on pouvait s’y attendre, l’algorithme de compression du *headstage* a produit, pour tous les CR, l’écart-type le plus petit, jusqu’à 6 fois en dessous de ceux obtenus avec les autres algorithmes. Ces résultats démontrent que le *headstage* compresse/décompresse les AP avec une qualité constante, comparativement aux autres algorithmes qui génèrent des résultats incertains selon les formes d’ondes des AP compressés.

Il est à noter que les résultats de la figure 6.5a et ceux de l’histogramme de la figure 6.5b sont détaillés dans les tableaux B.2, B.3, B.4 et B.5 à l’annexe B.

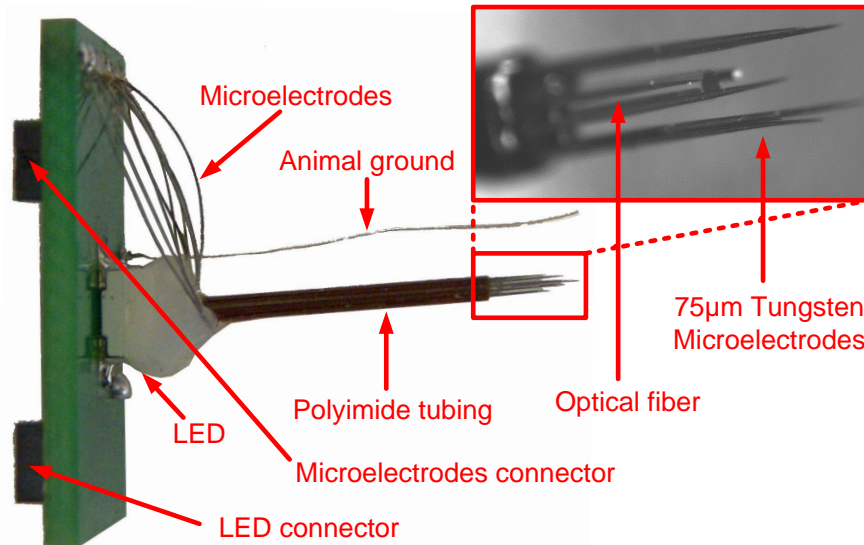


FIGURE 6.6 – Module implantable incorporant une DEL couplée à une fibre optique insérée dans un tube de polyimide avec 8 microélectrodes d’enregistrement, une microélectrode de référence et un fil de mise à la masse.

#### 6.2.4 Détection/compression sur 32 canaux simultanément

Les résultats détaillés de détection/compression à partir des 32 canaux simultanément sont présentés à la section B.2 de l’annexe B. Ces résultats proviennent d’un test d’une durée de 60 secondes avec un signal neuronal synthétique dupliqué sur les 32 canaux du *headstage* afin de reproduire le pire cas où des AP sont détectés simultanément sur les 32 canaux. Le signal neuronal synthétique utilisé possède un *firing rate* de 50 AP/s et un SNR de 10 dB. Tous les canaux ont détecté  $3103 \text{ AP} \pm 1$ , ce qui démontre que le *headstage* peut détecter et compresser les signaux issus de 32 canaux en parallèle. Le taux de compression atteint lors de cet essai est de 17.2. Sans compression, transmettre une telle quantité de données nécessite 10.24 Mbps, ce qui est de loin supérieur au 1.1 Mbps atteint par le type de transmetteur utilisé.

### 6.3 Résultats expérimentaux *in vivo*

L’efficacité du *headstage* dans des conditions d’opérations normales a été validée sur le cerveau d’une souris exprimant le gène Channelrhodopsin-2. Les tests ont été réalisés le 19 mai 2015 au CRIUSMQ. Un module implantable a été conçu sur mesure par la compagnie *Doric Lenses* afin de valider le système. L’implant n’a pas été fixé de façon permanente sur le crâne, afin de faciliter les premières procédures de test, mais plutôt retenu avec le *headstage* par un outil externe. La Figure 6.6 montre le prototype final de l’implant avec un agrandissement sur l’extrémité de la sonde. L’implant est constitué de 9 microélectrodes en Tungsten de  $75 \mu\text{m}$  (impédance de  $1\text{M}\Omega$ ), soit 8 microélectrodes d’enregistrement et 1 microélectrode de référence, d’un fil de mise à la masse afin de fixer le potentiel de l’animal ainsi que d’une

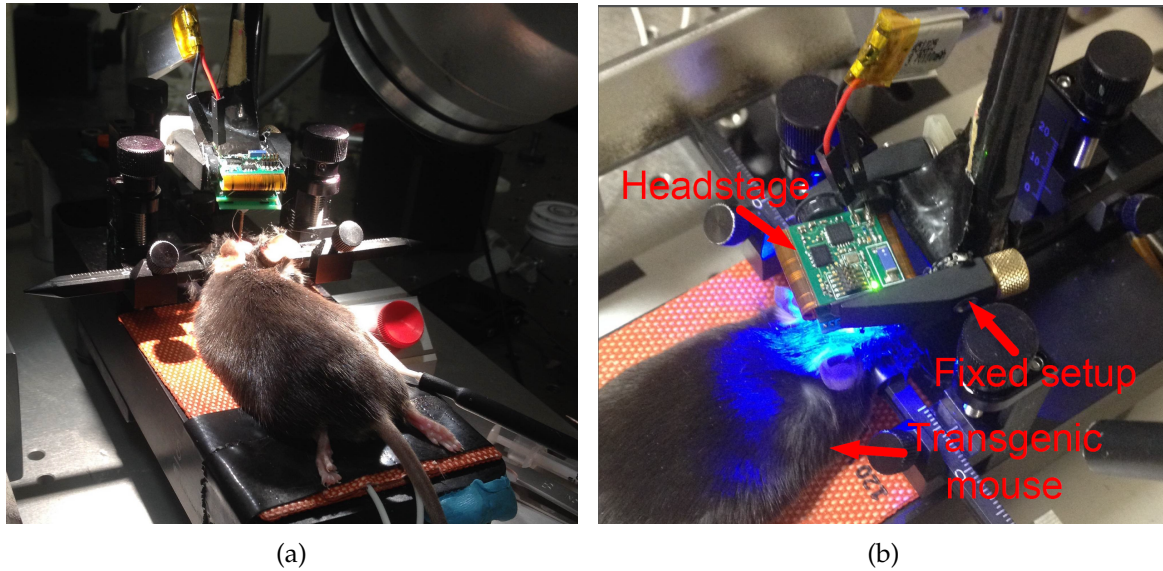


FIGURE 6.7 – a) Montage *in vivo*, b) *headstage* en fonctionnement pendant une expérience optogénétique.

DEL (465 nm) haute puissance [102] couplée à une fibre optique de 200  $\mu\text{m}$  de diamètre, le tout inséré dans un tube de polyimide. Le montage de test est présenté aux figures 6.7a et 6.7b. Il est à noter que les résultats reportés dans cette section ont été présentés dans deux publications scientifiques, voir [113] et [114].

### 6.3.1 Préparation de l'animal

La préparation de l'animal a été réalisée avec une souris adulte (50 g) en suivant la même procédure que celle présentée à la section 4.6.1, à l'exception que la sonde a été abaissée dans la partie frontale de l'Hypocampe du cerveau.

### 6.3.2 Paramètres du *headstage*

Le taux de compression des AP a été fixé à 4.17, soit 20 coefficients retenus sur 6 bits, et le gain du seuil de détection des AP a été fixé à 4 fois l'écart-type estimé. Un CR de 4.17 avec un débit de transmission effectif maximal de 1.1 Mbps permet d'enregistrer, sans perte, les signaux de 32 microélectrodes possédant un *firing rate* constant de 187 AP/s. Ce taux de compression a été sélectionné en considérant que le *firing rate* d'un neurone typique est rarement supérieur à 100 AP/s et ne dépasse pas les 200 AP/s à cause du temps de recharge du gradient d'ions interne à la cellule [119].

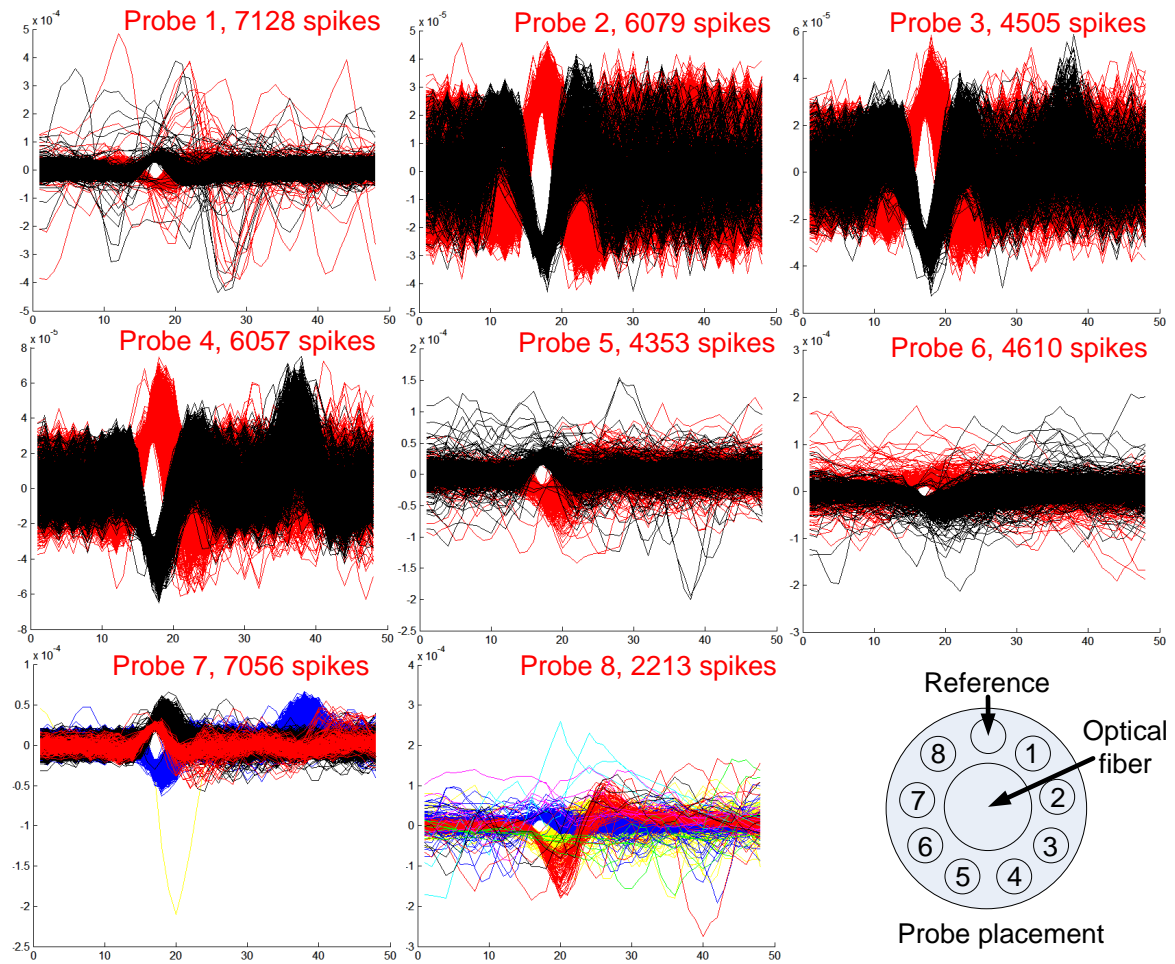


FIGURE 6.8 – AP détectés et regroupés en *clusters* à partir de l’enregistrement simultané des signaux provenant de 8 microélectrodes pendant une expérience *in vivo* d’une durée de 200 secondes.

### 6.3.3 Enregistrements à partir de toutes les microélectrodes

Afin de tirer avantage de la compression, tous les 8 canaux d’enregistrement disponibles sur le module implantable ont été activés simultanément. Le débit minimal requis afin de transmettre les signaux issus de 8 microélectrodes en pleine résolution est de 2.56 Mbps, soit un débit qui est hors de portée du transmetteur utilisé. Les AP détectés ont été regroupés en *clusters* à partir d’une séance d’enregistrement d’une durée de 200 secondes. Les résultats de *clustering* sont présentés à la Figure 6.8. Les associations AP à neurones ont été générées avec une analyse PCA suivie d’une étape de *clustering* en utilisant l’algorithme k-means, où chaque AP est associé à un neurone spécifique.

Comme il est possible de le constater, certaines microélectrodes ont enregistré les signaux provenant des mêmes neurones. Par exemple, les microélectrodes 2, 3 et 4 ont enregistré des formes d’AP similaires, mais d’amplitudes différentes, suggérant que certaines micro-

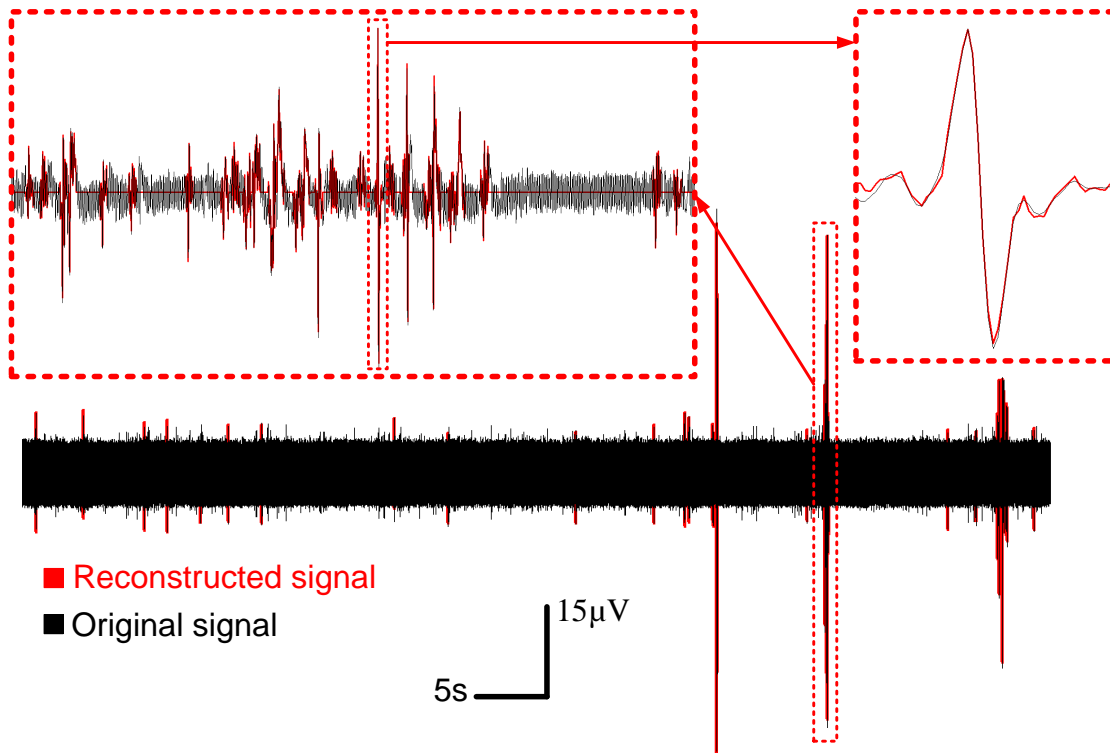


FIGURE 6.9 – Signal neuronal d’une durée de 60 secondes reconstruit avec les AP détectés et décompressés (en rouge) superposés au signal original (en noir).

électrodes étaient plus proches des neurones activés que les autres (voir le placement des microélectrodes à la figure 6.8). Certaines microélectrodes, comme les microélectrodes 1 et 8, ont enregistré des AP qui se sont produits moins fréquemment. Le taux de compression effectif de cette séance d’enregistrement est de 508, incluant la réduction due à la détection des AP multiplié par le CR de 4.17 provenant de la compression des AP.

#### 6.3.4 Résultats de détection et de compression

Les performances de détection et de compression des AP ont été validées en transmettant les AP détectés et compressés en même temps que le signal original pour un canal spécifique. La figure 6.9 montre 60 secondes d’un signal reconstruit (en rouge) superposé au signal original (en noir). Le SNDR moyen a été calculé pour tous les AP détectés et compressés par rapport aux AP originaux extraits du signal original. Il a été trouvé que le module de compression par ondelettes a produit un SNDR moyen de 17.0 dB, ce qui correspond aux résultats obtenus à la section 6.2.2 avec des signaux synthétiques ayant un SNR entre 10 dB et 5 dB.

Le détecteur d’AP a été évalué en comparant ses performances avec les performances de détection obtenues avec le logiciel *Wave Clus* (opérateur MAD) sur les formes d’ondes originales. Pour un seuil ayant le même gain (4 fois l’écart-type estimé), le détecteur du *headstage*

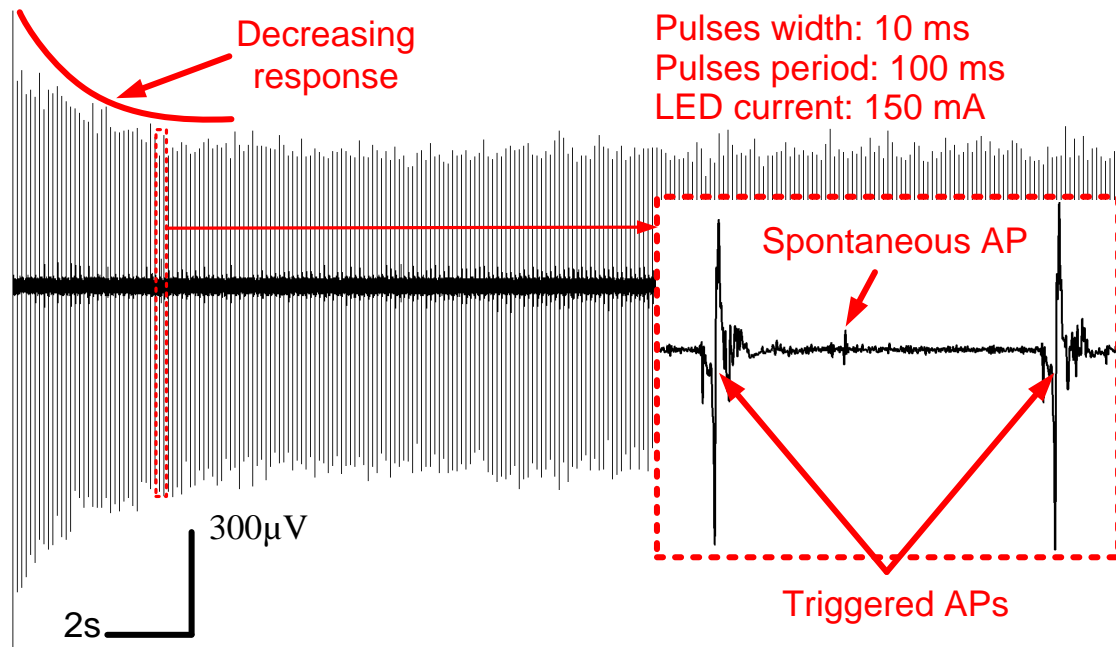


FIGURE 6.10 – Résultats expérimentaux *in vivo* montrant la réponse extracellulaire suite à une activation lumineuse pulsée de 10ms et ayant une période de 100 ms.

à généré 92.4% de concordance avec les AP détectés par *Wave Clus*. En fait, l'algorithme proposé a démontré une meilleure précision que l'algorithme de *Wave Clus* en détectant de plus petits AP qui étaient visuellement détectables avec une proportion de 9.87%.

### 6.3.5 Résultats de stimulation

Le *headstage* permet de personnaliser les patrons de stimulation optique. La largeur des pulses de lumière a été établie à 10 ms selon des documents sur la stimulation optogénétique [120], [112] et le courant dans la DEL a été ajusté à 150 mA, générant une intensité lumineuse mesurée de  $\sim 70 \text{ mW/mm}^2$  au bout de la fibre optique à une longueur d'onde de 465 nm. La figure 6.10 présente l'activité neuronale enregistrée par une des microélectrodes lors de la séance de stimulation optique pulsée. Les résultats expérimentaux démontrent que le *headstage* est capable d'enregistrer simultanément l'activité neuronale induite par la stimulation optique ainsi que l'activité neuronale spontanée (figure 6.10), avec des AP déclenchés par la lumière d'amplitude jusqu'à 1 mV. Dans la figure 6.10, il est possible d'observer une diminution progressive de l'amplitude des AP au début de la stimulation, jusqu'à une stabilisation après  $\sim 5$  secondes. Ce phénomène de *spike silencing* est attribuable à la dépolarisation de la cellule, tel que décrit dans la littérature [112].





## Chapitre 7

# Analyse comparative

Les caractéristiques des trois *headstages* présentés dans ce mémoire sont comparées aux caractéristiques de systèmes similaires dans le tableau 7.1. Tous les systèmes répertoriés dans ce tableau comprennent les données neuronales, soient par détection des AP et/ou en utilisant une méthode alternative. Une fonction de coût a été créée afin de comparer les systèmes entre eux. La fonction de coût est construite à partir des critères suivants :

- **Nombre de canaux d’enregistrement** : Le nombre de canaux d’enregistrement définit la résolution spatiale du système. L’équation suivante permet d’attribuer une note entre 0 et 1 à ce critère :

$$1 - e^{-10 \times \frac{REC_{ch}}{MAX_{rec}}} \quad (7.1)$$

Où  $REC_{ch}$  est le nombre de canaux d’enregistrement du système et  $MAX_{rec}$  est le nombre maximal de canaux d’enregistrement parmi tous les systèmes répertoriés. Cette fonction a été choisie afin de donner une note proche de 1 à tous les systèmes ayant un nombre de canaux d’enregistrement  $\geq 100$  canaux. Une pondération de 12.5% est attribuée à ce critère.

- **Nombre de canaux de stimulation** : Le nombre de canaux de stimulation définit la résolution et la puissance de stimulation du système. L’équation suivante permet d’attribuer une note entre 0 et 1 à ce critère :

$$1 - e^{-10 \times \frac{STIM_{ch}}{MAX_{stim}}} \quad (7.2)$$

Où  $STIM_{ch}$  est le nombre de canaux de stimulation du système et  $MAX_{stim}$  est le nombre maximal de canaux de stimulation parmi tous les systèmes répertoriés. Cette fonction a été choisie afin de donner une note proche de 1 à tous les systèmes ayant un nombre de canaux d’enregistrement  $\geq 12$  canaux. Une pondération de 12.5% est attribuée à ce critère.

- **Résolution** : La résolution est importante afin de pouvoir analyser, classer et traiter les signaux neuronaux acquisitionnés. L’équation suivante permet d’attribuer une note

entre 0 et 1 à ce critère :

$$\frac{RES - MIN_{res}}{MAX_{res} - MIN_{res}} \quad (7.3)$$

Où  $RES$  est la résolution binaire du système,  $MIN_{res}$  est la résolution minimale parmi tous les systèmes répertoriés et  $MAX_{res}$  est la résolution maximale parmi tous les systèmes répertoriés. Cette fonction a été choisie pour sa linéarité et afin de donner une note de 1 aux systèmes ayant la meilleure résolution et une note de 0 aux systèmes ayant la moins bonne résolution. Une pondération de 12.5% est attribuée à ce critère.

- **Fréquence d'échantillonnage** : Tout comme pour la résolution, la fréquence d'échantillonnage est importante afin de pouvoir analyser, classer et traiter les signaux neuro-naux acquis. L'équation suivante permet d'attribuer une note entre 0 et 1 à ce critère :

$$\frac{SAMP - MIN_{samp}}{MAX_{samp} - MIN_{samp}} \quad (7.4)$$

Où  $SAMP$  est la fréquence d'échantillonnage du système,  $MIN_{samp}$  est la fréquence d'échantillonnage minimale parmi tous les systèmes répertoriés et  $MAX_{samp}$  est la fréquence d'échantillonnage maximale parmi tous les systèmes répertoriés. Cette fonction a été choisie pour sa linéarité et afin de donner une note de 1 au système présentant la meilleure fréquence d'échantillonnage et une note de 0 au système présentant la moins bonne fréquence d'échantillonnage. Une pondération de 12.5% est attribuée à ce critère.

- **Poids** : Le poids du système est critique afin que le système puisse être utilisé sur de petits animaux. L'équation suivante permet d'attribuer une note entre 0 et 1 à ce critère :

$$e^{-\frac{WEIGHT - MIN_{weight}}{10}} \quad (7.5)$$

Où  $WEIGHT$  est le poids du système en gramme sans batterie et  $MIN_{weight}$  est le poids minimum en gramme de tous les systèmes non-ASIC répertoriés sans batterie. Cette fonction a été définie afin de donner une note de 1 au système le plus léger tout en pénalisant sévèrement les systèmes trop lourds ( $\geq 10$  g). Une pondération de 12.5% est attribuée à ce critère.

- **Puissance consommée** : La puissance consommée est inversement proportionnelle à l'autonomie du système. L'équation suivante permet d'attribuer une note entre 0 et 1 à ce critère :

$$e^{-\frac{POWER}{100}} \quad (7.6)$$

Où  $POWER$  est la puissance consommée par le système en mW. Une pondération de 12.5% est attribuée à ce critère.

- **Type de transmetteur** : Les transmetteurs commerciaux sont flexibles et peu coûteux, alors que les transmetteurs de type *custom* sont coûteux et longs à développer. Pour ces raisons, les systèmes incorporant un transmetteur commercial obtiennent une note de 1

alors que ceux n'incorporant aucun transmetteur ou bien incorporant un transmetteur de type *custom* obtiennent une note de 0 (critère nommé *Transceiver* dans la fonction de coût). Une pondération de 6.25% est attribuée à ce critère.

- **Méthode de seuillage :** Les systèmes incorporant une méthode de seuillage adaptative obtiennent une note de 1 pour ce critère alors que ceux ayant une méthode de seuillage fixe obtiennent une note de 0 (critère nommé *AdaptiveThr* dans la fonction de coût). Une pondération de 6.25% est attribuée à ce critère.
- **Transmission des AP :** Les systèmes qui transmettent la forme d'onde complète des *AP* obtiennent une note de 1 alors que les systèmes transmettant seulement un sous-ensemble des caractéristiques des *AP* obtiennent la note de 0 (critère nommé *FullSpikes* dans la fonction de coût). Une pondération de 12.5% est attribuée à ce critère.

La fonction de coût est définie par l'équation 7.7 et les résultats produits par cette fonction sont présentés dans la dernière colonne du tableau 7.1 et dans l'histogramme de la figure 7.1. Les critères ne pouvant pas être calculés pour certains systèmes sont présentés en rouge dans le tableau. Pour ces cas de figure, la fonction de coût est calculée en retirant les critères invalides et en répartissant la pondération proportionnellement sur les autres critères. Par exemple, les poids des systèmes *ASIC* une fois incorporés dans un système complet sont inconnus. Conséquemment, le critère  $e^{-\frac{WEIGHT-MIN_{weight}}{10}}$  est retiré de l'équation 7.7 et les pondérations 12.5% et 6.25% sont ajustées à 14.3% et 7.15% respectivement.

$$\begin{aligned}
 \text{Coût} = & \left( \left( 1 - e^{-10 \times \frac{REC_{ch}}{MAX_{rec}}} \right) + \left( 1 - e^{-10 \times \frac{STIM_{ch}}{MAX_{stim}}} \right) + \frac{RES - MIN_{res}}{MAX_{res} - MIN_{res}} + \right. \\
 & \frac{SAMP - MIN_{samp}}{MAX_{samp} - MIN_{samp}} + e^{-\frac{WEIGHT-MIN_{weight}}{10}} + e^{-\frac{POWER}{100}} + \\
 & \left. \frac{1}{2} Transceiver + \frac{1}{2} AdaptiveThr + FullSpikes \right) / 8 \times 100\%
 \end{aligned} \tag{7.7}$$

TABLE 7.1 – Comparaison des *headstages* présentés dans ce mémoire avec d'autres systèmes dotés de compression des signaux neuronaux. Les résultats de la fonction de coût sont présentés dans la dernière colonne du tableau afin de pouvoir comparer directement les performances de chacun des systèmes.

Systèmes tirés de	Type	# de canaux d'enregistrements	# de canaux de stimulation	Compression supplémentaire	Détection des AP	Seuil-Écart-type	Résolution (bits)	Échantillonnage (ksps)	Poids (g)	Puissance, sans stimulation (mW)	Type de transmetteur	Fonction de coût [%]
2 canaux	COTS	2	2	Aucune	Oui	Écart-type	12	20	4.9	43.0	Commercial (GFSK)	60.2
2 canaux, commercial	COTS	2	1	Aucune	Oui	Écart-type	12	20	1.1	43.0	Commercial (GFSK)	62.6
32 canaux	COTS	32	32	Ondelette	Oui	Écart-type	16	20	2.8	160	Commercial (GFSK)	75.6
[39]	COTS	1	1	Aucune	Oui	Fixe	N/A	11.7	8.4	40	Custom (Infrarouge)	36.1
[54]	COTS	16	0	Aucune Faible	Oui	Fixe	10	25	16.5	N/A	Bluetooth	51.7
[121]	COTS	96	0	quantification	Oui	Moyenne	8	31.25	17	100	Commercial (ASK)	56.6
[122]	ASIC/COTS	8	0	Aucune	Oui	RMS	12	16.67	23	4.62	ZigBee	52.9
[16]	ASIC	16	0	None	Oui	Fixe	8	30	Puce	2.21	Externe	48.6
[41]	ASIC	64	0	Occurrence des AP	Oui	Fixe	8	7.8	Puce	14.4	Custom (FSK)	29.4
[44]	ASIC	64	0	Occurrence des AP	Oui	Moyenne	8	24	Puce	0.130	Externe	45.6
[42]	ASIC	100	0	Occurrence des AP	Oui	Fixe	10	15	Puce	13.5	Custom (FSK)	36.2
[43]	ASIC	128	0	Caractéristiques de AP	Oui	Fixe	9	40	Puce	6.0	Custom (UWB)	47.1
[45]	ASIC	256	0	Caractéristiques de AP	Oui	Écart-type	5	20	Puce	5.4	Externe	40.4
[46]	ASIC	16	0	Delta	Non	N/A	N/A	N/A	Puce	0.22	Externe	61.6
[47]	ASIC	32	0	Ondelette	Non	N/A	12	25	Puce	0.076	Externe	64.7

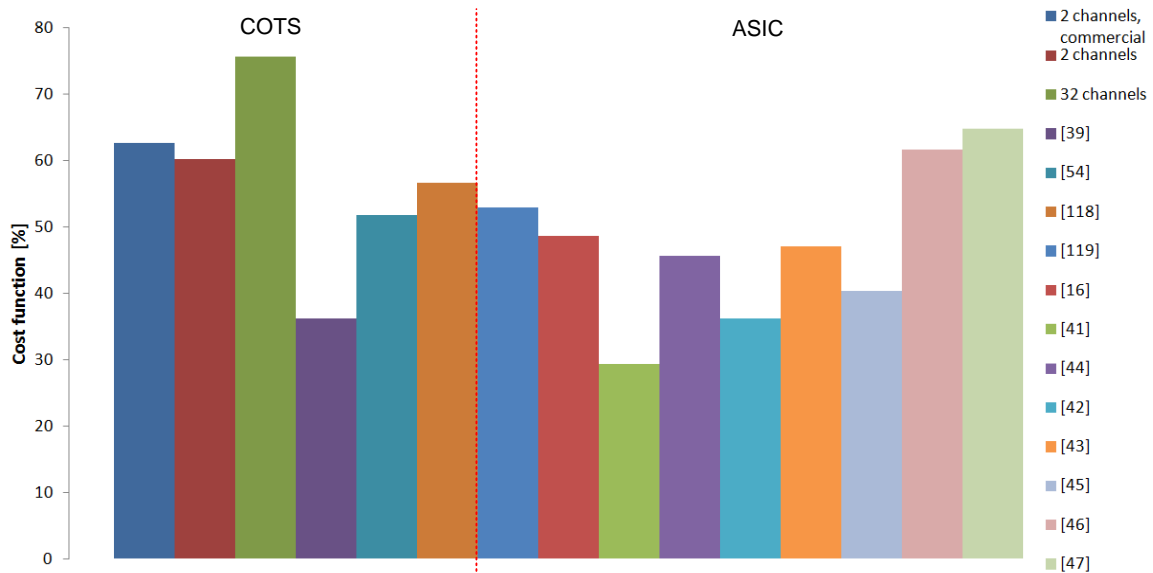


FIGURE 7.1 – Histogramme des résultats de la fonction de coût pour tous les systèmes répertoriés dans le tableau 7.1.

À la lumière des caractéristiques des différents systèmes répertoriés et selon les résultats de la fonction de coût, il est possible d'affirmer que le système à 32 canaux présenté dans ce mémoire surpasse tous les autres systèmes similaires. Le système à 32 canaux obtient une note de 75.6% alors que le système le plus proche, [47], obtient une note de 64.7%, soit 10.9% en dessous. Les deux systèmes à 2 canaux présentés dans ce mémoire obtiennent respectivement la 3<sup>eme</sup> et la 5<sup>eme</sup> places en tenant compte de tous les systèmes et la 2<sup>eme</sup> et 3<sup>eme</sup> place en considérant seulement les systèmes à base de composants COTS. Il était prévisible que certains systèmes de type ASIC allaient bien performer grâce à leur grand nombre de canaux et leur faible consommation électrique. Par contre, ces derniers ne possèdent pas de mécanisme de stimulation optique et ne proposent pas de solution adéquate pour la transmission sans fil, en utilisant soit des transmetteurs externes à haut débit ou des transmetteurs microélectroniques faits sur mesure. De plus, les systèmes ASIC ne forment pas un système complet et doivent être incorporés sur un PCB avec d'autres composants, augmentant ainsi leur poids et leur consommation électrique. Parmi tous les systèmes répertoriés dans le tableau 7.1, seulement les *headstages* présentés dans ce mémoire ont fait l'objet d'une validation *in vivo*, alors que la plupart des systèmes de type ASIC n'ont pas été incorporés dans des systèmes complets testables et que leurs caractéristiques/résultats ne sont que théoriques.



# Conclusion

Ce chapitre présente les contributions du travail de recherche ainsi que les travaux futurs pouvant être menés afin d'améliorer les algorithmes et/ou les systèmes développés.

## Contributions

Les contributions de ce travail de recherche vont de la création de nouveaux algorithmes permettant de compresser les signaux neuronaux, au développement de nouveaux outils de recherche optogénétiques qui permettent d'effectuer de nouvelles expériences sur de petits rongeurs transgéniques évoluant en toute liberté. Les contributions sont classées en fonction du type de *headstage* développé.

### Systèmes à deux canaux

Les deux *headstage* optogénétiques sans fil présentés au chapitre 4 sont les premiers à être dotés de stimulation optique, d'enregistrement électrophysiologique ainsi que de détection *in situ* des AP.

Les deux *headstages* sont dotés d'un nouvel algorithme ultra performant de détection des AP basé sur un nouvel algorithme de seuillage automatique fonctionnant avec une boucle de contrôle capable d'estimer l'écart-type du bruit inclus dans les signaux neuronaux. L'implantation de cet algorithme est entièrement numérique et optimisée pour utiliser seulement des ressources logiques de base, et ce, afin de faciliter son déploiement sur puce dédiée (ASIC), sur puce logique programmable (FPGA), ou sur microcontrôleur.

L'algorithme de détection ainsi que son implémentation optimisée ont fait l'objet d'une déclaration d'invention auprès de l'agent de brevet de l'Université Laval (# de divulgation 01691-DIV-02). La déclaration d'invention a été déclarée brevetable par la société de valorisation *Sovar* de l'Université Laval le 17 juillet 2015 et un brevet provisoire aux États-Unis a été obtenu (# de référence : 000819-0303)

Les deux *headstages*, conçus sur PCB flexible rigide, sont compacts, légers (4.9 g et 1.1 g pour la version commerciale) et leur fonctionnement a été validé par des tests *in vivo* sur le cerveau

de souris transgéniques au CRIUSMQ.

La version pré-commerciale du *headstage* à deux canaux est le plus petit système optogénétique sans fil répertorié et sa commercialisation est prévue pour octobre 2015 par la compagnie *Doric Lenses*, Canada.

La version non commerciale du *headstage* à deux canaux a fait l'objet de plusieurs publications scientifiques, en voici la liste exhaustive :

– **Articles de conférence :**

- G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. De Koninck and B. Gosselin, "An Optimized Adaptive Spike Detector for Behavioural Experiments", *IEEE International Symposium On Circuits and Systems (ISCAS'16)*, Montréal Canada, Mai 2016 (pending review).
- G. Gagnon-Turcotte, M. Sawan and B. Gosselin, "Low-Power Adaptive Spike Detector Based on a Sigma-Delta Control Loop", *IEEE Annual Conference on Engineering in Medicine and Biology Society (EMBC'15)*, Milan Italy, Août 2015 (accepted paper).
- G. Gagnon-Turcotte, C.-O. Dufresne Camaro, A. Avakh Kisomi, R. Ameli and B. Gosselin, "A Wireless Multichannel Optogenetic Headstage With On-The-Fly Spike Detection", *IEEE International Symposium On Circuits and Systems (ISCAS'15)*, Lisbon Portugal, pp. 1758-1761, Mai 2015.
- G. Gagnon-Turcotte, C.O Dufresne-Camaro and B. Gosselin, "Comparison of Low-Power Biopotential Processors for On-The-Fly Spike Detection", *IEEE International Symposium On Circuits and Systems (ISCAS'15)*, pp. 802-805, Mai 2015.

– **Articles de journaux :**

- G. Gagnon-Turcotte, A. Avakh Kisomi, R. Ameli, C.O Dufresne-Camaro, Y. LeChasseur, J.L Neron, P. B. Bareil, P. Fortier, C. Bories, Y. De Koninck and B. Gosselin, "A Wireless Optogenetic Stimulator Headstage with Multichannel Neural Recording Capability", *Sensors*, vol. 15, no. 9, pp. 22776-22797, 2015.
- G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. De Koninck and B. Gosselin, "A Low-Power Adaptive Spike Detector for In-Vivo Behavioural Neuroscience Experiments", *Transactions on Neural Systems & Rehabilitation Engineering*, 2015 (pending review).

## **Systeme 32 canaux**

Le *headstage* optogénétique sans fil présenté au chapitre 5 est le premier à être capable de stimuler les neurones optiquement avec 32 DEL haute puissance tout en étant capable de détecter et de compresser les AP à partir des signaux provenant de 32 microélectrodes.



Le *headstage* est doté du même algorithme puissant de détection des AP qui est incorporé dans les *headstages* à deux canaux, à la différence que l'implémentation optimisée sur FPGA effectue de la détection d'AP sur 32 canaux en parallèle.

Le nouvel algorithme de compression basé sur une DWT permet d'augmenter le nombre de canaux sous observation sans augmenter la consommation de l'émetteur-récepteur. Elle se distingue des méthodes de compression à base d'ondelettes existantes en permettant d'atteindre de meilleurs taux de compression tout en permettant une reconstruction des signaux compressés avec la même résolution que les données d'origine.

L'algorithme de compression ainsi que son implémentation optimisée ont fait l'objet d'une déclaration d'invention auprès de l'agent de brevet de l'Université Laval (# de divulgation 01736-DIV-01). La déclaration d'invention est en attente d'une réponse de la société de valorisation *Sovar* de l'Université Laval.

Le *headstage* permet de stimuler optiquement avec jusqu'à 32 DEL haute puissance simultanément, ce qui lui permet d'être compatible avec des réseaux d'optrodes [123] pour des applications optogénétiques futures.

La construction du *headstage*, à base d'un PCB flexible rigide, est compacte ( $15 \times 17 \times 20 \text{ mm}^3$ ), légère (2.8 g, masse de loin inférieure aux systèmes optogénétiques sans fil multicanaux existant [17], [124]) et son fonctionnement a été validé par des tests *in vivo* sur le cerveau d'une souris transgénique au CRIUSMQ.

Le *headstage* a fait l'objet de plusieurs publications scientifiques, en voici la liste exhaustive :

– **Articles de conférence :**

- G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. De Koninck and B. Gosselin, "A Wireless Optogenetic Headstage with Multichannel Neural Signal Compression", *IEEE Conference on Biomedical Circuits and System (BioCAS'15), Atlanta USA, Octobre 2015 (accepted paper)*.
- G. Gagnon-Turcotte and B. Gosselin, "Multichannel Spike Detector With an Adaptive Threshold Based On a Sigma-Delta Control Loop", *IEEE Annual Conference on Engineering in Medicine and Biology Society (EMBC'15), Milan Italy, Août 2015 (accepted paper)*.

– **Articles de journaux :**

- G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. Messaddeq, Y. De Koninck and B. Gosselin, "A Wireless Optogenetic Headstage with 32 Channels Neural Recording Capability", *IEEE Transactions on Biomedical Circuits and Systems, 2015 (pending review)*.

## Documents supplémentaires

Le code source, les schémas de circuits et les fichiers de PCB *Altium* des *headstages* sont disponibles à l'adresse suivante : [https://www.dropbox.com/sh/rid15fhohsbmxx6/AADnkst2Hfz21YyL\\_Tz9JHSVa?dl=0](https://www.dropbox.com/sh/rid15fhohsbmxx6/AADnkst2Hfz21YyL_Tz9JHSVa?dl=0). De plus, ce dossier comprend le code *Matlab* et les fichiers ayant servi pour obtenir les résultats de simulations présentées au chapitre 3, ainsi que pour les résultats de simulations présentés n'ayant pas fait l'objet d'une publication scientifique.

## Travaux futurs

Comme pour tous les travaux de recherche en ingénierie et en sciences, il est toujours possible d'améliorer ce qui a été développé.

### Tests *in vivo*

À cause de contraintes temporelles, il a été impossible de présenter dans ce travail un module implantable doté de 32 microélectrodes et de 32 DEL de stimulation optique afin de valider *in vivo* le *headstage* à 32 canaux à son plein potentiel. Un travail futur consisterait à développer un module implantable doté de 32 canaux d'enregistrement/stimulation qui pourrait être déposé sur le cortex du cerveau d'un animal transgénique. Ce type de module pourrait être réalisé à base de microélectrodes et de DEL standards ou bien à base de DEL et de microélectrodes micromachinées sur puce de silicium [123], [23].

### Améliorations

Une amélioration matérielle consisterait à augmenter le nombre de canaux d'enregistrement en augmentant le taux de compression des AP et en changeant/dupliquant le circuit d'acquisition neuronal du *headstage* à 32 canaux. Par exemple, il serait possible d'augmenter le CR du *headstage* à 8 et d'utiliser parallèlement 2 puces d'enregistrement électrophysiologiques RHD2132 en parallèle afin d'obtenir 64 canaux d'enregistrement.

Dans le *headstage* à 32 canaux, l'algorithme de détection des AP et l'algorithme de compression par ondelettes ont été implémentés dans un FPGA. Il serait possible de réduire la consommation électrique engendrée par l'utilisation d'un FPGA en implémentant ces algorithmes dans un système ASIC. Un système ASIC idéal incorporerait l'électronique d'acquisition et de compression des signaux neuronaux, l'électronique de stimulation optique sur plusieurs canaux ainsi qu'un transmetteur sans fil à basse consommation énergétique.

# Annexe A

## Résultats détaillés des simulations

Cette section présente les résultats détaillés des simulations *Matlab* présentées au chapitre 3.

### A.1 Ondelettes

#### A.1.1 Haar

TABLE A.1 – Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés avec ondelette de type *Haar*.

CR	# bits quant	# coefs	$SNDR_{moy}$ [dB]	$SNDR_{min}$ [dB]	$SNDR_{max}$ [dB]	déviatiion moyenne [%]
1.96	8	42	39.39	32.76	44.04	3.08
2.97	7	29	30.27	22.35	37.08	4.86
3.96	6	23	25.27	17.81	32.01	4.99
4.92	5	20	21.69	15.76	26.62	4.75
6.10	5	14	18.60	11.50	23.23	6.13
7.11	4	13	16.45	10.64	20.87	5.59
8	4	10	14.62	8.54	19.59	5.72

TABLE A.2 – Calcul du nombre de *clusters* avant compression (*clusters* originaux), après compression/décompression (*clusters* compressés), avec AP compressés/décompressés joint aux AP originaux (*clusters* mélangés) pour la compression avec ondelette de type *Haar*.

CR	# bits quant	# coefs	# <i>clusters</i> originaux	# <i>clusters</i> compressés	# <i>clusters</i> mélangés
1.96	8	42	4	4	4
2.97	7	29	4	4	4
3.96	6	23	4	4	4
4.92	5	20	4	4	4
6.10	5	14	4	4	4

7.11	4	13	4	4	4
8	4	10	4	4	4

TABLE A.3 – Déviation moyenne des AP compressés/décompressés par rapport au centre des *clusters* originaux, pourcentage d’association au bon *cluster* pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d’association au bon *cluster* pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l’algorithme de compression par ondelette de type *Haar*.

CR	# bits quant	# coefs	déviation moyenne [%]	regroupe- ment joint [%]	regroupe- ment unitaire [%]
1.96	8	42	0.0105	100	99.97
2.97	7	29	0.0517	100	99.97
3.96	6	23	0.139	99.97	99.88
4.92	5	20	0.135	99.92	99.91
6.10	5	14	0.414	99.86	99.91
7.11	4	13	0.343	99.81	99.81
8	4	10	1.48	99.66	99.75

### A.1.2 *Symmlet-2*

TABLE A.4 – Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés avec ondelette de type *Symmlet-2*.

CR	# bits quant	# coefs	$SNDR_{moy}$ [dB]	$SNDR_{min}$ [dB]	$SNDR_{max}$ [dB]	déviation moyenne [%]
1.96	8	42	39.38	35.67	43.81	2.40
2.97	7	29	33.60	29.0	37.27	3.13
3.96	6	23	28.19	20.56	32.75	4.58
4.92	5	20	23.11	17.32	27.14	4.80
6.10	5	14	20.50	12.95	26.28	8.29
7.11	4	13	17.41	11.25	22.47	7.29
8	4	10	15.42	8.07	22.47	10.31

TABLE A.5 – Calcul du nombre de *clusters* avant compression (*clusters* originaux), après compression/décompression (*clusters* compressés), avec AP compressés/décompressés joint aux AP originaux (*clusters* mélangés) pour la compression avec ondelette de type *Symmlet-2*.

CR	# bits quant	# coefs	# clusters originaux	# clusters compressés	# clusters mêlés
1.96	8	42	4	4	4
2.97	7	29	4	4	4
3.96	6	23	4	4	4
4.92	5	20	4	4	4
6.10	5	14	4	4	4
7.11	4	13	4	4	4
8	4	10	4	4	4

TABLE A.6 – Déviation moyenne des AP compressés/décompressés par rapport au centre des *clusters* originaux, pourcentage d’association au bon *cluster* pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d’association au bon *cluster* pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l’algorithme de compression par ondelette de type *Symmlet-2*.

CR	# bits quant	# coefs	déviation moyenne [%]	regroupe- ment joint [%]	regroupe- ment unitaire [%]
1.96	8	42	0.0114	100	100
2.97	7	29	0.0517	100	100
3.96	6	23	0.139	99.97	99.97
4.92	5	20	0.135	99.92	99.94
6.10	5	14	0.414	99.86	99.75
7.11	4	13	0.343	99.81	99.74
8	4	10	0.836	99.77	99.81

## A.2 Encodage *Delta*

TABLE A.7 – Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés avec l’algorithme *Delta*.

CR	# bits quant	$SNDR_{moy}$ [dB]	$SNDR_{min}$ [dB]	$SNDR_{max}$ [dB]	déviation moyenne [%]
1.96	8	39.94	16.09	62.41	12.37
2.59	6	34.67	16.10	51.33	9.03
3.84	4	26.34	15.99	39.41	18.94
5.05	3	20.78	11.57	34.83	27.96
7.38	2	13.40	3.84	27.45	41.27

TABLE A.8 – Calcul du nombre de *clusters* avant compression (*clusters* originaux), après compression/décompression (*clusters* compressés), avec AP compressés/décompressés joint aux AP originaux (*clusters* mélangés) pour l’algorithme de compression *Delta*.

CR	# bits quant	# <i>clusters</i> originaux	# <i>clusters</i> compressés	# <i>clusters</i> mélangés
1.96	8	4	4	4
2.59	6	4	4	4
3.84	4	4	4	4
5.05	3	4	4	4
7.38	2	4	4	4

TABLE A.9 – Déviation moyenne des AP compressés/décompressés par rapport au centre des *clusters* originaux, pourcentage d’association au bon *cluster* pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d’association au bon *cluster* pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l’algorithme de compression *Delta*.

CR	# bits quant	déviation moyenne [%]	regroupe- ment joint [%]	regroupe- ment unitaire [%]
1.96	8	0.0123	99.94	100
2.59	6	0.0142	99.97	99.97
3.84	4	0.0037	99.95	99.94
5.05	3	0.113	99.89	99.88
7.38	2	0.881	99.41	99.41

### A.3 Diminuer le nombre de bits de quantification (DNBQ)

TABLE A.10 – Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés en diminuant le nombre de bits de quantification.

CR	# bits quant	$SNDR_{moy}$ [dB]	$SNDR_{min}$ [dB]	$SNDR_{max}$ [dB]	déviation moyenne [%]
2	8	39.01	29.30	45.73	7.15
2.67	6	26.96	16.30	33.72	10.39
4	4	15.05	4.43	21.09	18.47
5.33	3	9.79	0.70	15.71	20.22
8	2	5.29	0	10.65	31.38

TABLE A.11 – Calcul du nombre de *clusters* avant compression (*clusters* originaux), après compression/décompression (*clusters* compressés), avec AP compressés/décompressés joint aux AP originaux (*clusters* mélangés) pour la compression obtenue en diminuant le nombre de bits de quantification.

CR	# bits quant	# <i>clusters</i> originaux	# <i>clusters</i> compressés	# <i>clusters</i> mélangés
2	8	4	4	4
2.67	6	4	4	4
4	4	4	4	4
5.33	3	4	4	4
8	2	4	4	4

TABLE A.12 – Déviation moyenne des AP compressés/décompressés par rapport au centre des *clusters* originaux, pourcentage d'association au bon *cluster* pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d'association au bon *cluster* pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue par compression en diminuant le nombre de bits de quantification.

CR	# bits quant	dévi- ation moyenne [%]	regroupe- ment joint [%]	regroupe- ment unitaire [%]
2	8	0.0027	100	99.97
2.67	6	0.0114	99.94	100
4	4	0.0041	99.68	99.81
5.33	3	0.703	99.46	99.01
8	2	5.97	98.67	98.39

## A.4 Compression de type *Compressed Sensing*

### A.4.1 Parcimonie dans le domaine des ondelettes (CSWD)

TABLE A.13 – Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés avec l'algorithme CS et parcimonie dans le domaine des ondelettes (*Daubechie-8*).

CR	# bits quant	# coefs	$SNDR_{moy}$ [dB]	$SNDR_{min}$ [dB]	$SNDR_{max}$ [dB]	dévi- ation moyenne [%]
2.04	8	45	28.70	12.85	42.23	16.27
3	8	30	13.52	0.158	34.42	38.31
4	8	22	9.44	-0.181	33.04	54.66
5.05	8	17	4.19	-1.36	19.85	61.09

6	8	14	3.47	-1.52	12.79	72.33
7.38	8	11	0.86	-2.05	13.44	133.72
8	8	10	1.90	-2.62	12.15	141.1

TABLE A.14 – Calcul du nombre de *clusters* avant compression (*clusters* originaux), après compression/décompression (*clusters* compressés), avec AP compressés/décompressés joint aux AP originaux (*clusters* mélangés) pour la compression avec l’algorithme CS et parcimonie dans le domaine des ondelettes (*Daubechie-8*).

CR	# bits quant	# coefs conservés	# <i>clusters</i> originaux	# <i>clusters</i> compressés	# <i>clusters</i> mélangés
2.04	8	45	4	4	4
3	8	30	4	4	4
4	8	22	4	4	4
5.05	8	17	4	3	4
6	8	14	4	3	3
7.38	8	11	4	2	3
8	8	10	4	2	3

TABLE A.15 – Déviation moyenne des AP compressés/décompressés par rapport au centre des *clusters* originaux, pourcentage d’association au bon *cluster* pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d’association au bon *cluster* pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l’algorithme de compression CS et parcimonie dans le domaine des ondelettes (*Daubechie-8*).

CR	# bits quant	# coefs conservés	déviation moyenne [%]	regroupement joint [%]	regroupement unitaire [%]
2.04	8	45	0.344	99.98	100
3	8	30	2.89	99.51	99.66
4	8	22	4.23	99.26	97.13
5.05	8	17	7.52	75.02	97.47
6	8	14	8.96	81.45	68.55
7.38	8	11	9.50	67.38	55.14
8	8	10	9.30	86.93	67.56

#### A.4.2 Base parcimonieuse optimale (CSOP)



TABLE A.16 – Calcul du SNDR moyen, minimum, maximum et de la déviation moyenne des signaux compressés/décompressés avec l’algorithme CS et matrice de parcimonie optimale.

CR	# bits quant	# coefs	$SNDR_{moy}$ [dB]	$SNDR_{min}$ [dB]	$SNDR_{max}$ [dB]	déviation moyenne [%]
2.04	8	45	35.1	16.80	45	10.88
3	8	30	34.74	16.20	44.91	6.99
4	8	22	30.43	4.15	44.66	13.97
5.05	8	17	24.60	-1.11	40.10	18.50
6	8	14	23.20	-1.11	38.72	18.28
7.38	8	11	19.31	-1.47	36.01	26.15
8	8	10	15.99	-1.98	34.65	29.33

TABLE A.17 – Calcul du nombre de *clusters* avant compression (*clusters* originaux), après compression/décompression (*clusters* compressés), avec AP compressés/décompressés joint aux AP originaux (*clusters* mélangés) pour la compression avec l’algorithme CS et matrice de parcimonie optimale.

CR	# bits quant	# coefs conservés	# <i>clusters</i> originaux	# <i>clusters</i> compressés	# <i>clusters</i> mélangés
2.04	8	45	4	4	4
3	8	30	4	4	4
4	8	22	4	4	4
5.05	8	17	4	4	4
6	8	14	4	3	4
7.38	8	11	4	3	3
8	8	10	4	5	5

TABLE A.18 – Déviation moyenne des AP compressés/décompressés par rapport au centre des *clusters* originaux, pourcentage d’association au bon *cluster* pour les AP compressés/décompressés ajoutés un à un aux AP originaux (regroupement unitaire), pourcentage d’association au bon *cluster* pour les AP compressés/décompressés joint aux AP originaux (regroupement joint) obtenue avec l’algorithme de compression CS et matrice de parcimonie optimale.

CR	# bits quant	# coefs conservés	déviation moyenne [%]	regroupe- ment joint [%]	regroupe- ment unitaire [%]
2.04	8	45	0.0072	99.97	99.97
3	8	30	0.0304	99.91	99.97
4	8	22	0.0371	99.88	99.81
5.05	8	17	0.191	99.23	99.29
6	8	14	0.0885	99.74	99.51
7.38	8	11	0.195	98.90	99.23

8

8

10

0.462

98.84

97.65

---

## Annexe B

# Résultats détaillés obtenus avec le *headstage* à 32 canaux

### B.1 Tests reliés au PSNR

Cette section présente les résultats de compression détaillés obtenus directement avec le *headstage* à 32 canaux et/ou à partir de simulations Matlab effectuées sur les signaux acquisitionnés par le *headstage*.

TABLE B.1 – Résultats du SNDR moyen, minimum, maximum, ainsi que l'écart-type pour des signaux ayant une SNR variant de 25 dB à 0 dB. Les résultats ont été obtenus à partir des AP détectés, compressés et transmis par le *headstage* à 32 canaux.

SNR [dB]	PSNR moyen [dB]	PSNR minimum [dB]	PSNR maximum [dB]	Écart-type [dB]
25	22.15	18.3	25.64	1.06
20	21.23	12.82	24.95	1.28
15	21.62	15.18	25.63	1.10
10	20.09	13.60	24.40	1.31
5	16.09	11.74	21.01	1.68
3	15.56	11.40	20.30	1.47
1	16.15	10.38	21.68	1.84
0	15.33	10.33	21.38	1.77

TABLE B.2 – PSNR moyen, minimum, maximum ainsi que l'écart-type relatif au PSNR moyen pour l'algorithme de compression par ondelettes avec maximisation dynamique de la quantification. Ces résultats ont été obtenus avec des signaux neuronaux synthétiques acquisitionnés avec le *headstage* à 32 canaux et importés dans *Matlab*.

CR	PSNR moyen [dB]	PSNR minimum [dB]	PSNR maximum [dB]	Écart-type relatif au PSNR moyen [%]
3	30.88	23.16	37.03	4.44
4	23.93	18.66	30.93	5.68
5	21.75	16.40	26.54	5.47
6	18.33	11.96	24.02	6.93
7	16.11	10.70	21.58	6.89
8	14.11	8.38	19.29	7.94

TABLE B.3 – PSNR moyen, minimum, maximum ainsi que l'écart-type relatif au PSNR moyen pour l'algorithme de compression par ondelettes sans maximisation dynamique de la quantification. Ces résultats ont été obtenus avec des signaux neuronaux synthétiques acquisitionnés avec le *headstage* à 32 canaux et importés dans *Matlab*.

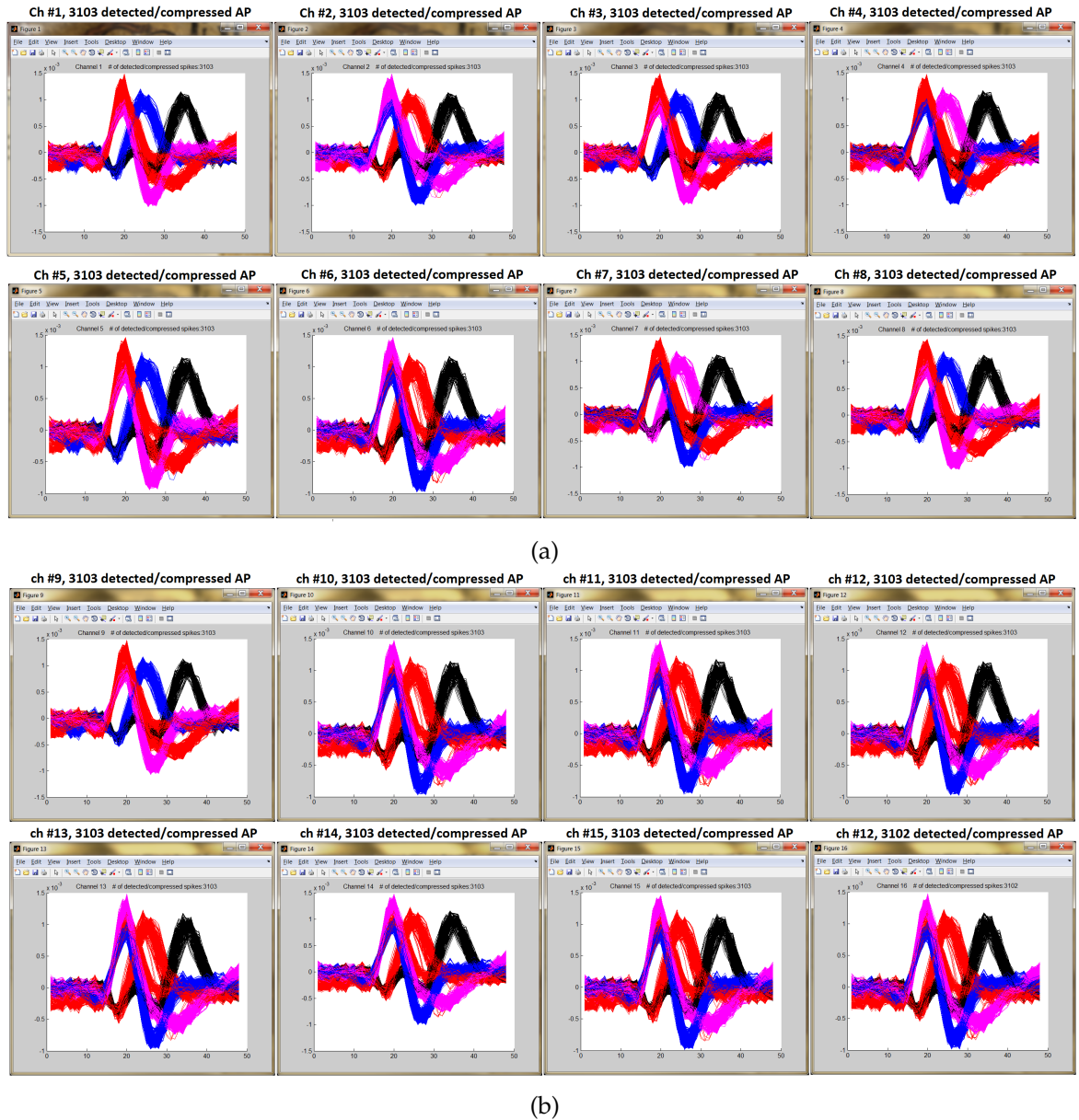
CR	PSNR moyen [dB]	PSNR minimum [dB]	PSNR maximum [dB]	Écart-type relatif au PSNR moyen [%]
3	17.34	8.19	22.63	10.03
4	12.08	3.52	17.72	14.82
5	7.76	0.363	13.08	17.91
6	7.76	0.363	13.08	17.91
7	4.66	0	8.01	24.68
8	4.66	0	8.01	24.68

TABLE B.4 – PSNR moyen, minimum, maximum ainsi que l'écart-type relatif au PSNR moyen pour l'algorithme de compression *Delta*. Ces résultats ont été obtenus avec des signaux neuronaux synthétiques acquisitionnés avec le *headstage* à 32 canaux et importés dans *Matlab*.

CR	PSNR moyen [dB]	PSNR minimum [dB]	PSNR maximum [dB]	Écart-type relatif au PSNR moyen [%]
3	26.19	6.69	45.08	18.67
4	21.50	6.70	34.83	12.79
5	17.15	7.11	27.71	18.83
7	11.84	3.82	22.21	36.99

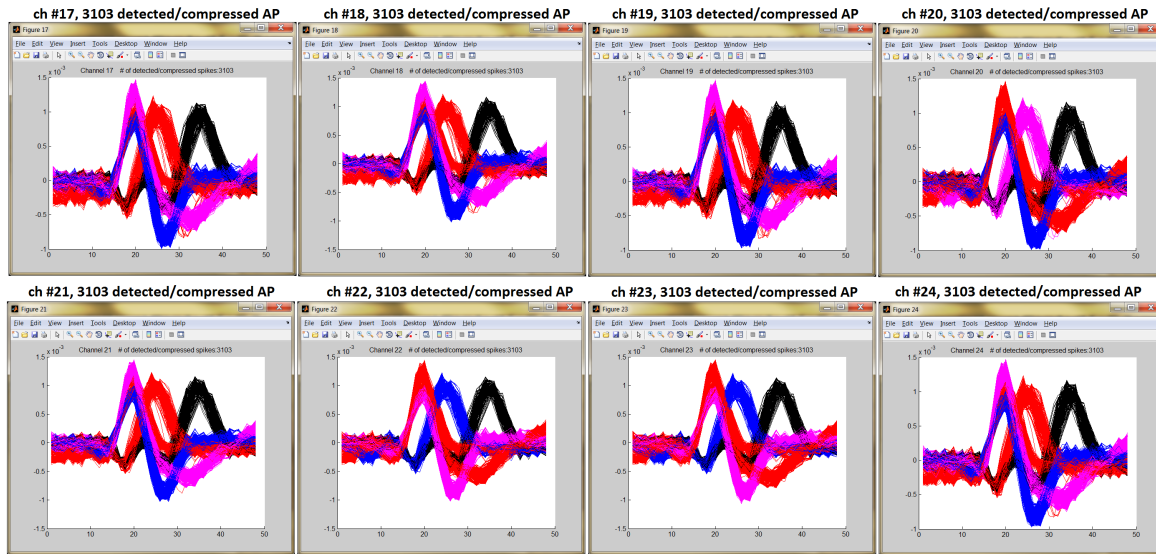
TABLE B.5 – PSNR moyen, minimum, maximum ainsi que l'écart-type relatif au PSNR moyen pour l'algorithme de compression CS avec matrice de parcimonie optimale. Ces résultats ont été obtenus avec des signaux neuronaux synthétiques acquisitionnés avec le *headstage* à 32 canaux et importés dans *Matlab*.

CR	PSNR moyen [dB]	PSNR minimum [dB]	PSNR maximum [dB]	Écart-type relatif au PSNR moyen [%]
3	30.59	12.04	41.93	9.41
4	24.51	5.08	36.61	14.20
5	20.57	1.86	35.31	18.96
6	18.37	1.61	32.27	21.94
7	14.62	-3.29	32.15	27.09
8	12.25	-7.71	28.72	32.16

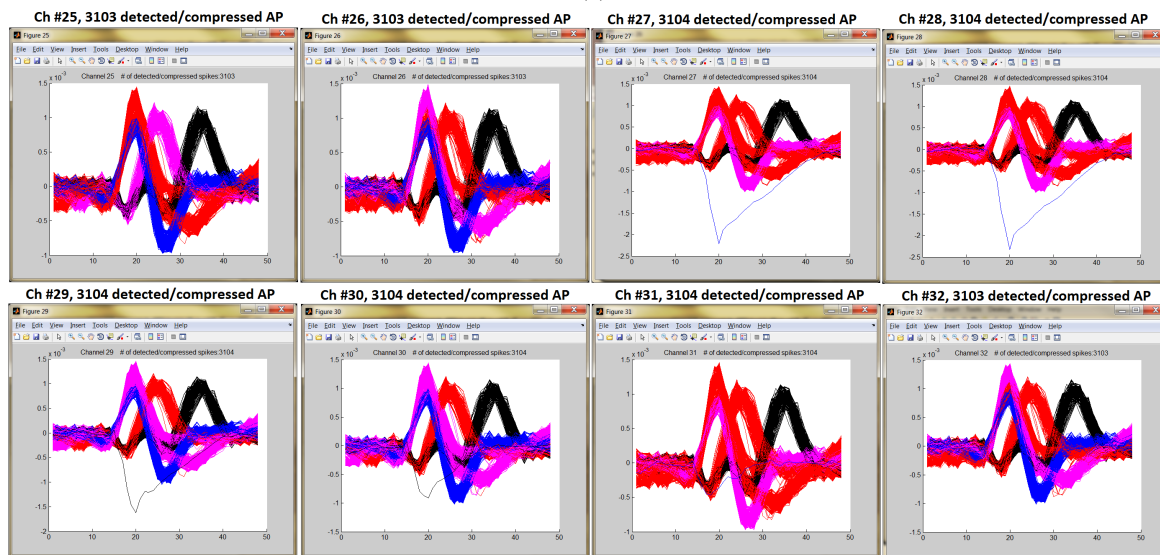


## B.2 Détection/compression sur 32 canaux simultanément

Cette section présente à la figure B.1 les résultats de détection du *headstage* avec les 32 canaux activés simultanément. Les résultats proviennent d'un test d'une durée de 60 secondes avec un signal neuronal synthétique dupliqué sur les 32 canaux du *headstage* afin de reproduire le pire cas où des AP sont détectés simultanément sur les 32 canaux. Le signal neuronal synthétique utilisé possède un *firing rate* de 50 AP/s et un SNR de 10 dB. Tous les canaux ont détecté 3103 AP  $\pm 1$ , ce qui démontre que le *headstage* peut détecter et compresser les signaux issus de 32 canaux en parallèle. Le taux de compression atteint lors de cet essai est de 17.20.



(c)



(d)

FIGURE B.1 – AP détectés et compressés par le *headstage* à partir des 32 canaux simultanément (signal neuronal synthétique dupliqué sur tous les canaux), a) canaux de 1 à 8, b) canaux de 9 à 16, c) canaux de 17 à 24, d) canaux de 25 à 32.

## Annexe C

# Cartes de développement et station de base

### C.1 Cartes de développement

Les *headstages* présentés dans ce mémoire ont été fabriqués à partir de cartes de développement qui ont servies à valider leur fonctionnement. Ces cartes ont été conçues dans le seul but de valider le fonctionnement des *headstages* avant leur fabrication sur PCB flexible rigide. Cette section montre les deux cartes de développement ayant servi à développer le *headstage* à deux canaux présenté au chapitre 4 (figure C.1) et le *headstage* à 32 canaux présenté au chapitre 5 (figure C.2).

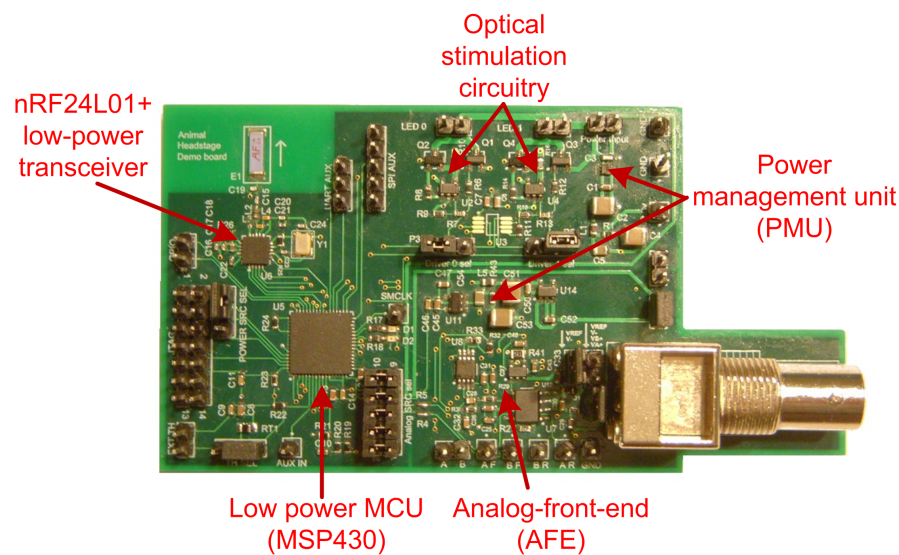


FIGURE C.1 – Prototypé de développement du *headstage* à deux canaux présenté au chapitre 4.

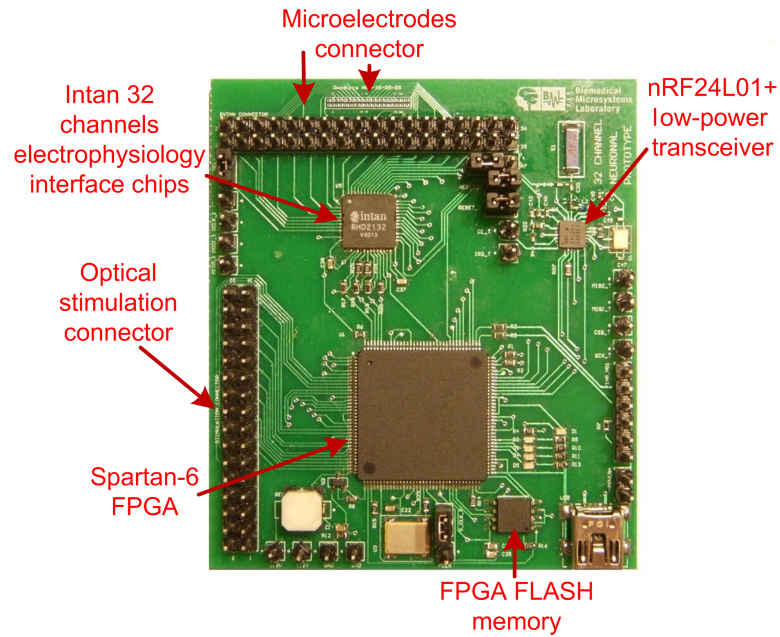


FIGURE C.2 – Prototypé de développement du *headstage* à 32 canaux présenté au chapitre 5.

## C.2 Station de base

Cette section montre à la figure C.3 la station de base utilisée afin de transférer les données transmises par les *headstages* à un ordinateur. Cette station de base est compatible avec tous les *headstages* présentés dans ce mémoire.

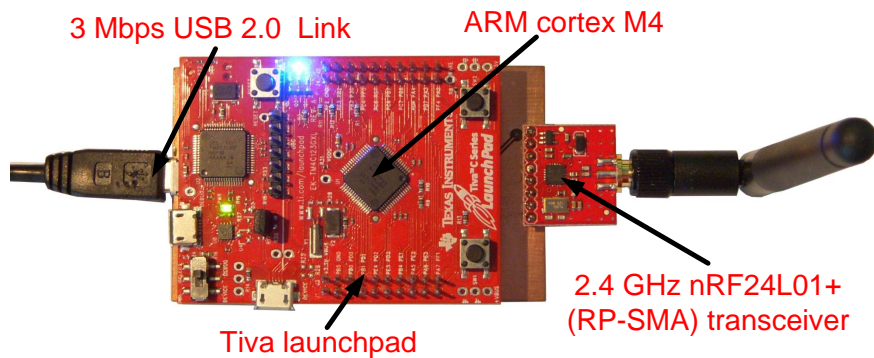


FIGURE C.3 – Station de base compatible avec tous les *headstages*.



# Annexe D

## Interfaces usager

### D.1 Interface systèmes à 2 canaux

L'interface d'acquisition des *headstages* à 2 canaux est présentée à la figure D.1. Cette interface est codée entièrement en langage Qt/C++.

### D.2 Interface système à 32 canaux

L'interface d'acquisition du *headstage* à 32 canaux est présentée à la figure D.2. Cette interface est codée entièrement en langage C#.

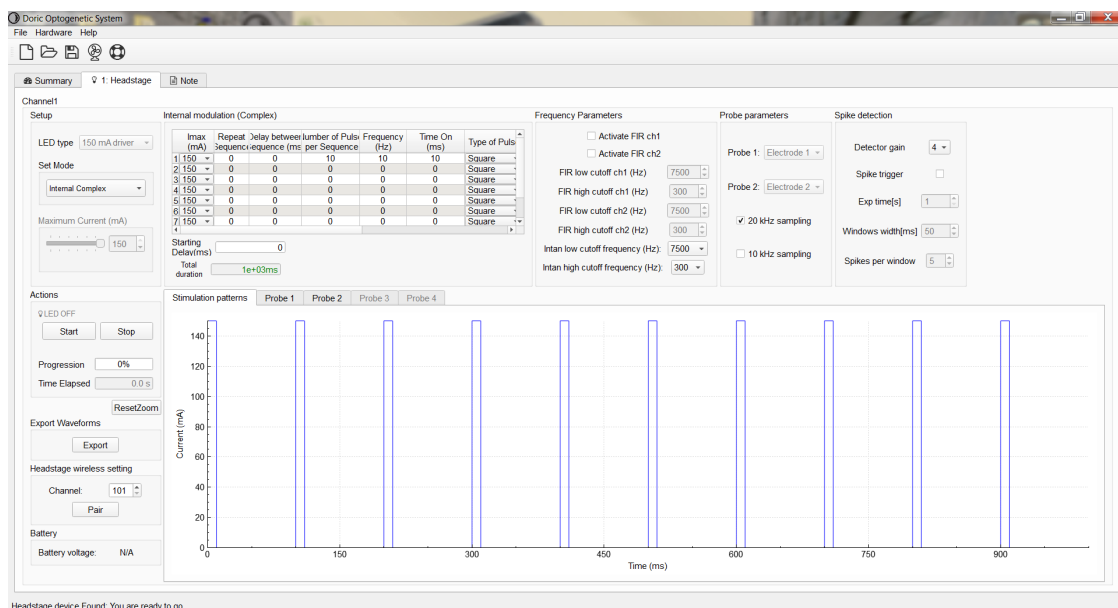


FIGURE D.1 – Interface d'acquisition des *headstages* à 2 canaux.

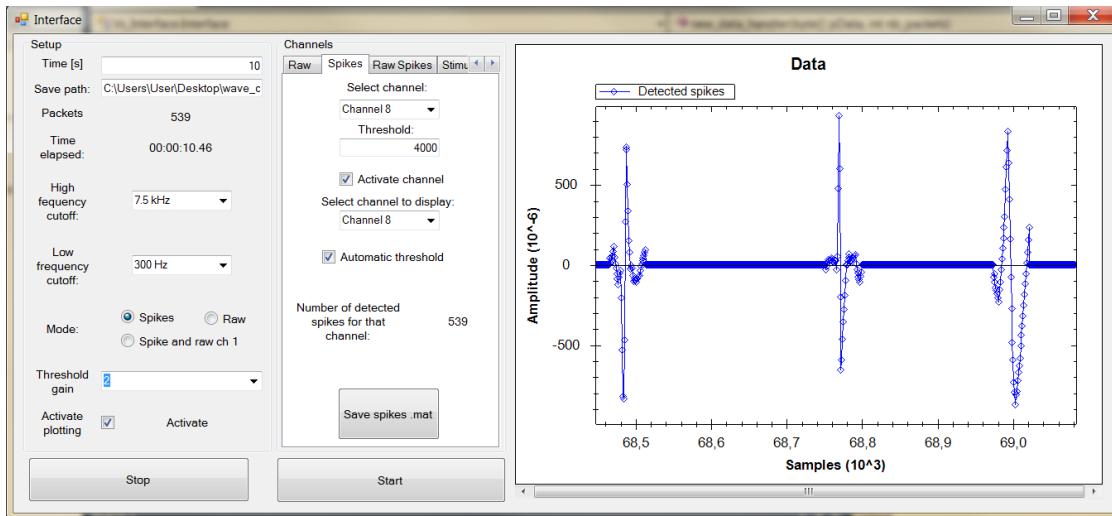


FIGURE D.2 – Interface d’acquisition du *headstage* à 32 canaux.

# Bibliographie

- [1] K. Deisseroth, "Optogenetics," *Nature Methods*, vol. 8, no. 1, pp. 26–29, 2011.
- [2] S. C. P. Williams and K. Deisseroth, "Optogenetics," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 110, no. 41, pp. 16 287–16 287, 2013.
- [3] "Statistiques de l'Institut de Recherche en Santé du Canada (IRSC)," <http://www.cihr-irsc.gc.ca/f/47914.html>, accessed : 2015-07-02.
- [4] "White House BRAIN Initiative," <https://www.whitehouse.gov/share/brain-initiative>, accessed : 2015-07-02.
- [5] "Human Brain Project," <https://www.humanbrainproject.eu/>, accessed : 2015-07-02.
- [6] "Fonds Canadien de Recherche sur le Cerveau," <http://can-acn.org/fr/canada-brain>, accessed : 2015-07-02.
- [7] S. A. Mirbozorgi *et al.*, "A smart cage with uniform wireless power distribution in 3d for enabling long-term experiments with freely moving animals," *IEEE Transactions on Biomedical Circuits and Systems*, 2015 (to be published).
- [8] G. M. Shepherd, "Neurobiology," O. U. Press, Ed., 1994, p. 760.
- [9] O. Sporns, "Networks of the brain," M. Press, Ed., 2010.
- [10] B. J. Sadock and V. A. Sadock, "Kaplan and sadock's synopsis of psychiatry : Behavioral sciences/clinical psychiatry," L. W. . Wilkins, Ed., 2007.
- [11] E. R. Kandel, J. H. Schwartz, T. M. Jessell, S. A. Siegelbaum, and A. J. Hudspeth, "Principles of neural science," 2012.
- [12] "Leicester University Banque de signaux neuronaux," <http://www2.le.ac.uk/departments/engineering/research/bioengineering/neuroengineering-lab/spike-sorting>, accessed : 2015-02-17.
- [13] L. Liu *et al.*, "Neural recording front-end IC using action potential detection and analog buffer with digital delay for data compression," *IEEE Engineering in Medicine and Biology Society (EMBC'13)*, pp. 747–750, 2013.

- [14] B. Ulrich and O. Maurits, "A neural recorder with mixed signal feedback loop for a spectral separation of spikes and LFP," *Conference on Ph.D. Research in Microelectronics and Electronics (PRIME'12)*, pp. 1–4, 2012.
- [15] B. Gosselin *et al.*, "A low-power integrated bioamplifier with active low-frequency suppression," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 1, no. 3, pp. 184–192, 2007.
- [16] B. Gosselin *et al.*, "A mixed-signal multichip neural recording interface with bandwidth reduction," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 3, no. 3, pp. 129–141, 2009.
- [17] R. Ameli *et al.*, "A wireless and batteryless neural headstage with optical stimulation and electrophysiological recording," *IEEE Annual International Conference of Engineering in Medicine and Biology Society (EMBC'13)*, pp. 5662–5665, 2013.
- [18] B. Gosselin, "Recent advances in neural recording microsystems," *Sensors*, vol. 11, no. 5, pp. 4572–4597, 2011.
- [19] "MicroProbes microélectrodes métalliques," <http://www.microprobes.com/index.php/products/metal-microelectrodes>, accessed : 2015-02-17.
- [20] "NeuroNexus microélectrodes flexibles de surface," <http://neuronexus.com/products/neural-probes/surface-probes>, accessed : 2015-02-17.
- [21] "MicroProbes tableau de microélectrodes linéaire sur silicium," <http://www.microprobes.com/index.php/products/multi-channel-arrays/lma-linear-microelectrode-arrays-mc>, accessed : 2015-02-17.
- [22] "MicroProbes tableau de microélectrodes métalliques," <http://www.microprobes.com/index.php/products/multi-channel-arrays/mea-micro-electrode-arrays>, accessed : 2015-02-17.
- [23] "BlackRock Microsystems tableau de microélectrodes sur silicium," <http://www.blackrockmicro.com/content.aspx?id=78>, accessed : 2015-02-17.
- [24] K. D. Harris *et al.*, "Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements," *Journal of Neurophysiology*, vol. 84, no. 1, pp. 401–414, 2000.
- [25] G. Buzsaki, "Large-scale recording of neuronal ensembles," *Nature Neuroscience*, vol. 7, no. 5, pp. 446–451, 2004.
- [26] J. Wells *et al.*, "Optical stimulation of neural tissue in vivo," *Optics Letters*, vol. 30, no. 5, pp. 504–506, 2005.

- [27] O. Yizhar *et al.*, "Optogenetics in neural systems," *Neurons*, vol. 71, no. 1, pp. 9–34, 2011.
- [28] "The Jackson Laboratory," <http://www.jax.org/>, accessed : 2015-08-17.
- [29] "Institut Universitaire en Santé Mentale de Québec, axe de recherche en neurosciences cellulaires et moléculaires," <http://www.crulrg.ulaval.ca/fr/recherche/axes/chercheurs/chercheur.php?id=61g/>, accessed : 2015-08-17.
- [30] "Triangle Biosystem International M-Series, systèmes d'enregistrement multicanal filaire," <http://www.trianglebiosystems.com/m-series-systems.html>, accessed : 2015-07-28.
- [31] "Triangle Biosystem International T-Series, headstage d'enregistrement multicanal filaire," <http://www.trianglebiosystems.com/t-series-systems.html>, accessed : 2015-07-28.
- [32] "BlackRock Microsystems CerePlex M, dispositif d'enregistrement neural à 96 électrodes," <http://www.blackrockmicro.com/content.aspx?id=38>, accessed : 2015-02-17.
- [33] "NeuroNexus SmartBox, système d'enregistrement de signaux électrophysiologiques câblé," <http://neuronexus.com/products/electrode-instrumentation>, accessed : 2015-07-28.
- [34] "Doric Lenses programmable LED driver, système de stimulation optogénétique câblé," <http://doriclenses.com/life-sciences/led-drivers/782-programmable-led-driver.html>, accessed : 2015-07-28.
- [35] "BlackRock Microsystems Optogenetics starter kit, système de stimulation optogénétique câblé," <http://www.blackrockmicro.com/content.aspx?id=138>, accessed : 2015-07-28.
- [36] C. T. Wentz *et al.*, "A wirelessly powered and controlled device for optical neural control of freely-behaving animals," *Journal of Neural Engineering*, vol. 8, no. 4, pp. 1741–2560, 2011.
- [37] T. Ativanichayaphong *et al.*, "A combined wireless neural stimulating and recording system for study of pain processing," *Journal of Neuroscience Methods*, vol. 170, no. 1, pp. 25–34, 2008.
- [38] X. Ye *et al.*, "A portable telemetry system for brain stimulation and neuronal activity recording in freely behaving small animals," *Journal of Neuroscience Methods*, vol. 174, no. 2, pp. 186–193, 2008.

- [39] J. Mavoori *et al.*, "An autonomous implantable computer for neural recording and stimulation in unrestrained primates," *Journal of Neuroscience Methods*, vol. 148, no. 1, pp. 71–77, 2005.
- [40] H. Miranda *et al.*, "HermesD : A high-rate long-range wireless transmission system for simultaneous multichannel neural recording applications," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 4, no. 3, pp. 181–191, 2010.
- [41] A. M. Sodagar *et al.*, "An implantable 64-channel wireless microsystem for single-unit neural recording," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 9, pp. 2591–2604, 2009.
- [42] R. R. Harrison *et al.*, "A low-power integrated circuit for a wireless 100-electrode neural recording system," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 1, pp. 123–133, 2007.
- [43] M. S. Chae *et al.*, "A 128-channel 6 mw wireless neural recording IC with spike feature extraction and UWB transmitter," *IEEE Transactions on Systems and Rehabilitation Engineering*, vol. 17, no. 4, pp. 312–21, 2009.
- [44] V. Karkare *et al.*, "A 130- $\mu$ W, 64-channel neural spike-sorting DSP chip," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 5, pp. 1214–1222, 2011.
- [45] R. H. Olsson and K. D. Wise, "A three-dimensional neural recording microsystem with implantable data compression circuitry," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 12, pp. 2796–2804, 2005.
- [46] V. Majidzadeh *et al.*, "A 16-channel 220  $\mu$ W neural recording IC with embedded delta compression," *IEEE Conference on Biomedical Circuits and Systems (BioCAS'11)*, pp. 9–12, 2011.
- [47] A. M. Kamboh, M. Raetz, K. G. Oweiss, and A. Mason, "Area-power efficient VLSI implementation of multichannel DWT for data compression in implantable neuroprosthetics," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 1, no. 2, pp. 128–135, 2007.
- [48] Y. Perelman and R. Ginosar, "An integrated system for multichannel neuronal recording with spike/LFP separation, integrated A/D conversion and threshold detection," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 1, pp. 130–7, 2007.
- [49] H. Gao *et al.*, "HermesE : A 96-channel full data rate direct neural interface in 0.13  $\mu$ CMOS," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 4, pp. 1043–1055, 2012.
- [50] D. S. Schregardus *et al.*, "A lightweight telemetry system for recording neuronal activity in freely behaving small animals," *Journal of Neuroscience Methods*, vol. 155, no. 1, pp. 62–71, 2006.

- [51] C. N. Chien and F. S. Jaw, "Miniature telemetry system for the recording of action and field potentials," *Journal of Neuroscience Methods*, vol. 147, no. 1, pp. 68–73, 2005.
- [52] H. M. Lee *et al.*, "A wireless implantable switched-capacitor based optogenetic stimulating system," *IEEE Annual International Conference of Engineering in Medicine and Biology Society (EMBC'14)*, pp. 878–881, 2014.
- [53] S. Roy and X. Wang, "Wireless multi-channel single unit recording in freely moving and vocalizing primates," *Journal of Neuroscience Methods*, vol. 203, no. 1, pp. 28–40, 2012.
- [54] R. E. Hampson, V. Collins, and S. A. Deadwyler, "A wireless recording system that utilizes bluetooth technology to transmit neural activity in freely moving animals," *Journal of Neuroscience Methods*, vol. 182, no. 2, pp. 195–204, 2009.
- [55] "Programme Mitacs de partenariat entre le milieu universitaire et l'industrie," <https://www.mitacs.ca/fr>, accessed : 2015-07-24.
- [56] I. Obeid and P. D. Wolf, "Evaluation of spike-detection algorithms for a brain-machine interface application," *IEEE Transactions on Bio-Medical Engineering*, vol. 51, no. 6, pp. 905–911, 2004.
- [57] B. Gosselin and M. Sawan, "An ultra low-power cmos automatic action potential detector," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17, no. 4, pp. 346–353, 2009.
- [58] B. Gosselin and M. Sawan, "Adaptive detection of action potentials using ultra low-power cmos circuits," *IEEE Biomedical Circuits and Systems Conference (BIOCAS'08)*, pp. 209–212, 2008.
- [59] V. Shalchyan, W. Jensen, and D. Farina, "Spike detection and clustering with unsupervised wavelet optimisation in extracellular neural recordings," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 91, pp. 2576–2585, 2012.
- [60] H. L. Chan *et al.*, "Detection of neuronal spikes using an adaptive threshold based on the maxmin spread sorting method," *Journal of Neuroscience Methods*, vol. 172, pp. 112–121, 2008.
- [61] R. R. Harrison, "A low-power integrated circuit for adaptive detection of action potentials in noisy signals," *IEEE Annual International of Engineering in Medicine and Biology Society (EMBC'03)*, vol. 4, pp. 3325–3328, 2003.
- [62] C. P. Breanne *et al.*, "Comparison of spike sorting and thresholding of voltage waveforms for intracortical brain-machine interface performance," *Journal of Neural Engineering*, vol. 12, no. 1, pp. 1741–2560, 2015.

- [63] G. D. J. Phillies, "Wavelets : a new alternative to fourier transforms," *Computers in physics*, vol. 10, no. 3, 1996.
- [64] A. Quotb, Y. Bornat, and S. Renaud, "Wavelet transform for real-time detection of action potentials in neural signals," *Frontiers in Neuroengineering*, 2011. doi : 10.3389/fneng.2011.00007
- [65] B. Vidakovic and P. Müller, "Wavelets for kids," I. D. -. Wavelet Resources, Duke University, Ed., 1991.
- [66] P. S. Addison, "Wavelet transforms and the ECG : a review," *Physiological Measurement*, vol. 26, no. 5, 2005. doi : 10.1088/0967-3334/26/5/R01
- [67] R. X. Gao and R. Yan, *Wavelets Theory and Applications for Manufacturing*, 2011.
- [68] P. S. Addison, *The illustrated wavelet transform handbook*, 2002.
- [69] S. G. Mallat, "A theory for multiresolution signal decomposition : the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.
- [70] A. Salmanpour, L. J. Brown, and J. K. Shoemaker, "Performance analysis of stationary and discrete wavelet transform for action potential detection from sympathetic nerve recordings in humans," *IEEE Engineering in Medicine and Biology Society (EMBC'08)*, pp. 2932–2935, 2008.
- [71] K. H. Kim and S. J. Kim, "Wavelet-based action potential detector for the extracellular neural signal with low signal-to-noise ratio," *IEEE Transactions on Biomedical Engineering*, vol. 50, no. 8, pp. 999–1011, 2003.
- [72] N. Boukhenoufa, K. Benmahammed, M. A. Abdi, and F. Djeflal, "Wavelet-based ECG signals compression using SPIHT technique and VKTP coder," *IEEE International Conference on Signals, Circuits and Systems (SCS'09)*, pp. 1–5, 2009.
- [73] G. H. Jeong and I. S. Lee, "Wavelet-based ECG compression using dynamic multi-stage vector quantization," *IEEE Conference on Industrial Electronics and Applications (ICIEA'09)*, pp. 2100–2105, 2009.
- [74] Y. Cai and B. Zhao, "Implementation and algorithm analysis of wavelet lifting scheme," *Cross Strait Quad-Regional Radio Science and Wireless Technology Conference (CS-QRWC)*, pp. 1418–1421, 2011.
- [75] K. E. Thomson *et al.*, "B-Spline vs. lifting architecture for neuroprosthetic devices," *IEEE International Conference on Electro/information Technology*, pp. 641–644, 2006.



- [76] A. Keuer *et al.*, "FPGA based time detection of spikes within neural signals," *Sensor Array and Multichannel Signal Processing Workshop Proceedings*, pp. 186–190, 2004.
- [77] K. Qureshi and V. P. Patel, "Efficient data compression of ECG signal using discrete wavelet transform," *International Journal of Research in Engineering and Technology (IJ-RET)*, vol. 02, no. 4, pp. 2319–1163, 2013.
- [78] M. Abo-Zahhad, *ECG Signal Compression Using Discrete Wavelet Transform*, 2011.
- [79] M. Abo-Zahhad, S. M. Ahmed, and A. Zakaria, "ECG signal compression technique based on discrete wavelet transform and QRS estimation," *Signal Processing : An International Journal (SPIJ)*, vol. 4, no. 2, pp. 138–160, 2010.
- [80] Y. Chompusri and S. Yimman, "Energy packing efficiency based threshold level selection for DWT ECG compression," *International Journal of Applied Biomedical Engineering*, vol. 2, no. 2, 2009.
- [81] S. Narasimhan, M. Tabib-Azar, H. J. Chiel, and S. B. Case, "Neural data compression with wavelet transform : A vocabulary based approach," *IEEE International Conference on Neural Engineering (EMBS'07)*, pp. 666–669, 2007.
- [82] G. C. Chang and Y. D. Lin, "An efficient lossless ECG compression method using delta coding and optimal selective huffman coding," *IFMBE Proceedings*, vol. 31, pp. 1327–1330, 2010.
- [83] Y. Ghanbari, P. Papamichalis, and L. Spence, "Robustness of neural spike sorting to sampling rate and quantization bit depth," *IEEE International Conference on Digital Signal Processing*, pp. 1–6, 2009.
- [84] "K-Means Clustering," <http://www.mathworks.com/help/stats/kmeans.html>, accessed : 2015-02-28.
- [85] F. Chen, A. P. Chandrakasan, and V. Stojanovi, "A signal-agnostic compressed sensing acquisition system for wireless and implantable sensors," *IEEE Conference on Custom Integrated Circuits (CICC'10)*, pp. 1–4, 2010.
- [86] C. Bulach, U. Bihl, and M. Ortmanns, "Evaluation study of compressed sensing for neural spike recording," *IEEE International Conference of Engineering in Medicine and Biology Society (EMBC'12)*, pp. 3507–3510, 2012.
- [87] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *The Journal of Machine Learning Research*, vol. 11, pp. 19–60, 2010.
- [88] J. Zhang *et al.*, "Reconstruction of neural action potentials using signal dependent sparse representations," *IEEE International Symposium on Circuits and Systems (IS-CAS'13)*, pp. 1520–1523, 2013.

- [89] "Algorithm K-SVD," <https://retarget-toolkit.googlecode.com/svn-history/r204/trunk/Matlab/Saliency/k-svd/ksvdbox13/ksvd.m>, accessed : 2015-02-28.
- [90] Z. Charbiwala *et al.*, "Compressive sensing of neural action potentials using a learned union of supports," *IEEE Body Sensor Networks (BSN'11)*, pp. 53–58, 2011.
- [91] S. Craciun *et al.*, "Wireless transmission of neural signals using entropy and mutual information compression," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 1, pp. 1534–4320, 2011.
- [92] M. Singh, V. Kumar, and S. C. Saxena, "Modified spih wavelet compression for ECG signal," *Journal of Medical Engineering and Technologies*, vol. 31, no. 1, pp. 29–35, 2007.
- [93] D. A. Huffman, *A Method for the Construction of Minimum-Redundancy Codes*.
- [94] "Mean Shift Algorithm," <http://www.mathworks.com/matlabcentral/fileexchange/10161-mean-shift-clustering>, accessed : 2015-02-27.
- [95] G. Gagnon-Turcotte, C. Dufresne-Camaro, and B. Gosselin, "Comparison of low-power biopotential processors for on-the-fly spike detection," *IEEE International Symposium On Circuits and Systems (ISCAS'15)*, pp. 802–805, 2015.
- [96] G. Gagnon-Turcotte, M. Sawan, and B. Gosselin, "Low-power adaptive spike detector based on a sigma-delta control loop," *IEEE Annual Conference on Engineering in Medicine and Biology Society (EMBC'15)*, 2015 (accepted paper).
- [97] G. Gagnon-Turcotte, C.-O. D. Camaro, A. A. Kisomi, R. Ameli, and B. Gosselin, "A wireless multichannel optogenetic headstage with on-the-fly spike detection," *IEEE International Symposium On Circuits and Systems (ISCAS'15)*, pp. 1758–1761, 2015.
- [98] G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. D. Koninck, and B. Gosselin, "A low-power adaptive spike detector for in-vivo behavioural neuroscience experiments," *IEEE Transactions on Neural Systems & Rehabilitation Engineering*, 2015 (pending review).
- [99] G. Gagnon-Turcotte, A. A. Kisomi, R. Ameli, C. Dufresne-Camaro, Y. LeChasseur, J. Neron, P. B. Bareil, P. Fortier, C. Bories, Y. D. Koninck, and B. Gosselin, "A wireless optogenetic stimulator headstage with multichannel neural recording capability," *Sensors*, vol. 15, no. 9, pp. 22 776–22 797, 2015.
- [100] G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. D. Koninck, and B. Gosselin, "An optimized adaptive spike detector for behavioural experiments," *IEEE International Symposium On Circuits and Systems (ISCAS'16)*, 2016 (pending review).
- [101] J. Martinez *et al.*, "Realistic simulation of extracellular recordings," *Journal of Neuroscience Methods*, vol. 184, no. 2, pp. 285–93, 2009.

- [102] "OSRAM-Opto-Semiconductor, "LED LB G6SP"," <http://www.osram-os.com/>, accessed : 2015-04-05.
- [103] "MSP430F5328 Mixed signal microcontroller from Texas-Instruments," <http://www.ti.com/product/msp430f5328>, accessed : 2015-04-10.
- [104] "FUNKOS, real-time operating system by Funkenstein Software Consulting," <http://funkos.sourceforge.net/>, accessed : 2015-04-10.
- [105] "TINYOS, real-time operating system," <http://www.tinyos.net/>, accessed : 2015-04-10.
- [106] Z. Yang *et al.*, "Noise characterization, modeling, and reduction for in vivo neural recordings," *Advances in Neural Information Processing Systems*, pp. 2160–2168, 2010.
- [107] Y. Yang *et al.*, "Adaptive threshold neural spike detector using stationary wavelet transform in cmos," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. PP, no. 99, p. 1, 2015 (pending publication).
- [108] G. Gagnon-Turcotte and B. Gosselin, "Multichannel spike detector with an adaptive threshold based on a sigma-delta control loop," *IEEE Annual Conference on Engineering in Medicine and Biology Society (EMBC'15)*, 2015 (accepted paper).
- [109] B. R. Arenkiel *et al.*, "In vivo light-induced activation of neural circuitry in transgenic mice expressing Channelrhodopsin-2," *Neuron*, vol. 54, no. 2, pp. 205–18, 2007.
- [110] X. Han *et al.*, "Millisecond-timescale optical control of neural dynamics in the nonhuman primate brain," *Neuron*, vol. 62, no. 2, pp. 191–198, 2009.
- [111] X. Han, "Optogenetics in the nonhuman primate," *Prog Brain Res*, vol. 196, pp. 215–233, 2012.
- [112] A. M. Herman *et al.*, "Cell type-specific and time-dependent light exposure contribute to silencing in neurons expressing Channelrhodopsin-2," *eLIFE Neuroscience*, 2014. doi : 10.7554/eLife.01481. Epub 2014 Jan 28
- [113] G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. D. Koninck, and B. Gosselin, "A wireless optogenetic headstage with multichannel neural signal compression," *IEEE Conference on Biomedical Circuits and System (BioCAS'15)*, 2015 (accepted paper).
- [114] G. Gagnon-Turcotte, Y. LeChasseur, C. Bories, Y. Messaddeq, Y. D. Koninck, and B. Gosselin, "A wireless optogenetic headstage with 32 channels neural recording capability," *IEEE Transactions on Biomedical Circuits and Systems*, 2015 (pending review).
- [115] "RHD2132 Puce d'enregistrement de signaux electrophysiologiques de la compagnie Intan," [http://www.intantech.com/products\\_RHD2000.html](http://www.intantech.com/products_RHD2000.html), accessed : 2015-02-23.

- [116] "THORLABS optogenetic LEDs," [https://www.thorlabs.de/newgrouppage9.cfm?objectgroup\\_id=5206](https://www.thorlabs.de/newgrouppage9.cfm?objectgroup_id=5206), accessed : 2015-05-31.
- [117] Z. Yang *et al.*, "A microprobe for parallel optical and electrical recordings from single neurons in vivo," *Nature Methods*, vol. 8, pp. 319–325, 2011. doi : 10.1038/nmeth.1572
- [118] Y. Chompusri *et al.*, "Energy packing efficiency based threshold level selection for DTW ECG compression," *International Journal of Applied Biomedical Engineering*, vol. 2, no. 2, pp. 19–28, 2009.
- [119] A. H. Gittis *et al.*, "Mechanisms of sustained high firing rates in two classes of vestibular nucleus neurons : Differential contributions of resurgent Na, Kv3, and BK currents," *Journal of Neurophysiology*, vol. 104, no. 3, p. 1625–1634, 2010.
- [120] E. S. Boyden *et al.*, "Millisecond-timescale, genetically targeted optical control of neural activity," *Nature Neuroscience*, vol. 8, no. 9, pp. 1263–1268, 2005.
- [121] M. Rizk *et al.*, "A single-chip signal processing and telemetry engine for an implantable 96-channel neural data acquisition system," *Journal of Neural Engineering*, vol. 4, no. 3, pp. 309–321, 2007.
- [122] S. Farshchi *et al.*, "Embedded neural recording with TinyOS-based wireless-enabled processor modules," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 2, pp. 134–141, 2010.
- [123] "MicroProbes tableau de microélectrodes de pénétration métalliques avec fibre optique (optrode)," <http://www.microprobes.com/index.php/products/multi-channel-arrays/optogenetic-arrays#prettyPhoto>, accessed : 2015-02-23.
- [124] S. K. Arfin *et al.*, "Wireless neural stimulation in freely behaving small animals," *Journal of Neurophysiology*, vol. 102, no. 1, pp. 598–605, 2009.