



Mémoire de Projet de Fin d'Études

présenté par

M. BENAYAD AHMED

Filière : Automatique

Spécialité: Automatique et informatique industrielle

Thème

Détection des mouvements fœtaux basée sur des filtres à réponse impulsionnelle infinie

Encadré par : M. BENSLAMA ZOUBIR

Année Universitaire 2019-2020



Nos remerciements vont premièrement à Dieu tout puissant pour la volonté, la Santé, et la patience, qu'il nous a données durant toutes ces années d'étude.

Nous exprimons nos profondes gratitudes à nos parents pour leurs encouragements, leur soutien et pour les Sacrifices qu'ils ont endurés. Ainsi, nous tenons également à exprimer nos vifs remerciements à notre promoteur Mr BENSLAMA pour suivi tout le long de la réalisation de ce mémoire, et qui n'a pas cessé de nous donner ses conseils et remarques.

Nos remerciements vont également Mr A. MADDI pour son attention et son suivi.

Nos sincères remerciements à messieurs les membres du jury pour l'honneur qu'ils nous font en participant au jugement de ce travail.

Nos remerciements vont aussi à tous les enseignants du département de génie électrique.

ملخص:

المجال المعني هو الطب وخاصة البيو طبي، استخدمنا إشارات التسارع، (FFT)، أفنعة ثنائية وتصميم المرشحات لتحقيق هدفنا وهو: الكشف عن حركة الجنين ومراقبة الجنين.

قمنا بتركيب جهاز رصد حركة الجنين على أساس مقياس التسارع، باستخدام بطاقة (ARDUINO) كمتحكم دقيق للنظام. ويعتبر هذا النظام لا سلبي وغير غازي ويمكن ارتداؤه لفترة طويلة من الزمن، كما أنه لن يكون له أي آثار سلبية على الجنين أو الأم.

الكلمات الرئيسية: الطب الحيوي؛ FFT؛ تصميم؛ مستشعر التسارع.

Résumé : Le domaine concerné est la médecine et plus spécialement la biomédicale, on a utilisé les signaux d'accélération, la FFT, les masques binaires et la conception des filtres permettant d'atteindre notre objectif qui est : La détection des mouvements fœtaux et la surveillance fœtale.

Nous avons installé un moniteur de mouvement fœtal basé sur l'accéléromètre, utilisant la carte ARDUINO comme microcontrôleur pour le système. Ce système est considéré comme passif et non invasif et peut être porté pendant une plus longue période, car il n'aura pas d'effets négatifs sur ni le fœtus ni la mère.

Mots clés : biomédicale ; FFT ; conception ; accéléromètre.

Abstract: The field concerned is medicine and especially biomedical, we used acceleration signals, FFT, binary masks and the design of filters to achieve our goal which is: fetal motion detection and fetal surveillance.

We have installed a fetal motion monitor based on the accelerometer, using ARDUINO card as a microcontroller for the system. This system is considered passive and non-invasive and can be worn for a longer period of time, as it will not have any negative effects on either the fetus or the mother.

Keywords: biomedical; FFT; Design; Accelerometer.

Table des matières

Introduction général.....	2
Chapitre 1 : généralité sur le suivi de l'évolution d'un fœtus.....	5
1.1 Introduction	5
1.2 Historique de la "mort naissance"	5
1.3 Définition des problèmes.....	5
1.4 Outils et méthodes de détection des mouvements fœtaux utilisés dans la littérature .	6
1.4.1 Jauge de contrainte et capteurs de pression	7
1.4.2 Capteur de force	7
1.4.3 Capteur de force	8
1.4.4 Capteurs multiscales (physiologiques).....	9
1.4.5 Capteurs d'accélération capacitifs	9
1.4.6 Échographie	11
1.4.7 Capteurs acoustiques.....	11
1.4.8 Accéléromètres	12
1.5 Conclusion.....	13
Chapitre 2 : Arduino	15
2.1 Introduction	15
2.2 Pourquoi arduino ?	15
2.3 Introduction à la carte ARDUINO UNO	16
2.3.1 Synthèse des caractéristiques.....	17
2.3.2 Constituant de la carte Arduino.....	17
2.4 Conclusion.....	24
Chapitre 3 : Réalisation	26
3.1 Introduction	26
3.2 Synoptique général	26
3.2.1 Capteurs	27
3.2.2 Protocole de communication inter-intégré de circuit	29
3.3 Méthodes de traitement du signal utilisées	33
3.3.1 Transformation de fourier	33
3.3.2 Root-Mean-Square.....	36
3.3.3 Filtrage numérique.....	36
3.3.4 Méthodes d'évaluation	46
3.4 Système d'acquisition	49
3.4.1 Logiciel d'acquisition	49

3.4.2	Méthodologie acquisition	51
3.4.3	Signaux acquis	52
3.5	Méthodologie de détection de mouvement fetale	53
3.5.1	Retrait signal	53
3.5.2	Racine -moyen-carré.....	56
3.6	Conception du filtre	62
3.7	Implémentation de filtre.....	73
3.7.1	Computation de valeur absolu et détection de Seuil	78
3.7.2	Évaluation de la détection du mouvement féministe algorithme	80
3.8	Résultats de l'évaluation du rendement de la méthode de détection des mouvements fœtaux85	
3.9	Conclusion.....	86
	Conclusion générale	87
	Références.....	88
	Annexe	93

Liste des figures

Figure 2.1 Arduino Uno	16
Figure 2.2 composants de base d'Arduino	17
Figure 2.3 Logiciel Arduino.....	18
Figure 2.4 Déroulement du programme arduino.....	19
Figure 3.1. Acquisition, transforme et transfert des données.....	22
Figure 3.2. Exemple d'implémentation I2C [31].....	25
Figure 3.3. Réponses de filtre idéalisées [41].....	33
Figure 3.4. Paramètres du filtre de clé [41].....	33
Figure 3.5. Filtre passe-bas idéal [45].....	35
Figure 3.6. Filtre idéal à passe haut [45].....	36
Figure 3.7 Filtre idéal à passe-bande [45].....	36
Figure 3.8. Filtre à bande-stop idéale [45].....	37
Figure 3.9. Réponse de magnitude de filtre de Butterworth pour différents ordres [50]	38
Figure 3.10. Réponse de magnitude de filtre de Chebyshev pour différents ordres [52].....	40
Figure 3.11. Réponse de fréquence affichant les paramètres utilisés pour spécifier un filtre elliptique [49].....	41
Figure 3.12. Réponse inverse de magnitude de filtre de Chebyshev pour différents ordres [53].....	42
Figure 3.13. Branchement de système de signal d'acquisition.....	45
Figure 3.14. Numéro de porte (COM) sur Arduino UNO.....	46
Figure 3.15. Signaux d'accélération X, Y et Z pour la première expérience.....	48
Figure 3.16. Signaux d'accélération X, Y et Z pour la deuxième expérience.....	48
Figure 3.17. Signaux d'accélération X, Y et Z pour la troisième expérience.....	49
Figure 3.18. Feuille de calcul de données et fichier texte.....	50
Figure 3.19. Lecture du fichier de mesure VI.....	50
Figure 3.20. Lire à partir de la zone Dialogue du fichier de mesure VI.....	51
Figure 3.21. Statistiques Express VI.....	51
Figure 3.22. Statistiques Express VI Boîte de dialogue.....	52
Figure 3.23. Diagramme de bloc de calcul RMS.....	53
Figure 3.24. Racine-moyenne-carré des signaux d'accélération de la première expérience.....	53

Figure 3.25. Root-Mean-Square des signaux d'accélération de la troisième expérience.....	54
Figure 3.26. Root-Mean-Square des signaux d'accélération de la troisième expérience.....	54
Figure 3.27. Spectral Measurement Express VI.....	55
Figure 3.28. Boîte de dialogue Spectral Measurement Express VI.....	55
Figure 3.29. Réponse de magnitude du RMS pour la première expérience.....	56
Figure 3.30. Réponse de magnitude du RMS pour la deuxième expérience.....	56
Figure 3.31. Réponse de magnitude du RMS pour la troisième expérience.....	57
Figure 3.32. Butterworth Design VI.....	58
Figure 3.33. Diagramme de bloc de conception de filtre.....	59
Figure 3.34. Réponse de l'ampleur de Butterworth.....	59
Figure 3.35. Réponse de Butterworth Impulse.....	60
Figure 3.36. Réponse de Butterworth Step.....	60
Figure 3.37. Placement de pôle de Butterworth.....	61
Figure 3.38. Chebyshev Design VI.....	61
Figure 3.39. Réponse de magnitude de Chebyshev.....	62
Figure 3.40. Réponse de Chebyshev Impulse.....	62
Figure 3.41. Réponse de Chebyshev Step.....	63
Figure 3.42. Placement de pôle de Chebyshev.....	63
Figure 3.43. Elliptic Design VI.....	64
Figure 3.44. Réponse de magnitude elliptic.....	64
Figure 3.45. Réponse d'Elliptic Impulse.....	65
Figure 3.46. Réponse elliptic step.....	65
Figure 3.47. Placement de pôle elliptic.....	66
Figure 3.48. Inverse Chebyshev Design VI.....	66
Figure 3.49. Réponse inverse de l'ampleur de Chebyshev.....	67
Figure 3.50. Réponse inverse de Chebyshev Impulse.....	67
Figure 3.51. Réponse inverse de Chebyshev Step.....	68
Figure 3.52. Placement inverse de pôle de Chebyshev.....	68
Figure 3.53. Filtre Express VI.....	69
Figure 3.54. Boîte de dialogue Filter Express VI.....	70
Figure 3.55. RMS filtré à l'aide du filtre Butterworth.....	71

Figure 3.56. Réponse de fréquence de signal filtré (Butterworth).....	71
Figure 3.57. RMS filtré à l'aide du filtre Chebyshev.....	72
Figure 3.58. Réponse de fréquence de signal filtré (Chebyshev).....	72
Figure 3.59. RMS filtré à l'aide du filtre elliptic.....	73
Figure 3.60. Réponse de fréquence de signal filtré (Elliptic).....	73
Figure 3.61. RMS filtré à l'aide du filtre Inverse Chebyshev.....	74
Figure 3.62. Réponse de fréquence de signal filtré (Inverse Chebyshev).....	74
Figure 3.63. Diagramme de bloc d'algorithme de détection de seuil.....	75
Figure 3.64. Masque binaire du mouvement détecté contre le vrai mouvement.....	75
Figure 3.65. Diagramme de bloc de calcul positif vrai.....	76
Figure 3.66. Diagramme de bloc de calcul négatif vrai.....	77
Figure 3.67. Diagramme de bloc de calcul faux positif.....	77
Figure 3.68. Diagramme de bloc de calcul faux négatif.....	77
Figure 3.69. Algorithme de calcul des valeurs de test de classification binaire.....	78
Figure 3.70. Diagramme de bloc de calcul des valeurs de test de classification binaire.....	79
Figure 3.71. Diagramme de bloc de calcul des mesures statistiques.....	80
Figure 3.72. Diagramme de bloc de calcul des mesures statistiques.....	80

Liste des figures Liste des tableaux

Tableau 1.1 Chronologie du développement de la surveillance de la fréquence cardiaque fœtale [4]	6
Tableau 1.2 Avantages et inconvénients de différents types de capteurs de force.....	8
Tableau 1.3 Avantages et inconvénients des capteurs capacitifs généraux.....	11
Tableau 2.1 Synthèse des caractéristiques.....	17
Tableau 3.1 Avantages et inconvénients d'accéléromètre	24
Tableau 3.2 Avantages et inconvénients I2C.....	26
Tableau 3.3. Polynôme normalisés de Butterworth [49]	38
Tableau 3.4. Polynôme de filtre de Chebyshev [49]	39
Tableau 3.5. Test de classification binaire.....	43
Tableau 3.6. L'IIR filtre les performances pour la détection des mouvements fœtaux avec différents seuils.....	81
Tableau 3.7. L'IIR filtre les performances pour la détection des mouvements fœtaux avec un seuil moyen.....	81
Tableau A.1 L'IIR filtre les performances détaillées pour la détection des mouvements fœtaux avec différents seuils.....	89
Tableau A.2. L'IIR filtre les performances détaillées pour la détection des mouvements fœtaux avec un seuil moyen.....	89

Introduction général

La surveillance des mouvements fœtaux au cours du dernier trimestre de la grossesse est une tâche très importante à faire pendant cette période parce qu'elle indique la santé du fœtus et évite les pires scénarios comme la mort naissance. Les mouvements fœtaux commencent à être remarqués par la mère à la 16e à la 20e semaine de gestation, après une petite période, certains modèles de mouvement commencent à apparaître. Le mouvement fœtal peut différer d'un coup de pied, d'un flottement et d'un swish ou d'un rouleau [1]. Tout changement soudain ou significatif de modèle de mouvement est un signe clinique potentiellement important. Les études de la physiologie fœtale utilisant l'ultrason ont montré une relation entre RFM (mouvements fœtaux réduits) et les résultats périnataux pauvres. La majorité des femmes (55%) qui ont connu une mort naissance ont remarqué un changement de modèle (Réduction) dans leurs mouvements fœtaux. Le nombre d'études sur les décès fœtaux en Norvège et au Royaume-Uni a révélé qu'une réponse inappropriée des cliniciens à la perception maternelle de la RFM était un facteur contributif courant dans la mort naissance [1].

Les premiers mouvements fœtaux, qui sont ressentis par la mère, sont appelés accélération. Ces mouvements alerteront la femme enceinte qu'elle a un fœtus qui pousse dans son utérus. Cependant, les mouvements d'accélération ne sont pas une indication absolue d'un fœtus à venir parce que d'autres mouvements du corps de la femme peuvent imiter les mouvements fœtaux précoces tels que le péristaltisme, le flatul, et les contractions abdominales de muscle [2]. Habituellement, la femme enceinte est enseignée par leur fournisseur de soins de santé pour surveiller les mouvements de son fœtus. On peut leur apprendre à compter le nombre de coups de pied qu'ils ressentent dans un laps de temps déterminé et à suivre ces chiffres pendant un certain nombre de jours, puis à déterminer si le fœtus se déplace moins que la normale un jour donné. Les cliniciens conseillent souvent de commencer la surveillance vers la 28e semaine et de continuer à compter tout le reste de la période de grossesse.

La méthode la plus célèbre s'appelle (Comte à Dix). Dans cette méthode, la femme peut s'allonger sur son dos et compter les coups de pied de son fœtal. Une fois que les coups de pied atteignent dix, alors elle doit garder un registre du temps qui a pris ces dix mouvements à la fin. Au fur et à mesure que le bébé se développe, le nombre et le type de mouvements, ainsi que les modèles, varient. Habituellement, les périodes de l'après-midi et du soir sont des périodes de pointe pour le fœtus. Pendant le jour et la nuit, le fœtus a des périodes de sommeil qui durent la plupart du temps entre 20 et 40 minutes, et ils sont rarement plus de 90 minutes. Le nombre de mouvements tend à augmenter jusqu'à 32 semaines de grossesse, puis reste à peu près le même; toutefois, le type de mouvement peut changer à mesure que la mère se rapproche de sa date d'échéance [3]. Il n'y a pas de nombre spécifique de mouvements, ce qui est normal. Pendant la grossesse, la mère doit être consciente du modèle de mouvement de son bébé. Comme nous l'avons mentionné ci-dessus, le changement de modèle est ce qui est important.

Étant donné que ce sujet est très important pour la société et a un grand impact sur sa prospérité, nous avons eu le privilège d'écrire la thèse sur cette question.

Chapitre 1

Généralité sur le suivi de l'évolution d'un fœtus

Chapitre 1 : généralité sur le suivi de l'évolution d'un fœtus

1.1 Introduction

Plusieurs techniques sur le suivi de fœtus qui ont été développées, il y a eu celles qui ont donné de bons résultats et d'autres qui ont été délaissées par les familles médicales. Dans ce chapitre on va énumérer et développer les principales techniques de suivi pour qu'à la fin nous parlerons de la méthode qu'il on utilise dans notre thèse.

1.2 Historique de la ‘mort naissance’

Au **IV^{ème} siècle avant JC**, Hippocrate a soutenu que les premiers mouvements fœtaux apparaissaient 70 à 90 jours après la conception.

Ambroise Paré, en **1534**, a été le premier à affirmer que “si ni la main du chirurgien ni la mère ne peuvent sentir bouger l'enfant, celui-ci est mort”. Toujours au XVI^{ème} siècle, VESALE a étudié les mouvements respiratoires déclenchés par l'asphyxie chez un fœtus de chien. Ses travaux furent poursuivis par BECLARD en **1813**, MINKOWSKI en **1938** et DAWES en **1970**.

Depuis une vingtaine d'années, un regain d'intérêt s'est manifesté pour les mouvements fœtaux. Des équipes israéliennes, britanniques, scandinaves, canadiennes et françaises se sont intéressées à la valeur pronostique du recensement des mouvements fœtaux.

En Allemagne et aux États-Unis, des études ont été faites sur l'activité motrice fœtale objectivée par échographie, d'une part, et sur le retentissement de ces mouvements sur le rythme cardiaque fœtal enregistré par cardiotocographe, d'autre part.

En **1977**, SADOVSKY a établi une relation entre le compte quotidien des mouvements fœtaux et le pronostic obstétrical. BENHAMOU a constaté, par la statistique, le peu de valeur à accorder à la subjectivité des mères françaises.

Des recherches ont alors été menées pour permettre un recensement objectif des mouvements. COLETTE a proposé un appareil permettant un recensement objectif des mouvements fœtaux : le Tococinon.

1.3 Définition des problèmes

La surveillance manuelle du mouvement fœtal peut être très frustrante pour la mère, même si, elle peut s'allonger sur son dos et aller de l'avant avec elle, mais encore, elle ne sera pas en mesure de détecter ces mouvements avec une grande précision.

Pour résoudre ce problème de nombreuses approches a eu lieu tout au long du siècle dernier et plus, le but de ces méthodes est de détecter les mouvements fœtaux, le signal électrocardiographique, le pouls cardiaque, ainsi que de nombreux autres signaux.

La plupart des méthodes étudiées sont non invasives, en raison de l'utilisation d'un capteur externe pour surveiller les activités fœtales. Le tableau suivant passe en revue plusieurs capteurs utilisés au fil des ans pour surveiller la fréquence cardiaque fœtale à titre d'exemple.

Année	La méthode
1906	FECG abdominale et intra-vaginale
1917	Le Fetoscope
1931	Le phonocardiogramme
1947	Le tocodynamomètre
1958	Surveillance continue du FECG
1964	Échographie Doppler
1968	Cardiotocographie (CTG)
1972	Électrodes fœtales de cuir chevelu
2000	Oxymétrie des impulsions fœtales

Tableau 1.1 Chronologie du développement de la surveillance de la fréquence cardiaque fœtale [4].

1.4 Outils et méthodes de détection des mouvements fœtaux utilisés dans la littérature

Outre la perception maternelle, de nombreux capteurs ont été utilisés tout au long de l'histoire. Dans cette partie, nous allons parler des différents capteurs que nous rencontrons dans la littérature; nous donnerons leurs avantages ainsi que leurs inconvénients; aussi, nous allons parler des méthodes que les auteurs ont utilisées dans leurs recherches, et les résultats qu'ils ont obtenus en utilisant tout cela.

1.4.1 Jauge de contrainte et capteurs de pression

Dans certains articles de recherche, nous avons vu l'utilisation des transducteurs de jauge de souche comme l'un des moyens simples et sûrs de mesurer le mouvement fœtal. Les transducteurs ont répondu à tous les mouvements abdominaux, cependant, la discrimination entre les mouvements maternels et les mouvements fœtaux pourrait être atteint en choisissant une taille appropriée pour le transducteur et en utilisant un filtre adaptatif comme méthode de filtrage [5].

En outre, dans un document examiné, un sac pressurisé à air a été conçu pour être enroulé autour de l'abdomen des femmes enceintes; et il a été relié à un capteur de pression afin de détecter la force exercée contre la paroi de l'utérus dans n'importe quelle direction [6].

Le but de ces systèmes est d'appliquer les techniques passives, où ni l'énergie mécanique ni électrique n'est utilisée pour perturber l'environnement intra-utérin. Les auteurs ont également conclu que la technique passive convient au système de surveillance à long terme.

1.4.2 Capteur de force

Ce type de technique est très similaire à la technique du capteur de pression. Dans un article de revue examiné, les auteurs ont présenté un système de détection pour la classification fœtale de mouvement basé sur l'ampleur de la force exercée sur la paroi abdominale d'une femme enceinte. En outre, les auteurs ont fait valoir que même si l'efficacité de la méthode d'échographie dans la surveillance fœtale, la caractérisation du mouvement dépend de la compétence d'un expert. Cependant, leur système de détection a été développé en incorporant le tableau de résistance sensible à la force dans une ceinture portable pour les femmes enceintes, il a réussi à caractériser le mouvement de la main, des jambes et du tronc avec une précision allant de 75% à 93%, et ils ont conclu que ce système est une technique prometteuse pour la détection de mouvement fœtal à faible coût [7]. Le tableau suivant montre certains des avantages et des inconvénients des différents types de capteurs de force.

1.4.3 Capteur de force

Ce type de technique est très similaire à la technique du capteur de pression. Dans un article de revue examiné, les auteurs ont présenté un système de détection pour la classification fœtale de mouvement basé sur l'ampleur de la force exercée sur la paroi abdominale d'une femme enceinte. En outre, les auteurs ont fait valoir que même si l'efficacité de la méthode d'échographie dans la surveillance fœtale, la caractérisation du mouvement dépend de la compétence d'un expert. Cependant, leur système de détection a été développé en incorporant le tableau de résistance sensible à la force dans une ceinture portable pour les femmes enceintes, il a réussi à caractériser le mouvement de la main, des jambes et du tronc avec une précision allant de 75% à 93%, et ils ont conclu que ce système est une technique prometteuse pour la détection de mouvement fœtal à faible coût [7]. Le tableau suivant montre certains des avantages et des inconvénients des différents types de capteurs de force.

Type de capteur de force	Avantages	Inconvénients
Cellule de chargement	• Il est facilement disponible. • Il a une meilleure précision, qui est de 0,1% de l'échelle complète. • Les fabricants ou les fournisseurs le calibrent habituellement.	• Il est volumineux dans la taille. • Il est rigide dans la construction. • Il faut de l'électronique coûteuse.
Piézo-Résistif	• Il est très mince et flexible dans la construction. • Il utilise des appareils électroniques peu coûteux. • Il est facilement disponible et peut être personnalisé. • Il est léger. • Il nécessite une faible puissance pour son fonctionnement.	• Il a une précision inférieure (environ +/-5% pleine échelle) comparer à la cellule de charge. • Il faut l'étalonnage par l'utilisateur.

Tableau 1.2 Avantages et inconvénients de différents types de capteurs de force.

1.4.4 Capteurs multiscales (physiologiques)

Cette technologie a été présentée dans un article, car les auteurs proposent un système basé sur la technologie portable pour détecter les mouvements du fœtus de manière pratique et précise. Leur système comprend un vêtement intelligent intégrant plusieurs capteurs multi-échelles connectés les uns aux autres. Une plate-forme standardisée de cloud-computing (informatique en nuage) sur laquelle le diagnostic médical est réalisé à l'aide d'un système expert médical. Sur la base des signaux mesurés, ils proposent une méthode combinant le débruitage par ondelettes qui utilise une décomposition en trois couches basées sur un seuil souple et un filtre passe-bande pour extraire les caractéristiques pertinentes des mouvements fœtaux [8].

1.4.5 Capteurs d'accélération capacitifs

Dans un article, dirigé par Kyoko Nishihara et coll., les auteurs ont développé un nouveau capteur avec une capacité électrostatique qui peut capter l'accélération du mouvement fœtal. Ils ont fait deux expériences. Dans le premier, ils ont vérifié la fiabilité de leur système d'enregistrement des mouvements fœtaux alors qu'ils le testaient sur trente-deux femmes enceintes (de 19 à 39 semaines de gestation) qui, de leur côté, ont été invitées à s'allonger sur un lit pendant environ une heure et à appuyer sur un bouton comme marqueur subjectif lorsqu'elles se sentaient en mouvement fœtal. Ils ont simultanément enregistré des polysomnogrammes maternels et le mouvement fœtal de l'abdomen des mères. Dans la deuxième expérience, ils ont enregistré sept polysomnogrammes et mouvements fœtaux de femmes enceintes simultanément pendant le sommeil toute la nuit à la maison en utilisant leur capteur développé pendant les semaines 33 et 36 de la gestation [9] [10].

En outre, ils ont développé un nouveau système logiciel automatisé pour analyser les mouvements fœtaux enregistrés [11]. Leur étude a introduit le système et compare ses résultats à ceux d'une analyse manuelle des mêmes signaux de mouvement fœtal effectués dans [9] [10]. La méthode principale était basée sur les filtres band-pass et band-stop comme méthode de prétraitement ainsi que le calcul des valeurs intégrales pour les signaux. Cette étude a montré de très bons résultats et précis, et elle réduit avec succès le nombre d'artefacts.

Le tableau suivant montre les avantages ainsi que les inconvénients des capteurs capacitifs en général; il n'est pas seulement lié aux capteurs d'accélération capacitifs.

Avantages	Inconvénients
• L'élément capacitif est mécaniquement simple et robuste. • Les capteurs capacitifs peuvent fonctionner sur une large plage de température et sont très tolérants aux conditions de surpression à court terme. • Comme aucun courant de courant n'est acheminé par le condensateur, il s'y trouve en soi de faible puissance. • Les capteurs capacitifs présentent une faible hystérèse et une bonne répétabilité des mesures. • Le temps de réponse est de l'ordre de millisecondes, et encore plus rapide	• La non-linéarité est exposée parce que la sortie est inversement proportionnelle à l'écart entre les électrodes parallèles. L'interface doit minimiser la capacité errante en ayant l'électronique aussi proche que possible du capteur. C'est un autre avantage de la technologie MEMS.

dans le cas des appareils MEMS. Parce qu'il s'appelle des appareils AC, les capteurs capacitifs conviennent aux applications sans fil.	
--	--

Tableau 1.3 Avantages et inconvénients des capteurs capacitifs généraux

1.4.6 Échographie

Sophie Ribes et coll. ont travaillé sur la collecte de paramètres d'activité fœtale tels que les mouvements, la fréquence cardiaque et les paramètres connexes pour séparer automatiquement les fœtus en bonne santé des fœtaux compromis [12]. Pour atteindre cet objectif, ils ont d'abord développé un système Doppler multi-capteurs et multi-portes; après cela, ils ont enregistré des signaux doppler multidimensionnels et estimé les paramètres d'activité fœtale par le biais de techniques de traitement du signal dédiées. Ensuite, ils ont combiné ces paramètres en quatre ensembles de paramètres pour déterminer l'ensemble de paramètres qui peuvent séparer les fœtus sains des autres fœtus. Dans l'ensemble, les meilleurs résultats ont été obtenus avec l'ensemble des paramètres se composant du mouvement fœtal, de la variabilité à court terme, de la variabilité à long terme, de la décélération, et du score d'ultrason.

1.4.7 Capteurs acoustiques

Étant donné qu'une surveillance fiable des mouvements fœtaux en dehors des environnements cliniques n'est pas actuellement possible, Jonathan Lai et coll. ont mis au point un système portable et non transmettant capable de détecter les mouvements fœtaux sur de longues périodes ainsi que d'établir des niveaux normaux de mouvement dans la population en général [13]. Basés sur une nouvelle combinaison d'accéléromètres et de capteurs acoustiques sur mesure, ils ont introduit un nouveau système portable et utilisé des techniques avancées de traitement du signal pour identifier les mouvements fœtaux. Leurs résultats démontrent la promesse de la détection multimodale pour le développement d'un moniteur portable à faible coût et non transmettant pour les mouvements fœtaux.

1.4.8 Accéléromètres

De nombreux chercheurs ont utilisé des capteurs d'accéléromètre pour détecter les mouvements fœtaux. Ces capteurs ont donné des résultats prometteurs dans certaines recherches et de très bons résultats dans d'autres, selon les méthodes post-acquisition utilisées.

Girier Thomas et coll. ont fait une étude préliminaire pour mettre au point une méthode permettant de détecter automatiquement les mouvements fœtaux à l'aide d'un système d'accéléromètre [14]. Ils évaluent l'efficacité de l'utilisation d'une méthode de seuil sur une gamme de bandes de fréquences différentes. Ils examinent également l'utilisation d'un ensemble de caractéristiques statistiques pour une méthode de détection. Malheureusement, leurs résultats n'indiquent qu'aucune des deux méthodes ne fonctionne suffisamment bien pour détecter automatiquement le mouvement fœtal.

M. Mesbah et coll. ont créé un accéléromètre ambulateur [15], un moniteur d'activité fœtale basé sur l'enregistrement des mouvements fœtaux 24 heures sur 24, et ils ont utilisé ce système pour élaborer des méthodes de traitement du signal comme l'informatique du RMS pour détecter et caractériser quantitativement les mouvements fœtaux automatiquement; mais aucune tentative n'a été faite pour éliminer les artefacts trouvés dans les données d'accélération. Leur première étape dans cette direction a été de tester la performance de l'accéléromètre dans la détection du mouvement fœtal contre l'imagerie par ultrasons en temps réel prise comme l'étalon-or. Les résultats montrent que la technique est plus efficace que celle perçue par la mère, aussi ils ont noté que le détecteur proposé fonctionne mieux dans l'identification des épisodes d'activité fœtale et des épisodes d'inactivité.

M. S. Khlif et coll. ont utilisé les données acquises par les capteurs d'accéléromètre [16], ils ont appliqué des distributions time-fréquence telles que le spectrogramme et la distribution modifiée de B pour caractériser les mouvements fœtaux dans le domaine de la fréquence temporelle, ainsi que l'utilisation du carré racine-moyen de la série de temps et évaluent sa performance par rapport à l'imagerie par ultrasons en temps réel, prise comme étalon-or. L'évaluation a montré une meilleure performance avec le détecteur basé sur RMS par rapport à la perception maternelle.

Siamak Layegh et coll. a présenté une approche de fréquence horaire pour la surveillance des mouvements fœtaux, qui est basée sur la classification des signaux d'accéléromètre prélevés sur l'abdomen des femmes enceintes [17]. Les caractéristiques extraites de la distribution de fréquences temporelles de ces signaux ont été fournies dans l'analyse statistique pour générer des mélanges de fonction-mesure. Quatre classes différentes sont subjectivement reconnues dans les données de l'accéléromètre en utilisant des outils objectifs tels que l'échographie. Analyse des caractéristiques de fonctionnement du récepteur utilisée pour calculer la performance des mesures de fonctionnalité pour la comparaison entre les différentes classes. Après cela, ils ont appliqué une sélection de fonctionnalités pour réduire la dimension d'espace de fonctionnalité à l'aide de l'analyse des composants principaux. Les mesures de

fonctionnalités sélectionnées ont ensuite été utilisées pour prendre en charge les classificateurs de machines vectorielles pour classer les mouvements d'artefacts et de fœtus dans différents sous-ensembles de classes disponibles. Les résultats indiquent que les événements de mouvement fœtal sont identifiés avec une précision de 92,19%.

Ni Diana binti Zakaria et coll. présentent un enregistreur de mouvement fœtal à l'aide d'un dispositif de mesure du mouvement basé sur l'accéléromètre [18]. Un module avec des capteurs à deux points a été placé sur la paroi abdominale maternelle, et les données ont été stockées dans une carte mémoire. Les résultats suggèrent que le dispositif proposé peut être un choix idéal pour la mesure du mouvement fœtal en raison de sa meilleure précision par rapport à la technique de perception maternelle.

Ni Diana Zakaria et coll. continuent leur travail. Ils ont ajouté quatre autres capteurs d'accéléromètre [19], et ont utilisé le microcontrôleur Arduino [20] interfacé avec l'outil de processus de signal Matlab [21] pour enregistrer, afficher et stocker les ensembles pertinents de mouvements fœtaux. La comparaison des données enregistrées à partir des mouvements fœtaux avec l'échographie et la technique de perception maternelle montrent que le système d'enregistrement des mouvements fœtal proposé a un meilleur taux de précision que la technique de perception maternelle, et il peut être comparé à l'échographie.

Marco Altini et coll. Dans l'un de leurs travaux publiés [22]; a utilisé un ensemble de données de quinze mesures recueillies à l'aide de six trois accéléromètres axiaux et a montré que l'inclusion d'un accéléromètre de référence à l'arrière du participant améliore constamment les performances de détection des mouvements fœtaux, quel que soit le nombre de capteurs utilisés. Ils montrent également que deux accéléromètres plus un accéléromètre de référence sont suffisants pour des résultats optimaux.

1.5 Conclusion

Après avoir vu la généralité nous allons dans le chapitre qui suit développer l'outil qui nous permettra d'arriver notre objectif à savoir la réalisation d'une carte de développement qui sera faite sur la base de la carte de prototypage Arduino.

Chapitre 2

Logiciels et matériaux utilisés

Chapitre 2 : Arduino

2.1 Introduction

Aujourd'hui, l'intégration des systèmes numériques dans le domaine de l'électronique tend à remplacer les systèmes analogiques, dans le but de leur miniaturisation, formant ainsi des systèmes informatiques embarqués, tout en réduisant leur coût de fabrication. Il en résulte des systèmes plus complexes et performants pour un espace réduit. En fait depuis l'avènement de l'électronique, son avancée continue n'a cessé de progresser. Dans ce chapitre on va introduire un ordinateur numérique qui n'est autre que la carte de prototypage Arduino qui servira comme outil d'acquisition et de traitement avec son outil de programmation.

2.2 Pourquoi arduino ?

L'Arduino Uno est un microcontrôleur programmable qui permet, comme son nom l'indique, de contrôler des éléments mécaniques : systèmes, lumières, moteurs, etc. Cette carte électronique permet donc à son utilisateur de programmer facilement des choses et de créer des mécanismes automatisés, sans avoir de connaissances particulières en programmation. Il est un outil pensé et destiné aux inventeurs, artistes ou amateurs qui souhaitent créer leur propre système automatique en le codant de toute pièce.

Il vous suffit de connecter votre carte électronique sur votre ordinateur (Windows, Mac ou Linux) et vous pouvez commencer à programmer quelque chose à partir du logiciel Arduino.

Tout commence en 2004, lorsque L'Arduino Uno a été développé par des enseignants et élèves d'une école de Design italienne, qui avaient pour vocation de démocratiser la programmation et de la rendre plus libre d'accès. Depuis lors, les plans ont été partagés et sont libres d'accès. N'importe qui peut se lancer dans la confection de son propre microcontrôleur programmable s'il le souhaite ; pour se faire, la toile regorge d'informations et de tutoriels en tout genre. Cependant, le nom Arduino est déposé et ne peut être repris. Ce qui semble légitime.

2.3 Introduction à la carte ARDUINO UNO

L'Arduino Uno est un microcontrôleur programmable qui permet, comme son nom l'indique, de contrôler des éléments mécaniques : systèmes, lumières, moteurs, etc. Cette carte électronique permet donc à son utilisateur de programmer facilement des choses et de créer des mécanismes automatisés, sans avoir de connaissances particulières en programmation. Il est un outil pensé et destiné aux inventeurs, artistes ou amateurs qui souhaitent créer leur propre système automatique en le codant de toute pièce.

Il vous suffit de connecter votre carte électronique sur votre ordinateur (Windows, Mac ou Linux) et vous pouvez commencer à programmer quelque chose à partir du logiciel Arduino.

Tout commence en 2004, lorsque L'Arduino Uno a été développé par des enseignants et élèves d'une école de Design italienne, qui avaient pour vocation de démocratiser la programmation et de la rendre plus libre d'accès. Depuis lors, les plans ont été partagés et sont libres d'accès. N'importe qui peut se lancer dans la confection de son propre microcontrôleur programmable s'il le souhaite ; pour se faire, la toile regorge d'informations et de tutos en tout genre. Cependant, le nom Arduino est déposé et ne peut être repris. Ce qui semble légitime.

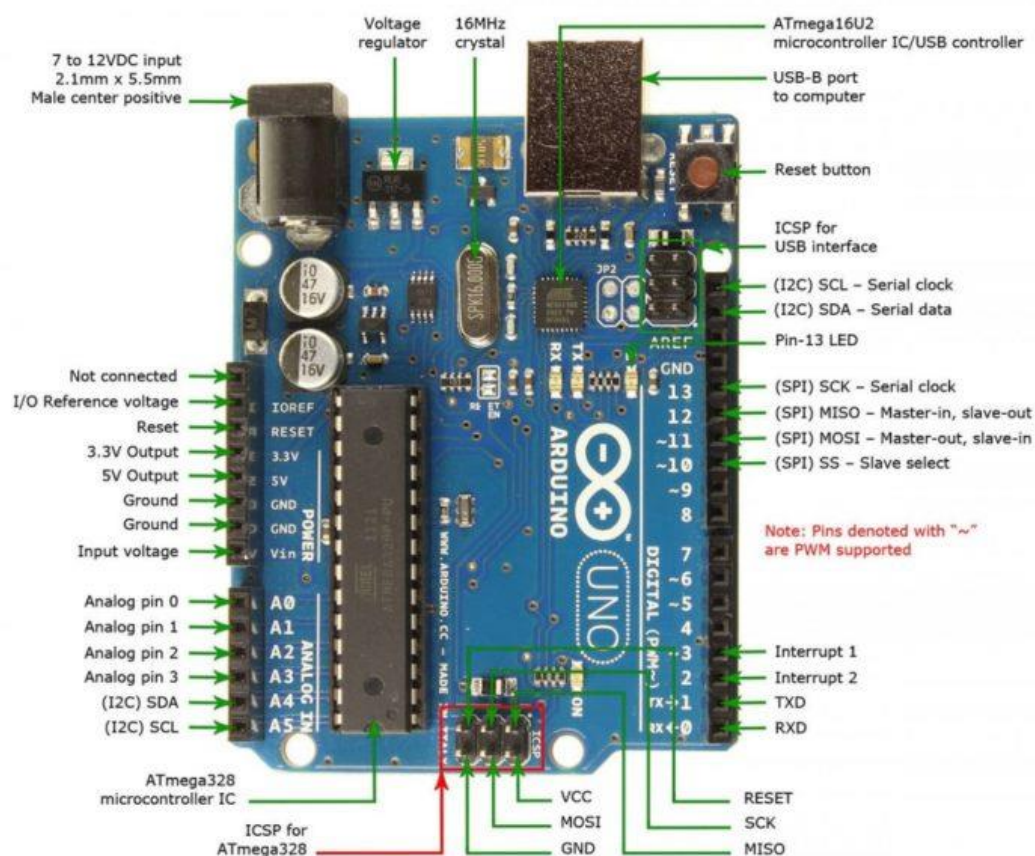


Figure. 2.1 Arduino Uno

2.3.1 Synthèse des caractéristiques

Microcontrôleur	Atmega328
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Tension d'alimentation (limites)	6-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM)
Broches d'entrées analogiques	6 (utilisables en broches E/S numériques)
Intensité maxi disponible par broche E/S (5V)	40 mA (ATTENTION : 200mA cumulé pour l'ensemble des broches E/S)
Intensité maxi disponible pour la sortie 3.3V	50 mA
Intensité maxi disponible pour la sortie 5V	Fonction de l'alimentation utilisée – 500 mA max si port USB utilisé seul
Mémoire Programme Flash	32 KB (Atmega328) dont 0.5 KB sont utilisés par le bootloader
Mémoire SRAM (mémoire volatile)	2 KB (Atmega328)
Mémoire EEPROM (mémoire non volatile)	1 KB (Atmega328)
Vitesse d'horloge	16 MHz

Tableau 2.1 Synthèse des caractéristiques

2.3.2 Constituant de la carte Arduino

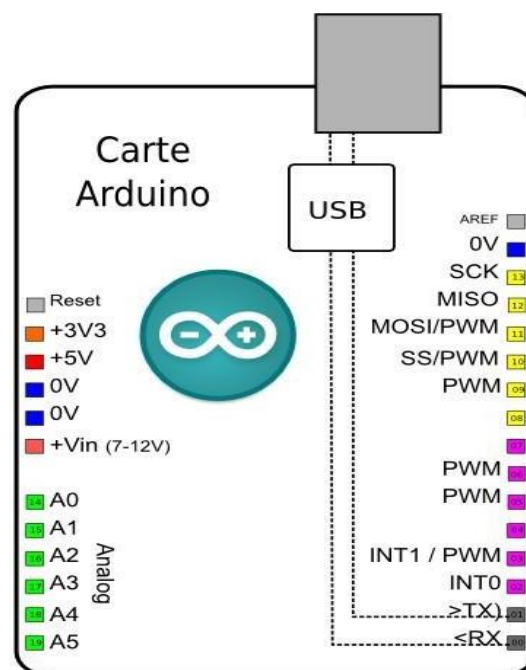


Figure. 2.2 composants de base d'Arduino

a) Le microcontrôleur :

Un microcontrôleur est un circuit intégré qui rassemble les éléments essentiels d'un ordinateur : processeur, mémoires (mémoire morte pour le programme, mémoire vive pour les données), unités périphériques et interfaces d'entrées-sorties. Les microcontrôleurs se caractérisent par un plus haut degré d'intégration, une plus faible consommation électrique (quelques milliwatts en fonctionnement, quelques nanowatts en veille), une vitesse de fonctionnement plus faible (quelques mégahertz à quelques centaines de mégahertz) et un coût réduit par rapport aux microprocesseurs polyvalents utilisés dans les ordinateurs personnels.

b) L'alimentation :

La carte Arduino Uno peut être alimentée soit via la connexion USB (qui fournit 5V jusqu'à 500mA) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte.

L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12V sous 500mA) ou des piles (ou des accus). L'adaptateur secteur peut être connecté en branchant une prise 2.1mm positif au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs des broches de la carte appelées Gnd (masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation.

La carte peut fonctionner avec une alimentation externe de 6 à 20 volts. Cependant, si la carte est alimentée avec moins de 7V, la broche 5V pourrait fournir moins de 5V et la carte pourrait être instable. Si on utilise plus de 12V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. Aussi, la plage idéale recommandée pour alimenter la carte Uno est entre 7V et 12V.

Les broches d'alimentation sont les suivantes :

VIN. La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer du 5V de la connexion USB ou autre source 5V régulée). Vous pouvez alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.

5V. La tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite "tension régulée" obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte

Arduino). Le 5V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB (qui fournit du 5V régulé) ou de tout autre source d'alimentation régulée.

+3V3. Une alimentation de 3.3V fournie par le circuit intégré FTDI (circuit intégré faisant l'adaptation du signal entre le port USB de votre ordinateur et le port série de l'ATmega) de la carte est disponible : ceci est intéressant pour certains circuits externes nécessitant cette tension au lieu du 5V). L'intensité maximale disponible sur cette broche est de 50mA.

GND. Broche de masse (ou 0V).

c) Entrées et sorties numériques :

Chacune des 14 broches numériques de la carte UNO (numérotées des 0 à 13) peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique, en utilisant les instructions pinMode (), digitalWrite () et digitalRead () du langage Arduino. Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité et dispose d'une résistance interne de "rappel au plus" (pull-up) (déconnectée par défaut) de 20-50 KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction digitalWrite (broche, HIGH).

De plus, certaines broches ont des fonctions spécialisées :

Communication Série : Broches 0 (RX) et 1 (TX). Utilisées pour recevoir (RX) et transmettre (TX) les données séries de niveau TTL. Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega8U2 programmé en convertisseur USB-vers-série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.

Interruptions Externes : Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur. Voir l'instruction attachInterrupt () pour plus de détails.

Impulsion PWM (largeur d'impulsion modulée) : Broches 3, 5, 6, 9, 10, et

11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction analogWrite ().

SPI (Interface Série Périphérique) : Broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la bibliothèque pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Mega.

I2C : Broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou

interface TWI (Two Wire Interface - Interface "2 fils"), disponible

en utilisant la librairie Wire/I2C (ou TWI - Two-Wire interface - interface "2 fils").

LED : Broche 13. Il y a une LED incluse dans la carte connectée à la broche

13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.

d) les broches :

La carte Uno dispose de 6 entrées analogiques (numérotées de 0 à 5), chacune pouvant fournir une mesure d'une résolution de 10 bits (càd sur 1024 niveaux soit de 0 à 1023) à l'aide de la très utile fonction `analogRead ()` du langage Arduino. Par défaut, ces broches mesurent entre le 0V (valeur 0) et le 5V (valeur 1023), mais il est possible de modifier la référence supérieure de la plage de mesure en utilisant la broche AREF et l'instruction `analogReference ()` du langage Arduino.

Note : les broches analogiques peuvent être utilisées en tant que broches numériques : elles sont numérotées en tant que broches numériques de 14 à 19.

e) Autres broches :

Il y a deux autres broches disponibles sur la carte :

AREF : Tension de référence pour les entrées analogiques (si différent du 5V). Utilisée avec l'instruction `analogReference ()`.

Reset : Mettre cette broche au niveau BAS entraîne la réinitialisation (= le redémarrage) du microcontrôleur. Typiquement, cette broche est utilisée pour ajouter un bouton de réinitialisation sur le circuit qui bloque celui présent sur la carte.

f) Mémoire :

L'ATmega 328 a 32Ko de mémoire FLASH pour stocker le programme (dont 0.5Ko également utilisés par le bootloader). L'ATmega 328 a également 2ko de mémoire SRAM (volatile) et 1Ko d'EEPROM (non volatile - mémoire qui peut être lue à l'aide de la librairie EEPROM).

Pour info : Le bootloader est un programme préprogrammé une fois pour toute dans l'ATméga et qui permet la communication entre l'ATmega et le logiciel Arduino via le port USB, notamment lors de chaque programmation de la carte.

g) Communication :

La carte Arduino Uno dispose de toute une série de facilités pour communiquer avec un ordinateur, une autre carte Arduino, ou avec d'autres microcontrôleurs. L'ATmega 328 dispose d'une UART (Universal Asynchronous Receiver Transmitter ou émetteur-récepteur asynchrone

universel en français) pour communication série de niveau TTL (5V) et qui est disponible sur les broches 0 (RX) et 1 (TX). Un circuit intégré ATmega8U2 sur la carte assure la connexion entre cette communication série vers le port USB de l'ordinateur et apparaît comme un port COM virtuel pour les logiciels de l'ordinateur. Le code utilisé pour programmer l'ATmega8U2 utilise le driver standard USB COM, et aucun autre driver externe n'est nécessaire. Cependant, sous Windows, un fichier .inf est requis.

Le logiciel Arduino inclut une fenêtre terminal série (ou moniteur série) sur l'ordinateur et qui permet d'envoyer des textes simples depuis et vers la carte Arduino. Les LEDs RX et TX sur la carte clignote lorsque les données sont transmises via le circuit intégré USB-vers-série et la connexion USB vers l'ordinateur (mais pas pour les communications série sur les broches 0 et 1).

Une librairie Série Logicielle permet également la communication série (limitée cependant) sur n'importe quelle broche numérique de la carte UNO.

L'ATmega 328 supporte également la communication par protocole I2C (ou interface TWI (Two Wire Interface - Interface "2 fils") et SPI :

Le logiciel Arduino inclut la librairie Wire qui simplifie l'utilisation du bus I2C.

Pour utiliser la communication SPI (Interface Série Périphérique), la librairie pour

h) Logiciel : IDE ARDUINO

Le logiciel de programmation des modules Arduino est une application Java, libre et multi-plateformes, servant d'éditeur de code et de compilateur, et qui peut transférer le firmware et le programme au travers de la liaison série (RS232, Bluetooth ou USB selon le module). Il est également possible de se passer de l'interface Arduino, et de compiler les programmes en ligne de commande.

Le langage de programmation utilisé est le C++, compilé avec avr-g++, et lié à la bibliothèque de développement Arduino, permettant l'utilisation de la carte et de ses entrées/sorties. La mise en place de ce langage standard rend aisé le développement de programmes sur les plates-formes Arduino, à toute personne maîtrisant le C ou le C++.



Figure.2. 3 Logiciel Arduino

h.1) Le menu :

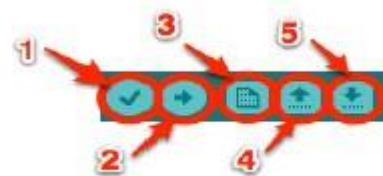
Bouton 1 : Ce bouton permet de vérifier le programme, il actionne un module qui Cherche les erreurs dans le programme.

Bouton 2 : Envoi du programme sur l'Arduino.

Bouton 3 : Créer un nouveau fichier.

Bouton 4 : Ouvrir un fichier existant.

Bouton 5 : Enregistrer un fichier.



Une fois le code écrit (ou collé) dans la fenêtre de programmation, il faut l'envoyer sur l'Arduino. Pour cela, il faut cliquer sur le bouton de vérification de programme avant d cliquer sur le bouton upload (*téléverser*), naturellement après avoir connecté l'Arduino à l'ordinateur.

i) Le déroulement du programme :

Le programme se déroule de la façon suivante :

1. Prise en compte des instructions de la partie déclarative.
2. Exécution de la partie configuration (fonction `setup()`).
3. Exécution de la boucle sans fin (fonction `loop()`): le code compris dans la boucle sans fin est exécuté indéfiniment.

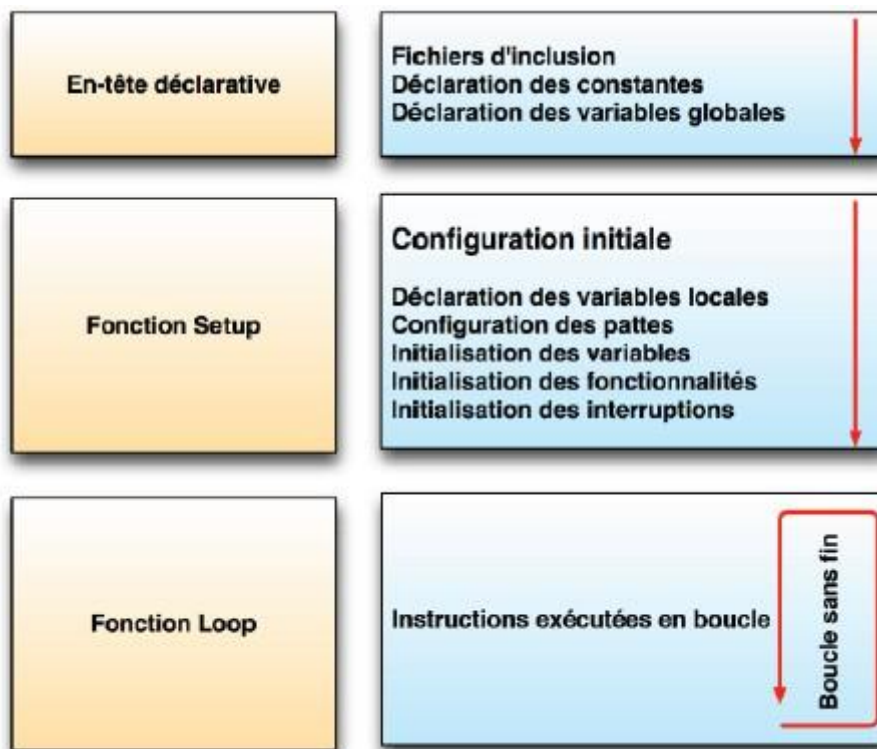


Figure 2.4 Déroulement du programme arduino

j) Fonction :

Fonction setup :

```
void setup () //fonction d'initialisation de la carte
{
//contenu de l'initialisation
//on écrit le code à l'intérieur
}
```

Cette fonction **setup ()** est appelée **une seule fois** lorsque le programme commence. C'est pourquoi c'est dans cette fonction que l'on va écrire le code qui n'a besoin d'être exécuté qu'une seule fois. On appelle cette fonction : "fonction d'initialisation". On y

retrouvera la mise en place des différentes sorties et quelques autres réglages.

Fonction Loop :

Une fois que l'on a initialisé le programme, il faut ensuite créer son "cœur", autrement dit le programme en lui-même.

```
void loop () //fonction principale, elle se répète (s'exécute) à l'infini
{
//contenu du programme
}
```

C'est donc dans cette fonction loop() que l'on va écrire le contenu du programme. Il faut savoir que cette fonction est appelée en permanence, c'est-à-dire qu'elle est exécutée une fois, puis lorsque son exécution est terminée, on le ré-exécute et encore et encore. On parle de *boucle infinie*.

2.4 Conclusion

Dans ce chapitre, nous avons détaillé la carte de prototypage Arduino, cette dernière servira comme outil de base pour la réalisation de notre application, de l'acquisition, la mémorisation à l'affichage. Nous avons surtout insisté et développé les parties que nous avons utilisé, tels que les entrées analogique, la transmission de données, l'affichage avec la syntaxe nécessaire pour la programmation. Cette étape nous a permis d'entamer le chapitre qui suit à savoir l'étape de réalisation de notre application en toute quiétude.

Chapitre 3

Réalisation

Chapitre 3 : Réalisation

3.1 Introduction

Dans cette section, nous allons passer en revue les différents logiciels que nous avons utilisés dans notre projet, les capteurs, le protocole de communication utilisé et le système d'acquisition que nous avons utilisé, ainsi que la méthodologie d'acquisition. Après cela, nous allons parler des méthodes de pré-traitement utilisées comme l'extraction de la moyenne statistique ainsi que le calcul de la RMS. En outre, nous allons concevoir différents filtres Infinité Impulse Réponse et les mettre en œuvre sur le signal acquis. Enfin, nous allons évaluer les performances des filtres en utilisant le test de classification binaire ainsi que les mesures statistiques.

3.2 Synoptique général

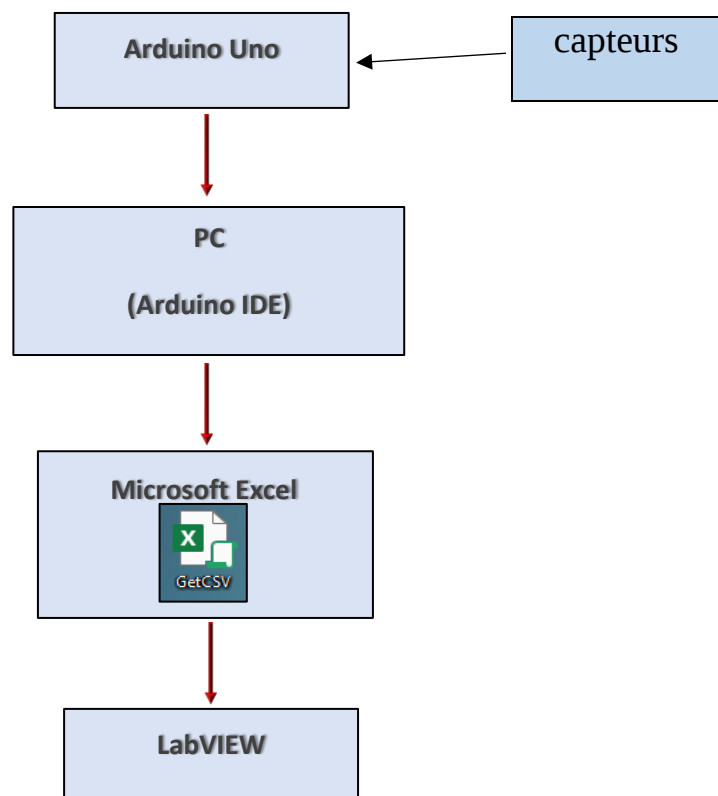


Figure 3.1. Acquisition, transforme et transfert des données.

3.2.1 Capteurs

a. accéléromètre

Un accéléromètre est un dispositif qui mesure l'accélération appropriée. Une bonne accélération, étant l'accélération (ou la vitesse de changement de vitesse) d'un corps dans son propre cadre de repos instantané [27].

Les accéléromètres sont utilisés dans de nombreux domaines tels que l'ingénierie, la biologie, l'industrie, le bâtiment et la surveillance structurelle ainsi que dans les applications médicales.

Dans le domaine de l'ingénierie, ces capteurs sont généralement utilisés pour mesurer l'accélération des véhicules, les vibrations sur les voitures, les machines, les bâtiments, les systèmes de contrôle des processus, les installations de sécurité, le transport ainsi que dans la navigation.

Dans le domaine de la biologie, l'utilisation de l'accéléromètre augmente; il est nécessaire dans l'enregistrement de l'accélération biaxiale ou triaxiale qui permet de déterminer les modèles comportementaux des animaux pendant qu'ils sont hors de vue.

Dans l'industrie, les accéléromètres sont également utilisés pour la surveillance de la santé des machines afin de signaler les vibrations et les changements dans le temps des arbres aux roulements d'équipements rotatifs tels que turbines, pompes, ventilateurs, rouleaux, compresseurs.

En outre, dans le département médical, les accéléromètres sont utilisés dans différents outils de surveillance de la santé pour les coureurs et autres sports, ainsi. Récemment, il a été suggéré de construire des casques de football américain avec un accéléromètre intégré pour mesurer l'impact des collisions de tête [28].

Avantages	Inconvénients
• Il est simple à interface et robuste dans la conception. • Il a une grande impédance. • Il offre une sensibilité plus élevée. • Il a une réponse à haute fréquence. • Il est disponible à moindre coût en raison de l'avancement de la technologie MEMS • Il utilise un circuit intégré de conditionnement de signal pour mesurer la capacité.	• Un accéléromètre mesure le risque de vitesse seulement. Il ne mesure pas une vitesse constante. • Un accéléromètre ne peut pas mesurer la rotation autour de son propre axe de mouvement. Pour cette raison, il est utilisé en conjonction avec un gyroscope pour mesurer la vitesse angulaire. • Il est sensible à la température et fonctionne sur une plage de température limitée • Son efficacité se dégrade avec le temps. Il a besoin d'une puissance externe pour son fonctionnement.

Tableau 3.1 Avantages et inconvénients d'accéléromètre

Nous avons choisi l'accéléromètre comme capteur pour notre application, en raison de l'insignifiance des inconvénients des accéléromètres sur notre application. En outre, en raison des avantages écrasants pour notre application, aussi parce que l'examen de la littérature montre que plusieurs recherches ont utilisé le même capteur (accéléromètre).

a. MPU6050

L'UMP6050 est un micro-électro-mécanique (MEMS) composé d'un accéléromètre à 3 axes et d'un gyroscope à 3 axes. Cela nous aide à faire de multiples mesures comme l'accélération, la vitesse, l'orientation, le déplacement et de nombreux autres paramètres liés au mouvement. Ce module dispose également d'un processeur de mouvement numérique (DMP) à l'intérieur, qui est assez puissant pour effectuer des calculs complexes et ainsi libérer le travail pour Microcontrôleur [29].

3.2.2 Protocole de communication inter-intégré de circuit

I2C est un protocole de série pour une interface à deux fils pour connecter des appareils à basse vitesse comme des microcontrôleurs, des interfaces informatiques et d'autres périphériques similaires dans les systèmes embarqués. Philips Semi-conducteur l'a inventé en 1982 [30]. Aujourd'hui, presque tous les grands fabricants de circuits intégrés l'utilisent.

Chaque appareil esclave I2C a besoin d'une adresse – ils doivent encore être obtenus auprès de NXP (anciennement semi-conducteurs Philips).

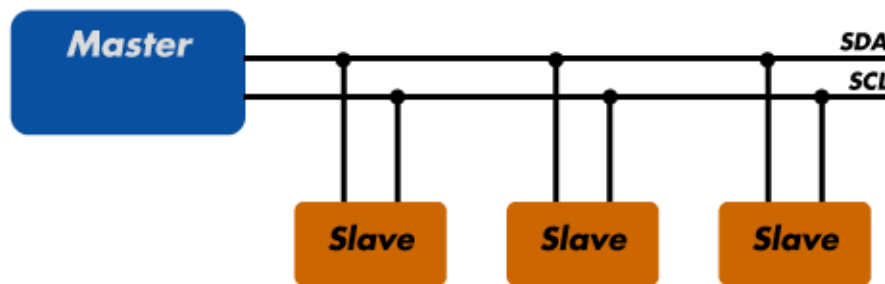


Figure 2.2. Exemple d'implémentation I2C [31].

L'interface à deux fils que nous avons mentionnée ci-dessus est l'utilisation de deux fils pour connecter le Maître avec les Esclaves; ces deux fils sont l'horloge série (SCL) et les données en série (SDA), 2. La figure montre une description de base du protocole de communication I2C. Nous avons choisi ce protocole de communication en raison de ses avantages significatifs par rapport à ses inconvénients qui n'ont aucun impact sur notre étude.

Avantages	Inconvénients
• Flexibilité : le protocole I2C prend en charge la communication multi-maître et multi-esclave qui permet d'ajouter beaucoup de fonctionnalités à votre conception. • Fonction d'adressage : l'utilisation d'une adresse aux puces permet d'ajouter facilement des composants à l'autobus sans aucune complexité. • Simplicité : Il suffit de deux lignes de signal bidirectionnelles pour établir la communication entre plusieurs appareils. Meilleur mécanisme de gestion des erreurs : le protocole I2C repose sur la fonctionnalité ACK/NACK, qui est une fonctionnalité de correction d'erreur robuste. ACK signifie Reconnaissance, tandis que NACK signifie Pas de reconnaissance.	• Conflits : En raison de l'adresse à puce, il y a toujours une possibilité d'un conflit d'adresse. • Vitesses plus lentes : le protocole I2C utilise des résistances de traction plutôt que celles utilisées par ses pairs. En raison de la conception à drain ouvert, la vitesse est limitée.

Tableau 3.2 Avantages et inconvénients I2C

a. LabVIEW

Laboratoire Virtual Instrument Engineering Workbench (LabVIEW); est une plate-forme de conception de systèmes et un environnement de développement pour un langage de programmation visuelle de National Instruments. Sorti pour l'Apple Macintosh en 1986 (il y a environ 33 ans), il est généralement utilisé pour l'acquisition de données, le contrôle d'instruments et l'automatisation industrielle sur une variété de systèmes d'exploitation (OS), y compris Microsoft Windows, diverses versions d'Unix, Linux et macOS [23]. La version que nous utilisons est LabVIEW 2014.

L'utilisation de LabVIEW présente un très large éventail d'avantages comme :

1. Interface utilisateur graphique : les professionnels du design utilisent la bibliothèque d'interface utilisateur de glisser-déposer en personnalisant de manière interactive les centaines d'objets utilisateur intégrés de la palette de contrôles.
2. Conception modulaire et conception hiérarchique : exécutez des VI LabVIEW modulaires par eux-mêmes ou sous-VIS et développez facilement et modularisent les programmes en fonction de l'application.
3. Plusieurs outils de développement de haut niveau : Développez-vous plus rapidement avec des outils de développement spécifiques à l'application, y compris le module Statechart LabVIEW, LabVIEW Control Design, et le module de simulation et le module LabVIEW FPGA.
4. Flexibilité et évolutivité : Les ingénieurs et les scientifiques ont des besoins et des exigences qui peuvent changer rapidement. Ils doivent également avoir des solutions extensibles et durables qui peuvent être utilisées pendant une longue période. En créant des instruments virtuels basés sur des logiciels de développement puissants tels que LabVIEW, vous concevez en soi un cadre ouvert qui intègre parfaitement les logiciels et le matériel. Cela garantit que vos applications fonctionnent non seulement bien aujourd'hui, mais que vous pouvez facilement intégrer les nouvelles technologies à l'avenir.
5. Connectivité et contrôle des instruments : LabVIEW dispose de bibliothèques prêtes à l'emploi pour l'intégration d'instruments autonomes, de dispositifs d'acquisition de données, de commandes de mouvement et de produits de vision, et des automates pour créer une solution complète de mesure et d'automatisation.
6. Environnement ouvert : LabVIEW fournit les outils nécessaires à la plupart des applications et est un environnement de développement ouvert. Cette langue ouverte

tire parti du code existant; il peut facilement s'intégrer avec les systèmes hérités et intégrer des logiciels tiers avec .NET, ActiveX, DLL, objets, TCP, services Web et formats de données XML.

7. Gestion des cibles : Gérez facilement plusieurs cibles, des appareils en temps réel aux périphériques embarqués, y compris les FPGA, les microprocesseurs, les microcontrôleurs, les PDA et les panneaux tactiles.
8. Conception orientée objet : Utilisez des structures de programmation orientées objet pour tirer parti de l'encapsulation et de l'héritage pour créer du code modulaire et extensible.
9. Conception d'algorithmes : Développez des algorithmes à l'aide d'une programmation textuelle axée sur les mathématiques et de la syntaxe interactive du script de fichier.m avec LabVIEW MathScript.

3.3 Méthodes de traitement du signal utilisées

Dans cette section, nous passons par différentes méthodes mathématiques que nous allons l'utilisés comme outils pour notre partie de traitement du signal.

3.3.1 Transformation de fourier

a. Qui est Fourier?

Fourier est né à Auxerre (aujourd'hui dans le département de l'Yonne); son père était un tailleur modeste; malheureusement, ses parents sont morts quand il avait neuf ans. Il a pris un cours militaire sur les mathématiques, et dans cette période, il a pris une part importante dans sa propre région dans la promotion de la Révolution Française, siégeant au Comité révolutionnaire local. Il fut brièvement emprisonné pendant la Terreur, mais, en 1795, fut nommé à l'École Normale et succéda par la suite à Joseph-Louis Lagrange à l'École Polytechnique.

En 1798, Fourier eut l'honneur de rejoindre Napoléon Bonaparte dans son expédition égyptienne en tant que conseiller scientifique et fut nommé secrétaire de l'Institut d'Égypte. Il a également contribué plusieurs articles mathématiques à l'Institut égyptien (également appelé l'Institut du Caire), que Napoléon a fondé au Caire. En 1801, Fourier retourne en France, [32] Fourier était déjà rentré de l'expédition napoléonienne en Égypte pour reprendre son poste académique de professeur à l'École Polytechnique lorsque Napoléon en décida autrement dans sa remarque et le nomma gouverneur du département de l'Isère à Grenoble.

D'où le fidèle de Napoléon, il prend le poste de préfet. C'est à Grenoble qu'il commence à expérimenter la propagation de la chaleur [32]. Le 21 décembre 1807, il présente son article sur la propagation de la chaleur dans les corps solides. Il a également contribué à la « Description monumentale de l'Égyptien » [33].

En 1830, sa santé affaiblie commence à prendre sa place car il avait déjà subi quelques crises d'anévrisme du cœur alors qu'il était en Égypte et à Grenoble. A Paris, il était impossible de se tromper quant à la cause première des suffisions fréquentes qu'il a vécues. Une chute, qu'il subit le 4 mai 1830, alors qu'il descendait les escaliers d'un avion, aggrave la maladie dans une mesure au-delà de ce qui aurait pu être craint [33]. Peu de temps après cet événement, il mourut dans son lit le 16 mai 1830.

Fourier a été enterré dans le « cimetière du Père Lachaise » à Paris, une tombe décorée d'un motif égyptien pour refléter sa position de secrétaire de l'Institut du Caire et sa collection de « Description de l'Égypte ». Son nom est l'un des 72 noms inscrits sur la Tour Eiffel.

b. De la série Fourier à la transformation Fourier

La série Fourier est nommée en l'honneur de Jean-Baptiste Joseph Fourier, qui a apporté d'importantes contributions à l'étude des séries trigonométriques. Après les enquêtes préliminaires de Leonhard Euler, Jean le Rond d'Alembert, et Daniel Bernoulli. Fourier a présenté la série pour résoudre l'équation thermique dans une plaque de métal, publiant ses

premiers résultats dans son article de 1807 « Mémoire Sur la propagation de la chaleur dans les corps solides », qui peut être traduit en anglais et cela signifie « Traité sur la propagation de la chaleur dans les corps solides ». En plus de publier son article « Théorie analytique de la chaleur », qui se traduit par « Théorie analytique de la chaleur » en 1822. Le premier article a introduit l'analyse Fourier, en particulier la série Fourier. La recherche de Fourier a été établie qu'une fonction arbitraire (continue) [34] peut être représentée par une série trigonométrique. Fourier fit la première annonce de cette grande découverte en 1807. Les premières idées de décomposition d'une fonction périodique en une somme d'autres fonctions oscillantes simples remontent au III^e siècle av. J.-C. lorsqu'un couple d'astronomes anciens a proposé un modèle empirique de mouvements planétaires, basé sur des déferents et des épicycles.

L'équation thermique sur laquelle Fourier faisait ses études était une équation différentielle partielle. Avant le travail de Fourier, il n'y avait pas de solution à ce genre d'équations. Bien que des solutions particulières aient été connues si la source de chaleur se comportait d'une manière simple, en particulier, si la source de chaleur était une onde sinusoïdale ou Cosinus, ces solutions simples sont maintenant parfois appelées solutions Eigen. L'idée de Fourier était de remplacer le modèle d'une source de chaleur compliquée par une simple combinaison linéaire d'ondes sinusoïdes et Cosinus simples, puis d'écrire la solution de ce système comme une combinaison linéaire des solutions Eigen correspondantes. Cette combinaison linéaire s'appelle maintenant la série Fourier.

D'un point de vue actuel, les résultats de Fourier étaient quelque peu informels en raison de l'absence d'une compréhension précise de la fonction et des intégrales au début du XIX^e siècle. Après cette période, Peter Gustav Lejeune Dirichlet [35] et Bernhard Riemann [36] [37], exprima les résultats de Fourier avec plus de précision et de formalité.

Puisque la série Fourier est une sorte de résumé d'une autre fonction, on peut clairement dire qu'il s'agit d'une synthèse. En outre, la transformation fourrier à temps discret est un exemple de synthèse, et nous pouvons dire aussi que le processus de dérivation des poids qui décrivent une fonction donnée est une forme d'analyse Fourier. Pour les fonctions à intervalles non liés, les analogies d'analyse et de synthèse, en général, sont fourrier transformer et inversement Fourier transformer.

c. Définition de la transformation de Fourier :

La transformation Fourier décompose une fonction du temps en fréquences qui le composent. La transformation Fourier s'appelle la représentation du domaine de fréquence du signal d'origine. Le terme Transformation Fourier fait référence à la fois à la représentation du domaine de fréquence et à l'opération mathématique qui associe la représentation du domaine de fréquence à une fonction du temps. La transformation Fourier ne se limite pas aux fonctions du temps, mais pour avoir une langue unifiée, le domaine de la fonction d'origine est communément appelé le domaine de temps. Pour de nombreuses fonctions d'intérêt pratique, on peut définir une opération qui inverse cela : la transformation inverse de Fourier, également appelée synthèse Fourier, d'une représentation de domaine de fréquence, combine les contributions de toutes les différentes fréquences pour récupérer la fonction originale du temps [38].

Les opérations linéaires effectuées dans un domaine ont des opérations correspondantes dans l'autre domaine, qui sont parfois plus faciles à exécuter. Le fonctionnement de la différenciation dans le domaine temporel correspond à la multiplication par la fréquence, de sorte que certaines équations différentielles sont plus faciles à analyser dans le domaine de fréquence. En outre, la convolution dans le domaine de temps correspond à la multiplication ordinaire dans le domaine de fréquence. Concrètement, cela signifie que tout système linéaire invariant de temps, tel qu'un filtre appliqué à un signal, peut être exprimé relativement simplement comme une opération sur les fréquences, après avoir effectué les opérations souhaitées, la transformation du résultat peut être faite de nouveau au domaine de temps.

La transformation Fourier peut être formellement définie comme une intégrale Riemann inappropriée, ce qui en fait une transformation intégrale, bien que cette définition ne convient pas à de nombreuses applications nécessitant une théorie d'intégration plus sophistiquée.

d. Description mathématique de la transformation Fourier

La transformation Fourier est une généralisation de la série Fourier [39]; À proprement parler, elle ne s'applique qu'aux fonctions continues et apériodiques, mais l'utilisation de la fonction d'impulsion permet l'utilisation de signaux discrets. La transformation Fourier est définie comme :

$$\mathbf{F(j\omega)} = \int_{-\infty}^{+\infty} \mathbf{f(t)}\mathbf{e^{-j\omega t}}\mathbf{dt} \quad (1.)$$

La transformation inverse est définie comme :

$$\mathbf{f(t)} = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \mathbf{F(j\omega)}\mathbf{e^{j\omega t}}\mathbf{dt} \quad (2.)$$

Où:

$\mathbf{f(t)}$: est le signal.

$\mathbf{F(j\omega)}$: est le signal de transformation Fourier.

e. Applications de la transformation Fourier

Même si la motivation originale de la série Fourier ou de la transformation Fourier était de résoudre l'équation thermique, il est devenu plus tard clair que les mêmes techniques pouvaient être appliquées à un large éventail de problèmes mathématiques et physiques, et en particulier ceux impliquant des équations différentielles linéaires avec des coefficients constants. La série Fourier et la transformation Fourier ont de nombreuses applications telles que : conception et utilisation d'antennes, traitement d'image, par exemple, dans le délupage de flou, dans le traitement de signal comme les filtres de passage élevé, de passage bas et de bande-pass, également des fonctions de corrélation croisée et de transfert [38].

f. La transformation de fourrier rapide

Une transformation Fourier rapide est un algorithme qui calcule la transformation fourrier discrète (DFT) d'un ensemble ou d'une séquence de valeurs, ou son inverse (IDFT). Le DFT est obtenu en décomposant un ensemble particulier de valeurs en composants de différentes fréquences. Cette opération est utile dans de nombreuses situations et champs. Toutefois,

l'informatique de ces composants directement à partir de la définition standard prendra beaucoup de temps; pour cela, nous allons faire appel pour ce qu'on appelle une FFT, ou transformation de fourrier rapide. La transformation de fourrier rapide calcule rapidement ces transformations en factorisant la matrice DFT en un produit de facteurs clairsemés (pour la plupart nuls). En conséquence, il parvient à réduire la complexité de l'informatique de la DFT. La différence de vitesse peut être énorme, essentiellement lorsque nous avons affaire à de longs ensembles de données. Il existe de nombreux algorithmes fft différents basés sur un large éventail de théories publiées, de l'arithmétique simple des nombres complexes à la théorie des groupes et à la théorie des nombres. Ces algorithmes sont extrêmement utilisés dans différents domaines comme l'ingénierie, la science et les mathématiques [40].

3.3.2 Root-Mean-Square

Root-Mean-Square (RMS) est défini comme la racine carrée du carré moyen (la moyenne arithmétique des carrés d'un ensemble de nombres) Le RMS est également connu sous le nom de moyenne quadratique.

La valeur RMS d'un ensemble de valeurs (ou d'une forme d'onde en temps continu) est la racine carrée de la moyenne arithmétique des carrés des valeurs, ou le carré de la fonction qui définit la forme d'onde continue. Dans notre application, nous avons trois formes d'ondes d'accélération; l'accélération dans la direction de X, Y et Z. Nous avons appliqué le RMS selon l'équation suivante :

$$S_{RMS}(t) = \sqrt{x(t)^2 + y(t)^2 + z(t)^2} \quad (3.)$$

Comme nous l'avons mentionné [14] [15] [16], nous avons utilisé le RMS, parce que nous voulons éviter de traiter les rotations involontaires des capteurs pendant les mouvements maternels [14] [15], et nous nous intéressons à la présence ou à l'absence du mouvement plutôt qu'à sa direction [16].

3.3.3 Filtrage numérique

Les filtres sont des systèmes qui traitent certains signaux du côté du domaine de fréquence et ces systèmes sont fabriqués à l'aide de condensateurs et d'inducteurs. Le changement de ces dernières valeurs donne lieu à un changement de réponse de fréquence de la fonction de transfert du circuit. Ces circuits peuvent être de simples diviseurs de tension où la jambe de shunt est une impédance réactive. Au fur et à mesure que la fréquence est modifiée, la valeur de l'impédance réactive change et le rapport de division de tension change. Les filtres peuvent être utilisés dans de nombreuses applications pratiques. Si l'on regarde un filtre simple, à pôle unique et à passage bas (l'intégrateur) est souvent utilisé pour stabiliser les amplificateurs en roulant le gain à des fréquences plus élevées où un décalage de phase excessif peut provoquer des oscillations. En outre, un filtre simple à pôle unique et à passage élevé peut être utilisé pour bloquer le décalage DE COURANT dans les amplificateurs à gain élevé ou les circuits d'alimentation uniques. Les filtres peuvent être utilisés pour séparer les signaux, en passant ceux d'intérêt, et en atténuant les fréquences indésirables [41].

Un filtre idéal aura une réponse d'amplitude qui est l'unité ou à n'importe quelle valeur fixe ou gain pour les fréquences d'intérêt et il est appelé la bande de passage, et zéro sur tous les autres intervalles et qui est appelé la bande d'arrêt. La fréquence de coupure est la valeur de

fréquence à laquelle la réponse passe de la bande de passage à la bande d'arrêt. Figure 2.3 montre plusieurs types de filtres idéalisés.

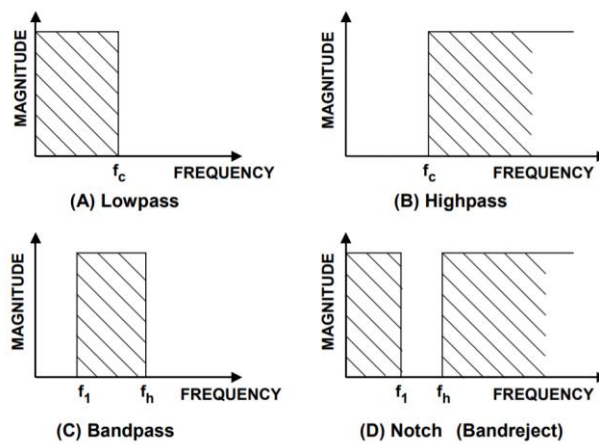


Figure 3.3. Réponses de filtre idéalisées [41].

Les filtres idéalisés définis ci-dessus, malheureusement, ne peuvent pas être facilement construits. La transition d'une bande de passage à une bande d'arrêt ne sera pas immédiate, mais il y aura plutôt une région de transition. Les paramètres d'un filtre pratique peuvent être vus dans. Figure2.4.

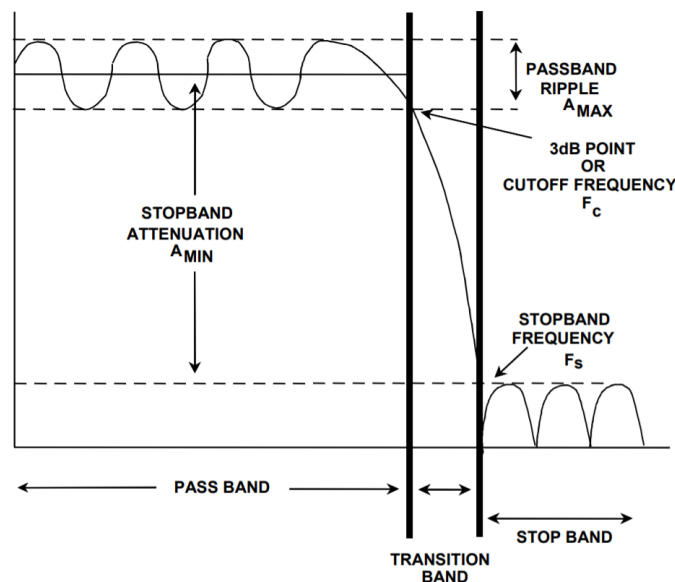


Figure 3.4. Paramètres du filtre de clé [41].

Ces cinq paramètres sont les suivants :

1. La fréquence De coupure f_c : c'est là que la réponse du filtre quitte la bande d'erreur.
2. La fréquence d'arrêt f_s : c'est là que la réponse du filtre atteint le minimum d'atténuation.

3. L'ondulation de la bande de passage $A_{\max}$: est la variation (bande d'erreur) dans la réponse de la bande de passage.
4. L'atténuation de la bande d'arrêt $A_{\min}$: définit l'atténuation minimale du signal dans la bande d'arrêt.

a. Types de filtrage

En général, nous pouvons classer le filtre en deux catégories la première série de filtres que nous appellerons ou les catégoriserons pour être des filtres de réponse d'impulsion finies, et les deuxièmes peuvent être des filtres de réponse d'impulsion infinies.

a.1. Filtres de réponse d'impulsion finie

Le type de filtre (FIR) est probablement le filtre le plus courant dans les communications numériques; la raison de leur utilisation vient du fait que les filtres de FIR symétriques sont également classés comme filtres de phase linéaire. Les filtres affecteront non seulement l'ampleur de leurs entrées, mais ils captent également une certaine distorsion de phase au signal de sortie. Le terme phase linéaire signifie que (dans la bande de passage du filtre) l'influence de phase des coefficients de filtre sur l'entrée est une fonction linéaire. Cette distorsion de phase linéaire se termine par un délai constant (et facilement enlevé). Dans certaines applications vocales, les effets de phase de filtre ne sont généralement pas importants en raison de la façon dont l'oreille humaine réagit à la phase. Cependant, dans les communications numériques, la phase de notre signal est très importante pour la détection et la démodulation correctes [42].

a.2. Filtres de réponse d'impulsion infinie

Les filtres de réponse d'impulsion infinie (IIR) sont récursifs et ne présentent pas de phase linéaire. Bien que cette caractéristique ne soit pas souhaitable pour la plupart des filtrages dans les communications numériques, il peut y avoir des cas où vous voulez obtenir plus d'atténuation avec un filtre d'ordre inférieur. Dans de nombreux cas, un filtre IIR de moindre ordre peut correspondre ou battre les performances du filtre d'un filtre FIR, ce qui peut se traduire par des améliorations de vitesse pour l'opération de filtrage [42].

La réponse d'impulsion infinie est une propriété s'appliquant à de nombreux systèmes linéaires invariants temporels. Les exemples courants de systèmes linéaires invariants dans le temps sont la plupart des filtres électroniques et numériques. Les systèmes avec cette propriété sont connus sous le nom de systèmes IIR ou filtres IIR, et se distinguent par une réponse d'impulsion, qui ne devient pas exactement zéro au-delà d'un certain point, mais continue indéfiniment [43]. Ces attributs nous ont obligés à choisir les filtres IIR comme une approche appropriée pour notre système.

b. Types de réponse de filtre

b.1. Filtre passe bas

Un filtre à passage bas (LPF) est un filtre qui transmet les signaux avec une fréquence inférieure à une fréquence de coupure sélectionnée et atténue les signaux avec des fréquences supérieures à la fréquence de coupure. La réponse exacte de la fréquence du filtre dépend de la conception du filtre.

Les filtres à passage bas existent sous différentes formes, y compris les circuits électroniques tels qu'un filtre à sifflement utilisé dans l'audio ou une autre main comme les filtres numériques pour lisser les ensembles de données. Les filtres à passage bas fournissent une forme plus fluide de signal, éliminant les variations rapides et à court terme et laissant la tendance à long terme [44].

Un filtre à passage bas idéal élimine toutes les fréquences au-dessus de la fréquence de coupure tout en passant celles qui sont inférieures à l'impôt; sa réponse de fréquence est une fonction rectangulaire (Figure 2.5). La région de transition parfaite de la bande de passage à la bande d'arrêt n'existe pas dans les filtres pratiques; par conséquent, un filtre idéal n'existe pas. Un filtre à passage bas idéal peut être réalisé mathématiquement (théoriquement) en multipliant un signal par la fonction rectangulaire dans le domaine de fréquence ou, de façon équivalente, la convolution avec sa réponse d'impulsion, une fonction « sinc », dans le domaine du temps.

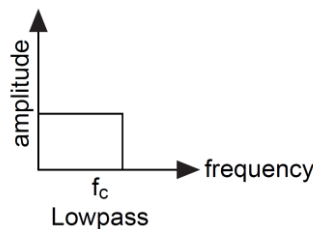


Figure 3.5. Filtre passe-bas idéal [45].

Cependant, le filtre idéal est impossible à réaliser sans avoir également des signaux d'une étendue infinie dans le temps, et donc généralement doit être approximatif pour les signaux réels en cours, parce que la région de soutien de la fonction « sinc » s'étend à tous les temps passés et futurs. Il est effectivement réalisable pour les signaux numériques préenregistrés en assumant des extensions de zéro dans le passé et l'avenir, ou plus généralement en rendant le signal répétitif et en utilisant l'analyse Fourier.

b.2. Filtre passe-haut

Un filtre passe-haut (HPF) est un filtre qui transmet des signaux avec une fréquence supérieure à une fréquence de coupure sélectionnée et atténue les signaux avec des fréquences inférieures à la fréquence de coupure. La réponse exacte de la fréquence du filtre dépend de la conception du filtre. La quantité d'atténuation pour chaque fréquence dépend de la conception du filtre. Un filtre passe-haut est habituellement modelé comme un système linéaire de temps-invariant. Les filtres passe-haut ont beaucoup d'utilisations, telles que le blocage de DC des circuits sensibles aux tensions moyennes non-zéro ou aux dispositifs de radiofréquence. Ils peuvent également être utilisés conjointement avec un filtre passe-bas pour produire un filtre passe-bande [46].

Un filtre passe-haut idéal élimine toutes les fréquences inférieures à la fréquence de coupure tout en passant celles qui sont au-dessus de l'inchangé ; sa réponse de fréquence est une fonction rectangulaire (Figure 2.6). La région de transition parfaite de la bande d'arrêt à la bande-passe n'existe pas dans les filtres pratiques; par conséquent, un filtre passe-haut idéal n'existe pas comme le passe-bas.

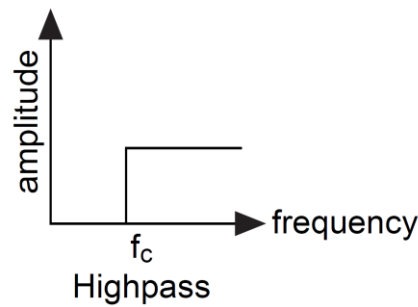


Figure 3.6. Filtre passe haut idéal [45].

b.3. Filtre pass-bande

Un filtre pass-bande (BPF) est un filtre qui passe des signaux avec une fréquence inférieure à la fréquence de coupure sélectionnée inférieure et atténue les signaux avec des fréquences supérieures à la fréquence de coupure sélectionnée plus élevée. La transition hors de la bande-passe aurait des caractéristiques de mur de briques tout comme. Figure 2.7 montre.

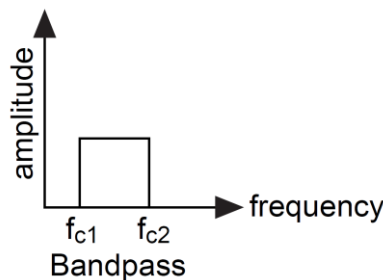


Figure 3.7 Filtre passe-bande idéal [45]

En pratique, aucun filtre passe-bande n'est idéal. Le filtre n'atténue pas complètement toutes les fréquences en dehors de la plage de fréquences souhaitée ; en particulier, il y a une région juste à l'extérieur de la bande-passe prévue où les fréquences sont atténuées, mais pas rejetées. Il s'agit du déploiement de filtre. Généralement, la conception d'un filtre cherche à rendre le déploiement aussi étroit que possible, permettant ainsi au filtre d'effectuer le plus près possible de sa conception prévue. Souvent, cela se réalise au détriment de la bande passe ou de l'ondulation de bande [47].

b.4. Bande-stop, filtre Notch

Un filtre de bande-stop ou également appelé filtre de bande-rejet est un filtre qui passe la plupart des fréquences inchangées et atténue ceux dans une gamme spécifique à des niveaux très bas. Fondamentalement, il agit inversement à la bande-filtre passe. Un filtre à encoche est un filtre de bande-stop avec une bande d'arrêt étroite [48].

Le filtre Band-stop peut être représenté comme une combinaison de filtres passe-bas et de haut-passe, tout comme la montre Figure 2.8.

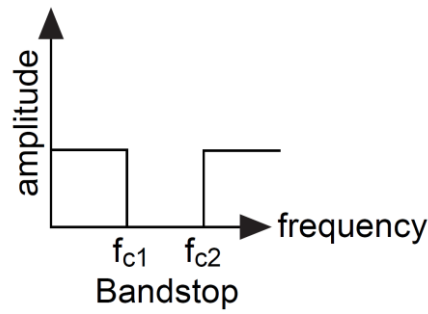


Figure 3.8. Filtre bande-stop idéale [45].

c. Algorithme de conception de filtre

Différents algorithmes de conception existent pour les filtres IIR comme Butterworth, Chebyshev, Elliptic et Inverse Chebyshev. Dans cette partie, nous allons passer en revue chacun des algorithmes mentionnés précédemment.

c.1. Filtre Butterworth

Les filtres passe-bas de Butterworth sont conçus pour avoir une caractéristique de réponse en amplitude aussi plate que possible aux basses fréquences, et qui diminue de façon monotone avec l'augmentation de la fréquence [49].

Fonction de transfert:

L'expression générale de la fonction de transfert d'un filtre passe-bas Butterworth d'ordre n est donnée par:

$$H(s) = \frac{1}{\prod_{i=1}^n (s-s_i)} = \frac{1}{(s-s_1)(s-s_2)\dots(s-s_n)} \quad (4.)$$

Le tableau suivant montre les facteurs de polynômes du filtre Butterworth

n	Facteurs de Polynomial $B_n(s)$
1	$(s + 1)$
2	$(s^2 + 1.4142s + 1)$
3	$(s + 1)(s^2 + s + 1)$
4	$(s^2 + 0.7654s + 1)(s^2 + 1.8478s + 1)$
5	$(s + 1)(s^2 + 0.6180s + 1)(s^2 + 1.6180s + 1)$
6	$(s^2 + 0.5176s + 1)(s^2 + 1.4142s + 1)(s^2 + 1.9319s + 1)$
7	$(s + 1)(s^2 + 0.4450s + 1)(s^2 + 1.2470s + 1)(s^2 + 1.8019s + 1)$

8	$(s^2 + 0.3902s + 1)(s^2 + 1.1111s + 1)(s^2 + 1.6629s + 1)(s^2 + 1.9616s + 1)$
9	$(s + 1)(s^2 + 0.3473s + 1)(s^2 + s + 1)(s^2 + 1.5321s + 1)(s^2 + 1.879s + 1)$
10	$(s^2 + 0.3129s + 1)(s^2 + 0.9080s + 1)(s^2 + 1.4142s + 1)(s^2 + 1.7820s + 1)(s^2 + 1.9754s + 1)$

Tableau 3.3. Polynomial normalisés de Butterworth [49].

Réponse de fréquence

La réponse à la fréquence Butterworth est fournie en Figure 2.9 qui montre la réponse de magnitude pour les filtres Butterworth de diverses commandes.

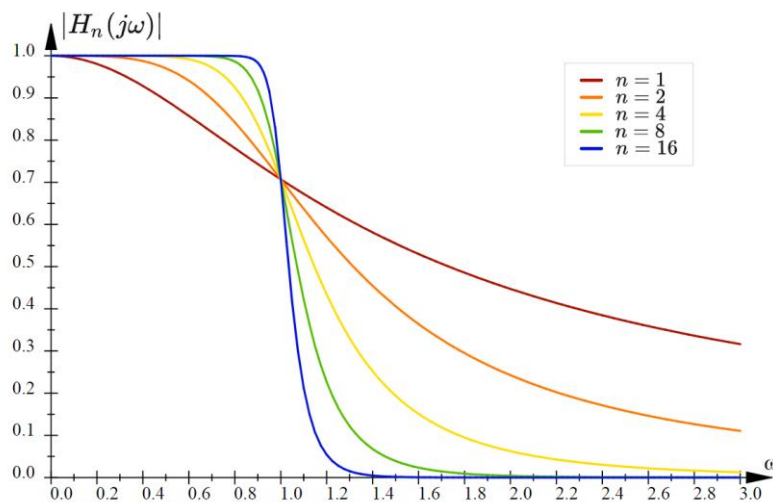


Figure 3.9. Réponse de magnitude de filtre de Butterworth pour différents ordres [50].

c.2 Filtre de Chebyshev

Les filtres Chebyshev sont conçus pour avoir une caractéristique de réponse en amplitude qui a une transition relativement nette de la bande passante à la bande d'arrêt. Cette netteté est obtenue au détriment des ondulations qui sont introduites dans la réponse. Plus précisément, les filtres Chebyshev sont obtenus comme une approximation en équirépartition de la bande passante d'un filtre passe-bas idéal. Il en résulte une caractéristique de filtre pour laquelle sa réponse en amplitude [49] est donnée par:

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 T_n^2(\omega)} \quad (5.)$$

Où:

$$\epsilon^2 = 10^{\frac{r}{10}} - 1$$

$T_n(\omega)$ = Chebyshev Ploynominal de l'ordre n.

r = ondulation de passage. dB.

Les polynômes de Chebyshev jouent un rôle important dans la théorie de l'approximation [51]. Le polynôme de n ordre Chebyshev peut être calculé en utilisant l'équation suivante :

$$T_n(\omega) = \cos(N \cos^{-1}(\omega)) \quad , |\omega| \leq 1 \quad (6.)$$

$$T_n(\omega) = \cosh(N \cosh^{-1}(\omega)) \quad , |\omega| > 1 \quad (7.)$$

Les premiers polynômes de Chebyshev sont listés dans 7.Table. À l'aide de

$T_0(\omega) = 1$ et $T_1(\omega) = \omega$, les polynômes de Chebyshev peuvent être générés de façon récursive en utilisant la relation :

$$T_{n+1}(\omega) = 2\omega T_n(\omega) - T_{n-1}(\omega) \quad (8.)$$

n	$T_n(\omega)$
0	1
1	ω
2	$2\omega^2 - 1$
3	$4\omega^3 - 3\omega$
4	$8\omega^4 - 8\omega^2 + 1$
5	$16\omega^5 - 20\omega^3 + 5\omega$
6	$32\omega^6 - 48\omega^4 + 18\omega^2 - 1$
7	$64\omega^7 - 112\omega^5 + 56\omega^3 - 7\omega$
8	$128\omega^8 - 256\omega^6 + 160\omega^4 - 32\omega^2 + 1$
9	$256\omega^9 - 576\omega^7 + 432\omega^5 - 120\omega^3 + 9\omega$
10	$512\omega^{10} - 1280\omega^8 + 1120\omega^6 - 400\omega^4 + 50\omega^2 + 1$

Tableau 3.4. Polynôme de filtre de Chebyshev [49].

Fonction de transfert :

L'expression générale pour la fonction de transfert [49] d'un filtre de bas-ordre de bas ordre de Chebyshev est donnée par :

$$H(s) = \frac{H_0}{\prod_{i=1}^n (s-s_i)} = \frac{H_0}{(s-s_1)(s-s_2)\dots(s-s_n)} \quad (9.)$$

Où :

$$H_0 = \begin{cases} \prod_{i=1}^n (-s_i) & \text{n odd} \\ 10^{r/20} \prod_{i=1}^n (-s_i) & \text{n even} \end{cases}$$

$$s_i = \sigma_i + j\omega_i$$

$$\sigma_i = \left[\frac{1-\gamma}{2} \right] \sin \frac{(2i-1)\pi}{2n}$$

$$\omega_i = \left[\frac{1-\gamma}{2} \right] \cos \frac{(2i-1)\pi}{2n}$$

$$\gamma = \left(\frac{1+\sqrt{1+\varepsilon^2}}{\varepsilon} \right)^{\frac{1}{n}}$$

$$\varepsilon = \sqrt{10^{\frac{r}{10}} - 1}$$

Les formules de pôles sont un peu plus compliquées que pour le filtre Butterworth, et plusieurs paramètres comme ε , γ et r ; doivent être déterminés avant que les valeurs du pôle puissent être calculées. En outre, tous les pôles sont impliqués dans le calcul du numérateur H_0

Réponse de fréquence :

Figure 2.10 montre la réponse de magnitude pour le filtre de Chebyshev qui ont des ordres divers avec des limites d'ondulation de bande passante de δ et une fréquence de coupure ω_c

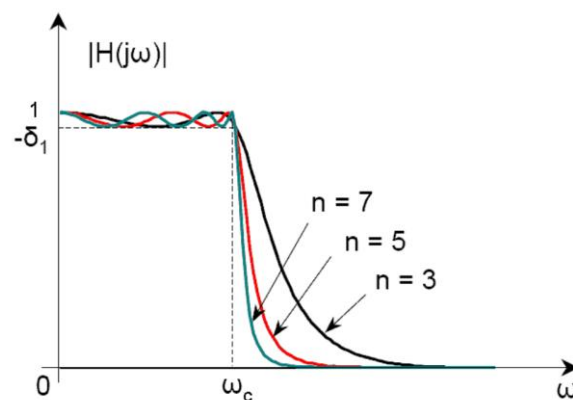


Figure 3.10. Réponse de magnitude de filtre de Chebyshev pour différents ordres [52]

c.3 Filtre elliptic

En autorisant les ondulations dans la bande passante, les filtres de Chebyshev obtiennent une meilleure sélectivité que les filtres Butterworth. Les filtres elliptiques améliorent les performances des filtres de Chebyshev en permettant des ondulations dans la bande passante et la bande d'arrêt. La réponse d'un filtre elliptique satisfait [49] :

$$|H(j\omega)|^2 = \frac{1}{1+\varepsilon^2 R_n^2(\omega, L)} \quad (10.)$$

Où $R_n^2(\omega, L)$ est une fonction rationnelle de Chebyshev d'ordre n avec le paramètre d'ondulation L . Les filtres elliptiques sont parfois appelés filtres de Cuuer

Spécification de paramètre :

Comme indiqué dans les titres précédents, la détermination de la fonction de transfert d'un filtre passe-bas Butterworth ne nécessite que deux paramètres : la fréquence de coupure et l'ordre de filtre n . En outre, la détermination de la fonction de transfert pour un filtre de Chebyshev nécessite les deux paramètres précédents plus une ondulation de bande passante. En outre, la détermination de la fonction de transfert pour un filtre elliptique nécessite spécification de l'ordre de filtre n plus les quatre paramètres suivants [49], qui sont représentés dans Figure 2.11:

A_p = perte maximale de la bande passe dans dB.

A_s = perte de bande d'arrêt minimum dans dB.

ω_p = fréquence de coupure de la bande passe.

ω_s = fréquence de coupure d'arrêt-bande.

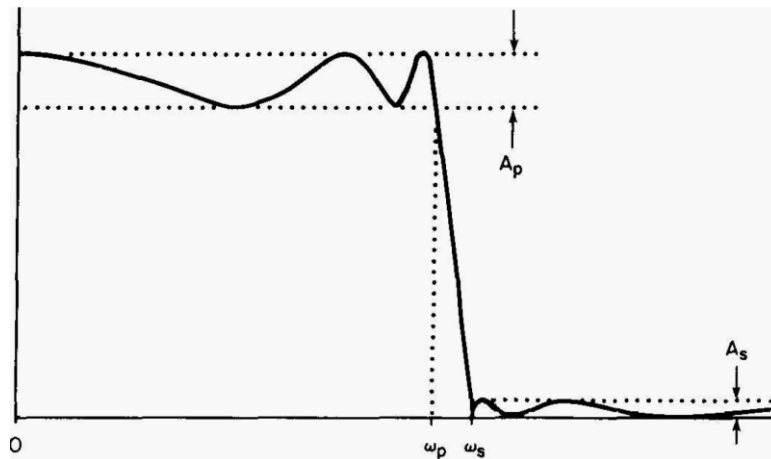


Figure 3.11. Réponse de fréquence affichant les paramètres utilisés pour spécifier un filtre elliptique [49]

Si l'amplitude maximale de la bande passe est l'unité, donc, A_p est la taille des ondulations de bande-passe, et A_s est la taille des ondulations d'arrêt-bande. Chacun des quatre paramètres de filtre peut être spécifié indépendamment, avec le cinquième (ordre n) puis être fixé par la nature de la réponse du filtre elliptique. La stratégie de conception habituelle consiste à spécifier A_p , A_s , ω_p et ω_s en fonction des exigences de l'application prévue.

c.4. Filtre de Chebyshev Inverse

Inverse le filtre de Chebyshev, sont étroitement liés aux filtres de Chebyshev et sont notés pour avoir une réponse plate-bande de magnitude de bande, et une réponse d'équirépartition dans le stop-bande. La réponse Inverse Chebyshev est souvent appelée la réponse de type II

Chebyshev, pour des raisons qui deviendront claires car la réponse de fréquence est développée ci-dessous [53].

Réponse de fréquence :

La réponse de magnitude carrée des filtres de Chebyshev inverse est définie comme [54]:

$$|H(j\omega)|^2 = \frac{\varepsilon^2 T_n^2\left(\frac{\omega_{scale}}{\omega}\right)}{1 + \varepsilon^2 T_n^2\left(\frac{\omega_{scale}}{\omega}\right)} \quad (11.)$$

La fréquence d'échelle pour les filtres Chebyshev inverse est définie comme égale à la fréquence de bord d'arrêt-bande. La réponse de magnitude carrée peut être présentée sous la forme

$$|H(j\omega)|^2 = \frac{\varepsilon^2 T_n^2\left(\frac{\omega_s}{\omega}\right)}{1 + \varepsilon^2 T_n^2\left(\frac{\omega_s}{\omega}\right)} \quad (12.)$$

La méthode d'approximation inverse Chebyshev est étroitement liée à l'approximation de Chebyshev. La principale caractéristique de la méthode approximation inverse Chebyshev est une ondulation dans la bande d'arrêt que Figure 2.12 montre.

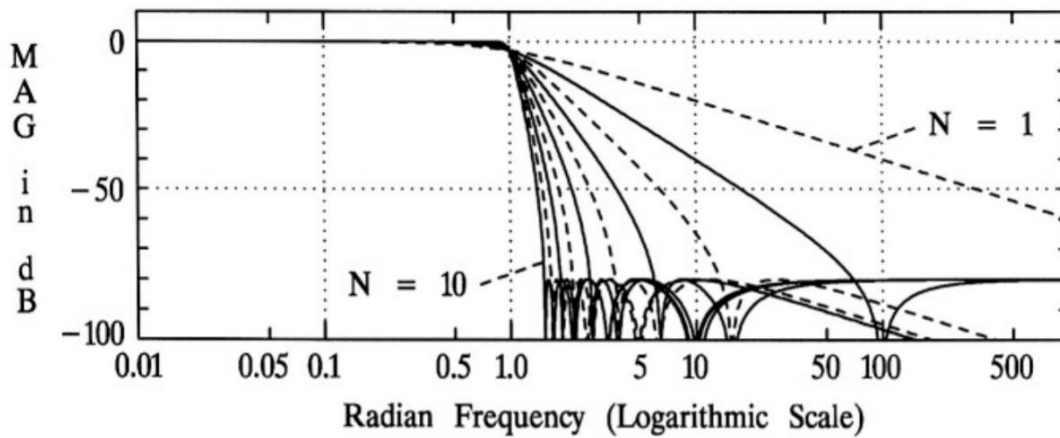


Figure 3.12. Réponse inverse de magnitude de filtre de Chebyshev pour différents ordres [53].

3.3.4 Méthodes d'évaluation

a. Test de classification binaire

Le test de classification binaire (BCT) est basé sur le calcul du vrai positif, du vrai négatif, du faux positif, du faux négatif. Dans cette partie, nous parlerons de chacun de ces paramètres et de leur impact sur notre étude.

Vrai positif: Le vrai positif d'une expérience (dans notre cas) est quand le mouvement existe et il est correctement identifié comme mouvement. Dans notre expérience, nous recherchons

le moment où le mouvement réel est égal à un et le mouvement détecté est également égal à un. Par conséquent, le vrai positif dans ce cas devrait également être égal à un, sinon, il devrait être égal à zéro.

Vrai négatif: Le vrai négatif d'une expérience (dans notre cas) est quand le mouvement n'existe pas, et il est correctement identifié comme non-mouvement. Dans notre expérience, nous recherchons le moment où le mouvement réel est égal à zéro et le mouvement détecté est également égal à zéro. Par conséquent, le vrai négatif, dans ce cas, devrait être égal à un, sinon, il devrait être égal à zéro.

Faux positif: Le faux positif d'une expérience (dans notre cas) est lorsque le mouvement n'existe pas et il est incorrectement identifié comme un mouvement. Dans notre expérience, nous recherchons quand le mouvement réel est égal à zéro mais le mouvement détecté est égal à un. Par conséquent, le faux positif dans ce cas doit être égal à un, sinon, il doit être égal à zéro.

Faux négatif: Le faux négatif d'une expérience (dans notre cas) est lorsque le mouvement existe et il est incorrectement identifié comme non-mouvement. Dans notre expérience, nous recherchons quand le mouvement réel est égal à un mais le mouvement détecté est égal à zéro. Par conséquent, le faux négatif, dans ce cas, il doit être égal à un, sinon, il doit être égal à zéro. Ce tableau (Tableau 2.5) représente les valeurs mentionnées précédemment et leur relation avec les expériences:

		Mouvement Vrai	
		1	0
Mouvement détecté	1	TP	FP
	0	FN	TN

Tableau 3.5. Test de classification binaire.

b. Mesures statistiques

Une fois que nous avons calculé les valeurs statistiques précédentes (TP, TF, FP et FN), nous pouvons facilement calculer la mesure statistique (sensibilité, spécificité et précision).

Sensibilité: la sensibilité dans notre cas se réfère à la méthode de détection du mouvement fœtal capacité à détecter correctement le mouvement fœtal et pas d'autres mouvements.

Mathématiquement, cela peut être exprimé [55] comme :

$$\text{Sensibilité} = \frac{\text{Vrai Positive}}{\text{vrai Positive} + \text{Fausse Negative}} \quad (13.)$$

Une méthode de détection à 100% de sensibilité n'aura pas de faux négatifs, ce qui signifie que chaque fois qu'il y a un mouvement, il est sûrement détecté. Cependant, la sensibilité par définition ne prend pas en compte les faux positifs ; à partir de cela, nous pouvons dire que la méthode de détection peut détecter de faux positifs et ont toujours une sensibilité de 100%.

Spécificité : la spécificité dans notre cas se rapporte à la capacité de la méthode de détection de mouvement fœtal de rejeter correctement le mouvement non fœtal. Mathématiquement, cela peut être écrit [55] comme :

$$\text{Spécificité} = \frac{\text{Vrai Negative}}{\text{Vrai Negative} + \text{Fausse Positive}} \quad (14.)$$

Une méthode de détection avec 100% de spécificité n'aura pas de faux positifs, ce qui signifie que chaque fois qu'il n'y a pas de mouvement non fœtal détecté comme un mouvement fœtal. Cependant, la spécificité par définition ne tient pas compte des faux négatifs, à partir de cela nous pouvons dire que la méthode de détection pourrait ignorer certains mouvements fœtaux, mais ceux détectés sont en effet des mouvements fœtaux, et si nous prenons cela en compte, nous nous retrouverons avec certains ou plusieurs mouvements non détectés à tous. En outre, nous pouvons dire que notre méthode est probablement d'avoir faux négatif et ont encore une spécificité de 100%.

Précision: l'exactitude a deux définitions, plus généralement, il s'agit d'une description des erreurs systématiques, une mesure de biais statistique; faible précision provoque une différence entre un résultat et une valeur « rai ». L'ISO appelle cela la véracité. Alternativement, l'ISO définit l'exactitude comme décrivant une combinaison des deux types d'erreur observationnelle ci-dessus (aléatoire et systématique), de sorte que la haute précision exige à la fois une grande précision et une grande véracité [56].

Dans les termes les plus simples, compte tenu d'un ensemble de points de données à partir de mesures répétées de la même quantité, l'ensemble peut être dit pour être précis si les valeurs sont proches les unes des autres, tandis que l'ensemble peut être considéré comme exact si leur moyenne est proche de la valeur réelle de la quantité mesurée. Dans le premier, la définition la plus commune ci-dessus, les deux concepts sont indépendants l'un de l'autre, de sorte qu'un ensemble particulier de données peut être dit soit précis, ou précis, ou les deux, ou ni l'un ni l'autre. Mathématiquement, cela peut être écrit comme:

$$\text{Précision} = \frac{\text{Vrai Positive} + \text{Vrai Negative}}{\text{Vrai Positive} + \text{Vrai Negative} + \text{Fausse Positive} + \text{Fausse Negative}} \quad (15.)$$

3.4 Système d'acquisition

L'acquisition de signal de mouvements fœtaux est faite à l'aide de deux capteurs d'accélération attachés (MPU-6050) au Arduino uno, qui agit comme le maître dans le protocole de communication I2C, tandis que les capteurs seront des esclaves agissantes. Nous avons utilisé ces capteurs parce qu'ils ont deux adresses X68 et X69 sur lesquelles ils pourraient être programmés. Par conséquent, nous ne pouvons pas traiter plus de deux capteurs d'accéléromètre dans la même connexion I2C. Figure 3.1 montre le diagramme de bloc essentiel du système. Dans l'examen de la littérature, il y avait un article [22] qui mentionnait que seulement un accéléromètre suffit. Mais, nous sommes allés un peu plus loin et a ajouté le 2ème pour se débarrasser des erreurs expérimentales autant que possible.

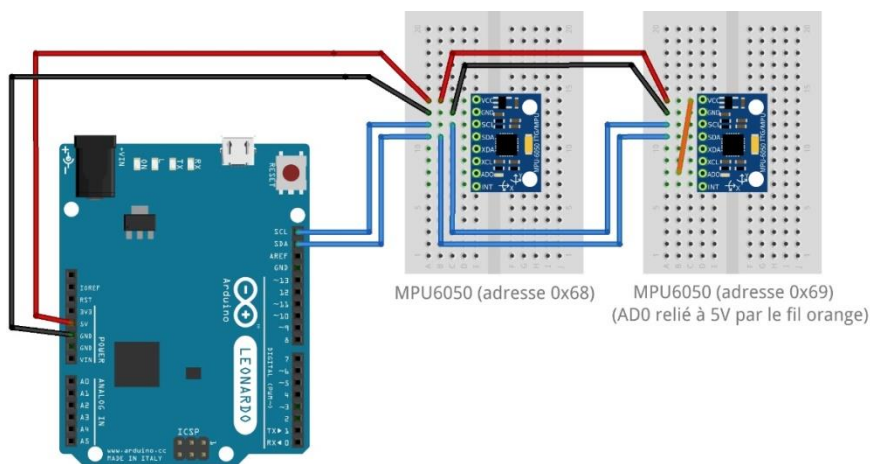


Figure 3.13. Branchement de système de signal d'acquisition.

3.4.1 Logiciel d'acquisition

Il n'y a pas de manque d'applications adaptées à l'épargne d'un fichier CSV à partir d'un port ouvert, qui peut ensuite être importé dans Microsoft Excel. Par exemple, GoBetwino, Processing et Eltima Port Logger sont des stratégies utiles et applicables.

Toutefois, il est possible d'avoir Visual Basic (sous Excel) écouter le port de série et de sortir les données directement dans Excel sans avoir besoin d'un autre programme ou plugin à charger dans les versions Mac et PC.

Le premier ordre du jour est de découvrir le numéro de port ouvert, à travers l'Arduino IDE. Cela se fait en chargeant l'Arduino IDE, en connectant votre appareil (par exemple Arduino Uno), puis en cliquant sur Outils --> Port.

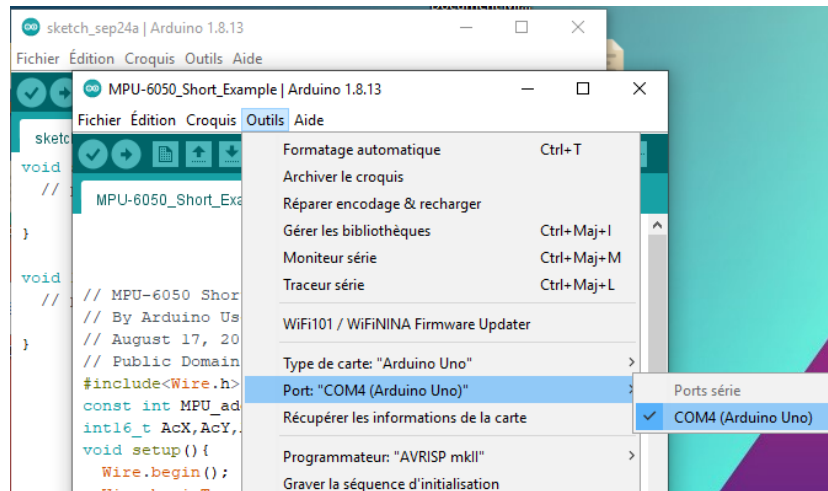


Figure 3.14. Numéro de porte (COM) sur Arduino UNO.

a. Visual Basic Macros pour excel dans windows

Ensuite, vous devrez créer une macro dans Visual Basic pour lire le port. Dans Windows, la routine de base dépouillée ressemble à ceci :

```
Sub ReadCommPC()
    Dim COMfile As Integer
    Dim COMstring As Variant
    Dim baudrate As Long
    Dim record_cat As Variant
    Dim record As String * 1
    Dim emptyRecord As String * 1

    COMport = "COM4" 'Enter the COM port here.
    baudrate = 9600 'Enter the baud rate here.

    'Open COM port with baud rate, no parity, 8 data bits and 1 stop bit
    COMfile = FreeFile
    COMstring = COMport & ":" & baudrate & ",N,8,1"

    Open COMstring For Random As #COMfile Len = 1
    record = ""
    record_cat = ""

    Do
        DoEvents 'Don't lock up excel while waiting
        Get #COMfile, , record 'data is read in 1 character at a time
        If record <> "," And Asc(record) <> 13 And Asc(record) <> 10 And record <> emptyRecord Then
            record_cat = record_cat & record
        End If
        If Asc(record) = 13 Then ' if carriage return
            MsgBox (Trim(record_cat))
            record_cat = ""
            record = ""
        ElseIf record = "," Then ' if comma
            MsgBox (Trim(record_cat))
            record_cat = ""
        End If
        Sleep 20
    Loop
    Close #COMfile
End Sub
```

Visual Basic se connecte directement au port COM. Cette version de la macro imprime uniquement un message des données, plutôt que la feuille de calcul. Toutefois, il est plus facile d'aller à la feuille de calcul à partir de ce format avec des commandes comme :

```
ActiveCell.Offset(1,0).Valeur = record_cat
```

3.4.2 Méthodologie acquisition

Afin de tester l'algorithme, nous avons fait trois expériences distinctes, et dans chacune de ces expériences nous avons déplacé l'accéléromètre avec une amplitude arbitraire. Afin de créer l'accélération et nous avons également essayé d'imiter les mouvements respiratoires de la mère ainsi que les coups de pied et de roulement. Fondamentalement, tout ce que nous avons est signaux de vibration, qui sont les signaux d'accélération détectés par l'accéléromètre. Le point de cette expérience est que nous ne savons pas les signaux acquis, tout comme si nous avons couru l'expérience sur une femme enceinte réelle, si nous secouer l'accéléromètre pour générer les signaux d'accélération. Pendant ce temps, nous sommes en appuyant sur le Button0 (bouton poussoir) pour indiquer qu'il y a des mouvements réels.

Le pressurage de bouton0 déterminera s'il y a un vrai mouvement en cours, ou il juste un fabriqué un. Nous avons donc une fois que nous avons l'intention de déplacer l'accéléromètre que nous appuyons sur la bouton0; une fois que nous faisons cela, le signal sortant de la Button0 fonctionnera comme un standard d'or pour l'existence de la non-existence du mouvement fœtal. Par exemple dans les papiers [17] [15] [14] [16], les auteurs ont utilisé le masque d'échographie comme norme d'or, malheureusement, nous n'avons pas eu accès à l'hôpital pour tester sur une femme enceinte, ni les outils Ultrasound, donc nous utilisons la bouton0 pour construire un masque détermine alors la véritable existence du mouvement ou de le nier. Si le signal d'accélération montre qu'il y a un mouvement alors que le bouton n'est pas pressé, cela signifie que ce mouvement n'est pas un véritable mouvement fœtal c'est juste un bruit aléatoire. Nous pouvons le déterminer simplement en regardant vers les deux signaux; cependant, l'algorithme de détection de seuil détectera ce mouvement comme un mouvement foetal réel; c'est là que nous testons les performances de la méthode que nous utilisons. D'autre part, si les signaux d'accélération indiquaient un mouvement et que le bouton0 unique prenait la valeur d'un (vrai) maintenant nous pouvons dire que ce mouvement était vraiment un moment foetal et à partir de cela nous pouvons aller à la pensée de calculer le vrai négatif, vrai positif, faux positif, faux négatif, ainsi que la sensibilité, la précision et la spécificité.

Nous avons généré trois signaux d'accélération avec trois masques binaires à partir du bouton0, et appliquer différents filtres IIR à ces expériences un par un.

3.4.3 Signaux acquis

Dans ces expériences, nous allons tracer les trois signaux sortant de l'accéléromètre qui sont l'accélération de l'axe X, l'accélération de l'axe Y et l'accélération de l'axe Z et nous le ferons pour les trois expériences.

Expérience 1 :

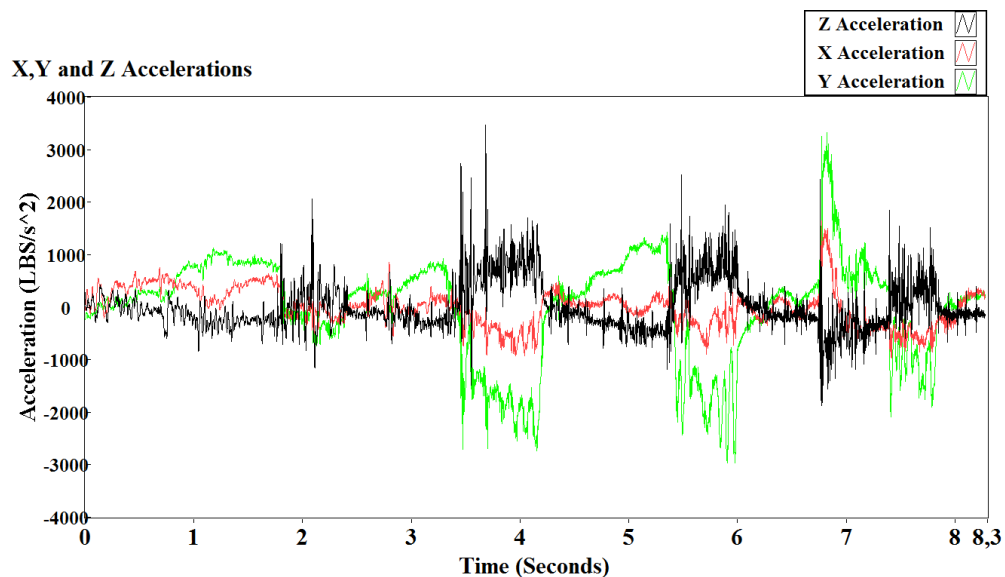


Figure 3.15. Signaux d'accélération X, Y et Z pour la première expérience.

Expérience 2 :

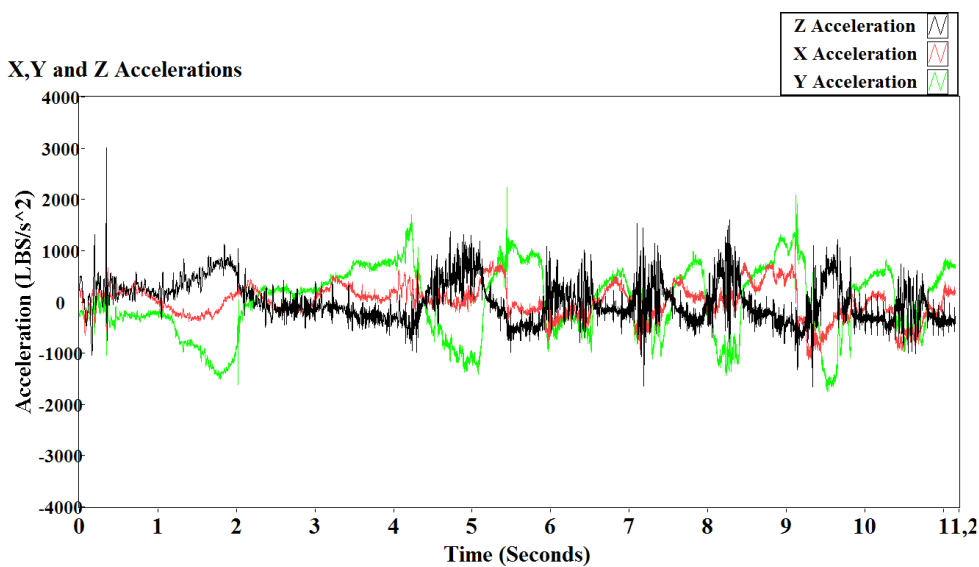


Figure 3.16. Signaux d'accélération X, Y et Z pour la deuxième expérience.

Expérience 3 :

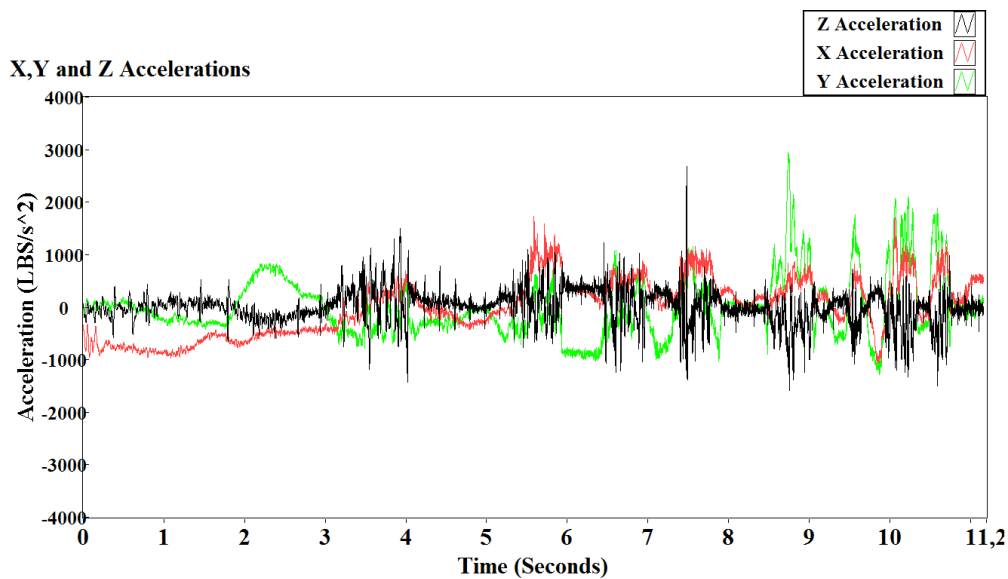


Figure 3.17. Signaux d'accélération X, Y et Z pour la troisième expérience.

3.5 Méthodologie de détection de mouvement fetale

3.5.1 Retrait signal

Les signaux prélevés de l'Arduino par le port "com" arrivent sur un ordinateur personnel exactement sur le fichier Excel qui est développée par un code d'acquisition (macron) et stockés comme un fichier tableur. Pour obtenir ces signaux en arrière, nous allons copier la colonne B (accélération Z prise comme un exemple) et ensuite passer dans un fichier texte (Figure 3.6), de sorte qu'il serait plus tard plus facile à acquérir pour notre algorithme de traitement.

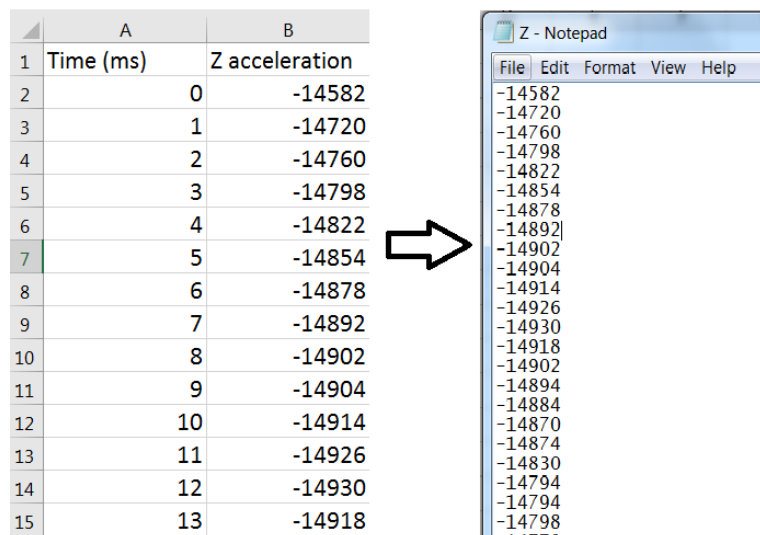


Figure 3.18. Feuille de calcul de données et fichier texte.

Après avoir fait les étapes précédentes, nous pouvons ouvrir LabVIEW et commencer à construire le code de traitement. D'abord, nous rattachons les fichiers texte. Pour ce faire, nous avons besoin pour obtenir la lecture du fichier de mesure VI sous la palette d'entrée express (Figure 3.7).

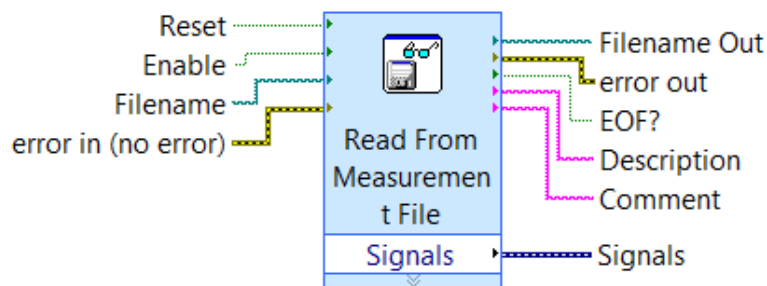


Figure 3.19. Lecture du fichier de mesure VI.

Une fois que nous double-cliquons sur ce VI, nous obtenons une boîte de dialogue pour définir les préférences et les paramètres multiples (Figure 3.8). Le paramètre important dont nous avons besoin est de définir l'emplacement du fichier que nous voulons acquérir, dans la section Format de fichier ; nous devons vérifier le texte (LVM), qui est pour LabVIEW Mesures. Après cela, nous avons vérifié la zone du fichier de texte générique de lecture. Enfin, nous cliquons sur Lire le fichier maintenant pour s'assurer que le fichier est lisible dans son format. Une fois que nous faisons cela, nous pouvons voir nos valeurs de mesure apparaissent sur la petite table de données d'échantillon juste comme Figure 3.8 montre.

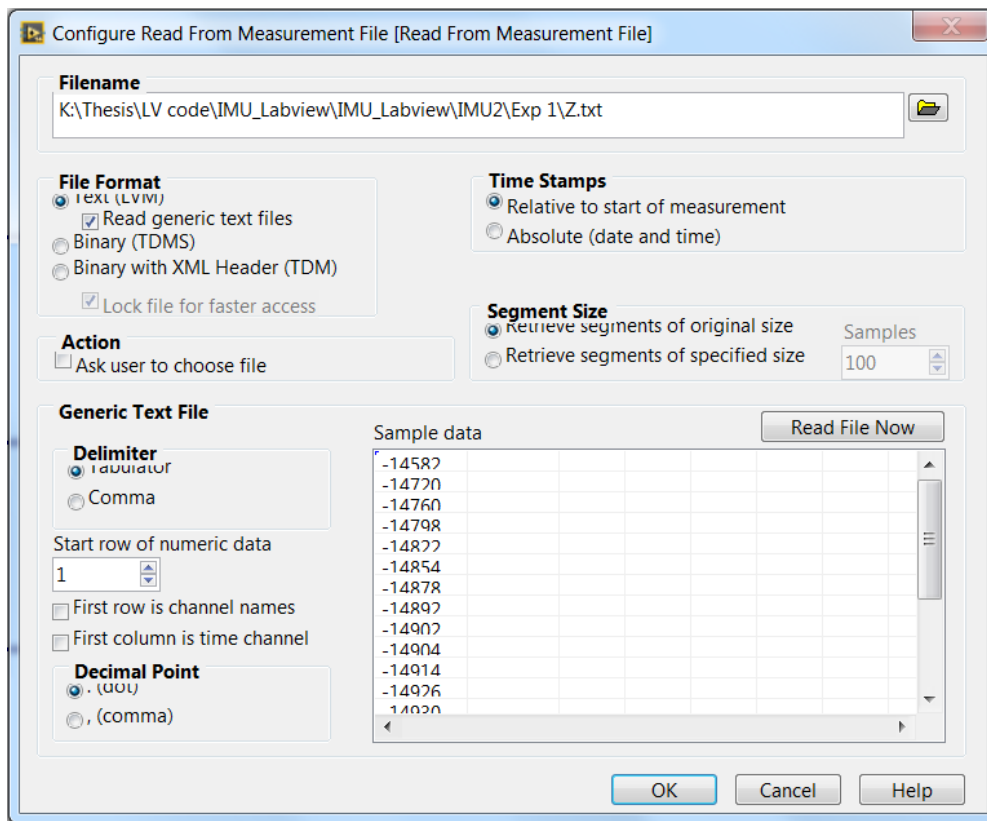


Figure 3.20. Lire à partir de la zone Dialogue du fichier de mesure VI.

Nous devons faire ces dernières étapes à tous les signaux que nous avons à la fois (X, Y, accélérations Z, Button0 Mask). Nous nous retrouvons avec quatre Lire à partir du fichier de mesure; chacun d'eux acquiert un signal spécifique.

L'étape suivante sera d'extraire la moyenne statistique des signaux d'accélération pour obtenir des signaux à zéro moyenne. Pour ce faire, nous devons obtenir la Statistique VI sous la palette d'Analyse Signal Express (Figure 3.9)

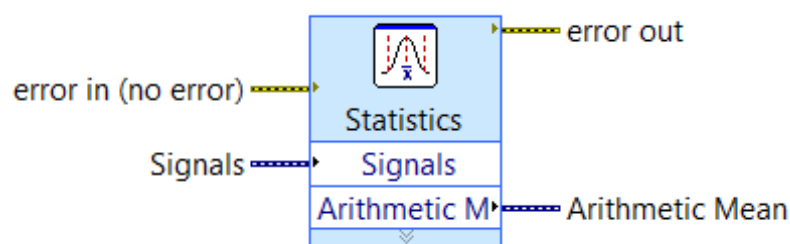


Figure 3.21. Statistiques Express VI

Une fois que nous double-cliquons sur ce VI, une boîte de dialogue apparaîtra (Figure 3.10), et il contient des mesures statistiques différentes comme le Mean Arithmétique, Médiane, Root Mean Square (RMS), Variance, et Kurtosis. La moyenne Arithmétique est la mesure qui nous intéresse, donc nous avons seulement besoin de cocher cette case et cliquez sur OK.

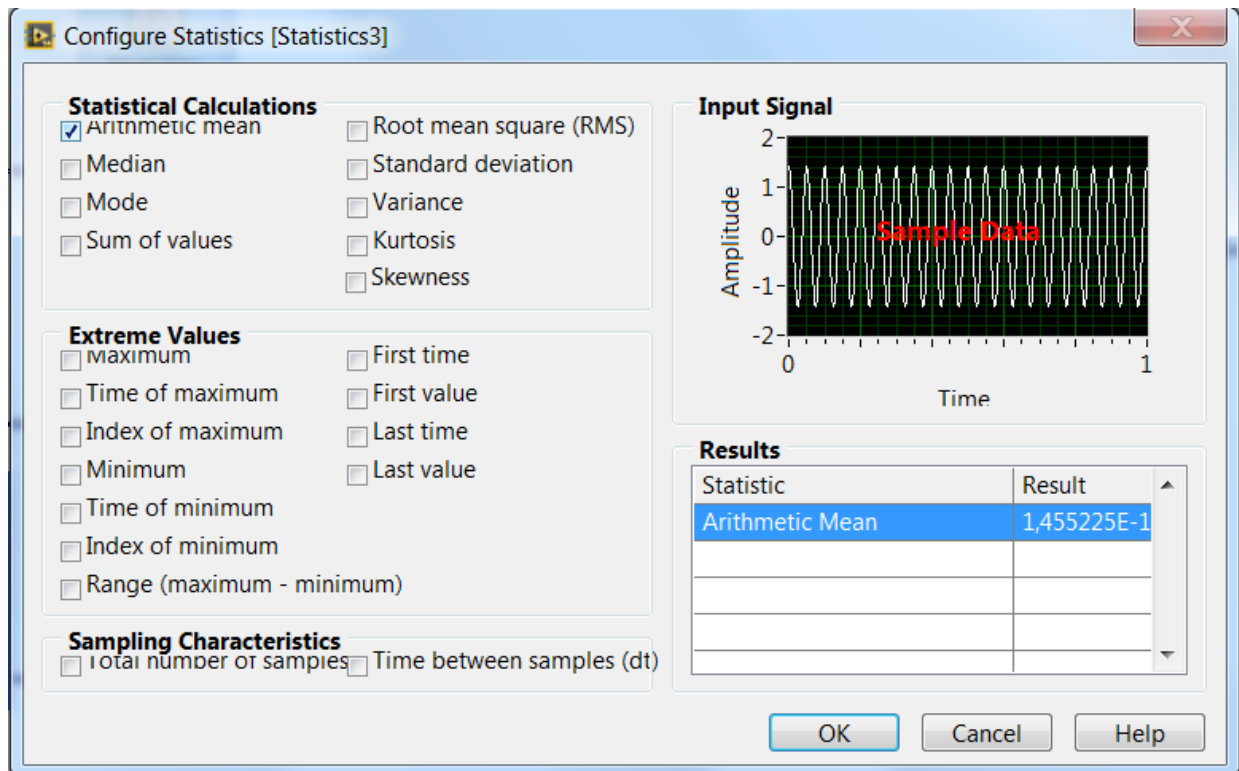


Figure 3.22. Statistiques Express VI Boîte de dialogue.

3.5.2 Racine -moyen-carré

Pour calculer le carré moyen racine de nos signaux, nous devons prendre la sortie du bloc de soustraction et y ajouter un bloc Convertir à partir de données dynamiques, afin que nous puissions convertir en un tableau 1D du canal scalaire unique après que nous le passions. Un bloc carré pour calculer le carré de ce signal. Nous devrions faire cela aux trois signaux acquis (accélération X, Y et Z). Après cela, nous additionnons ces trois signaux à l'aide d'un bloc arithmétique composé, puis le passons à travers un bloc de racine carrée, la sortie de ce bloc sera le carré moyen de ce signal. L'ensemble du processus peut être vu dans la Figure 3.11.

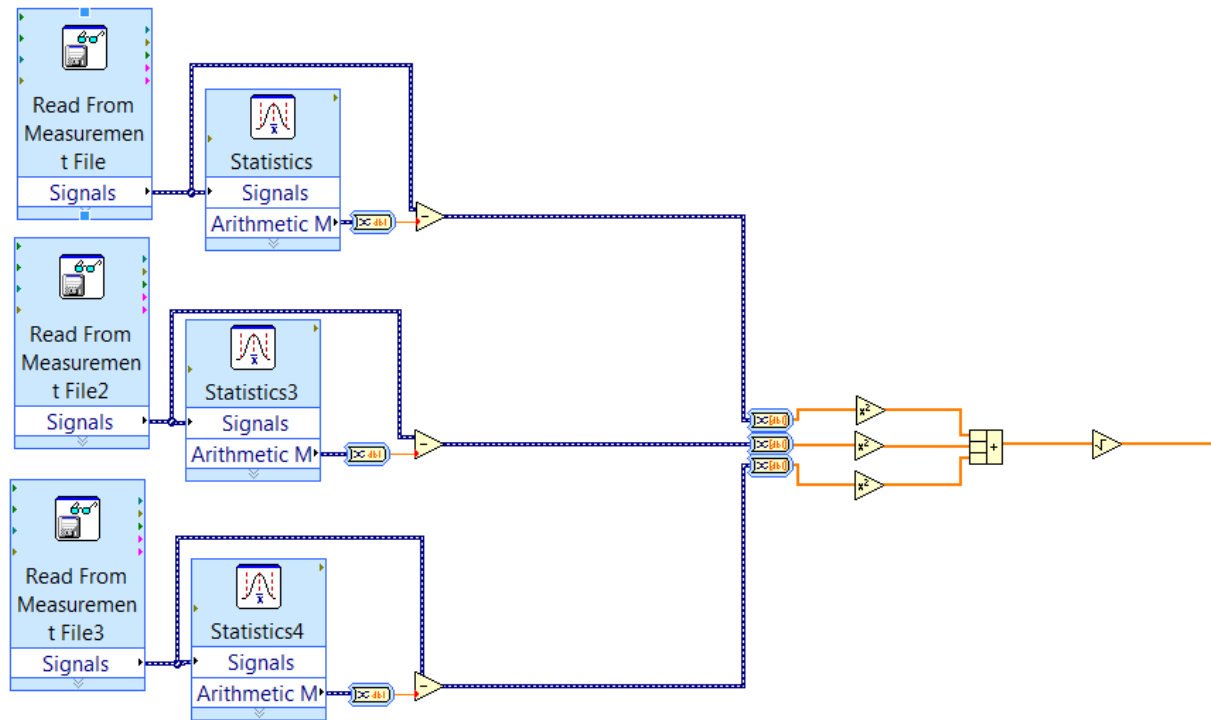


Figure 3.23. Diagramme de bloc de calcul RMS.

Le résultat de cette partie du code peut être vu dans les chiffres suivants, que nous avons tracé le Root-Mean-Squared pour l'accélération X, Y et Z, et pour les trois Expériences et ils peuvent être vus dans les chiffres suivants.

Expérience 1:

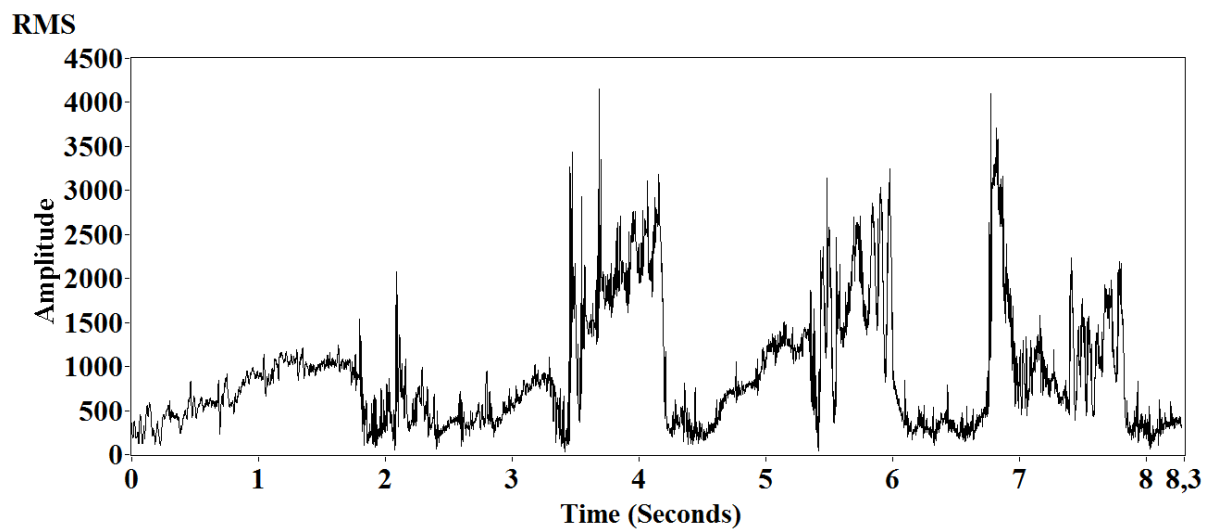


Figure 3.24. Racine-moyenne-carré des signaux d'accélération de la première expérience.

Expérience 2 :

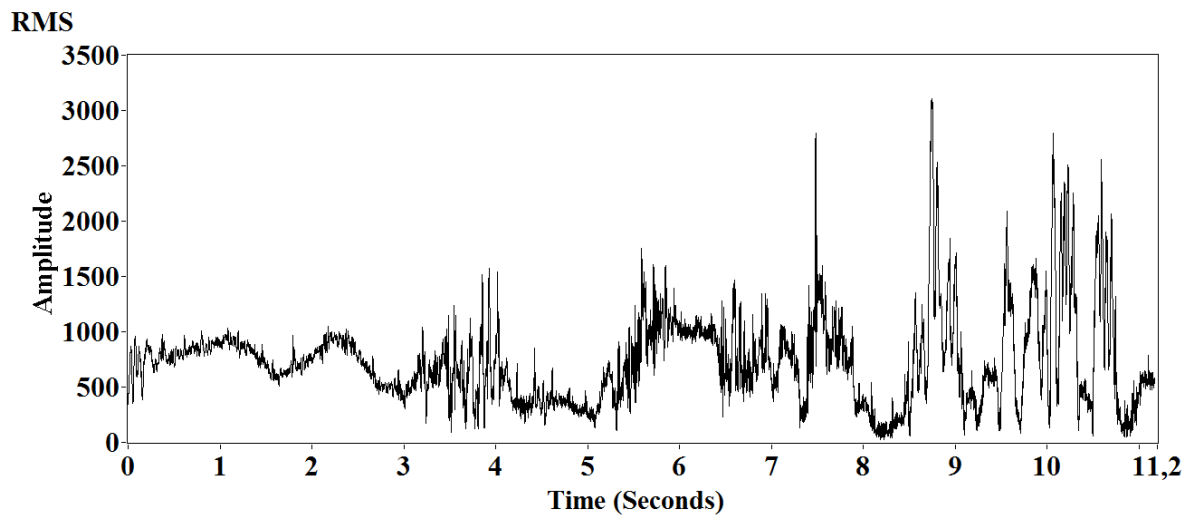


Figure 3.25. Root-Mean-Square des signaux d'accélération de la troisième expérience.

Expérience 3 :

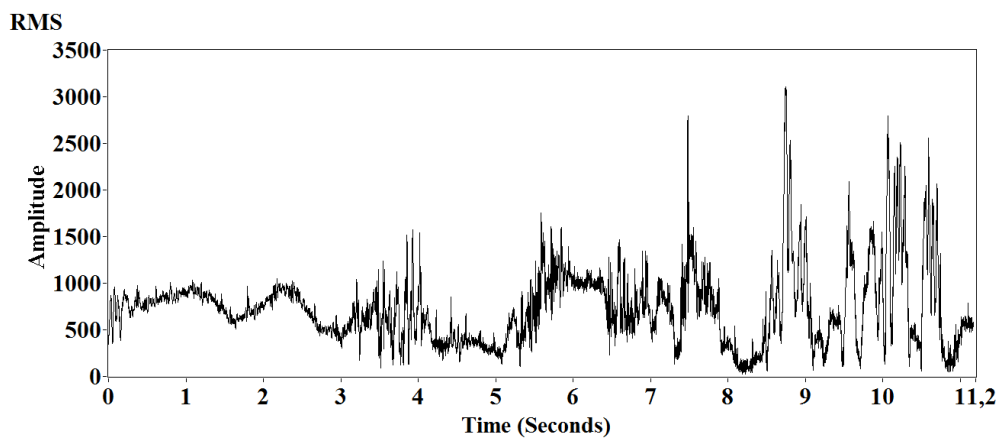


Figure 3.26. Root-Mean-Square des signaux d'accélération de la troisième expérience.

Une fois que nous avons la racine-moyen-carré pour nos signaux, nous pouvons effectuer la Transformation de Fourier rapide pour obtenir le spectre de magnitude pour notre signal ainsi que le spectre de la phase. Tout ce qui nous intéresse, c'est le spectre de magnitude de notre signal et les fréquences actuelles en elle.

Pour effectuer la Transformation de Fourier Rapide, nous devons obtenir le Spectral Mesurment Express VI (Figure 3.15) trouvé sous la palette d'Analyse Signal Express.

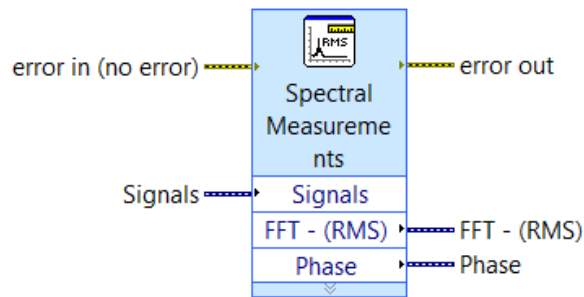


Figure 3.27. Spectral Measurement Express VI

Une fois que nous double-cliquons sur cet express, nous obtenons une boîte de dialogue (Figure 3.16) qui contient de multiples paramètres, comme la magnitude (RMS), magnitude (Peak), de ce signal particulier ainsi que le spectre de puissance et la densité spectrale de puissance. La seule chose qui nous intéresse est la magnitude (Peak). Pour la technique de windowing, nous allons laisser la fenêtre par défaut (Hanning).

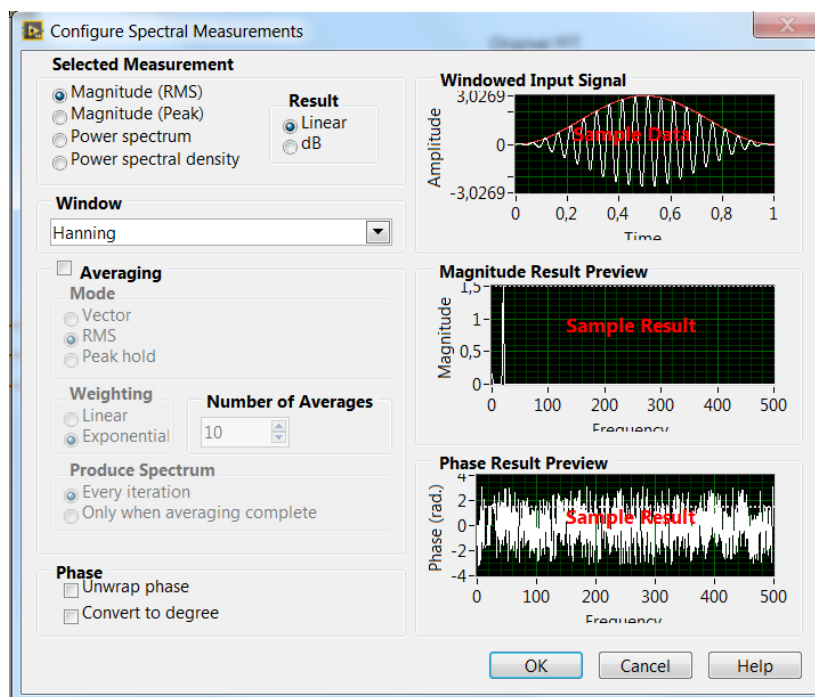


Figure 3.28. Boîte de dialogue Spectral Measurement Express VI

Les chiffres suivants montrent la réaction de magnitude pour les trois expériences menées.

Expérience 1 :

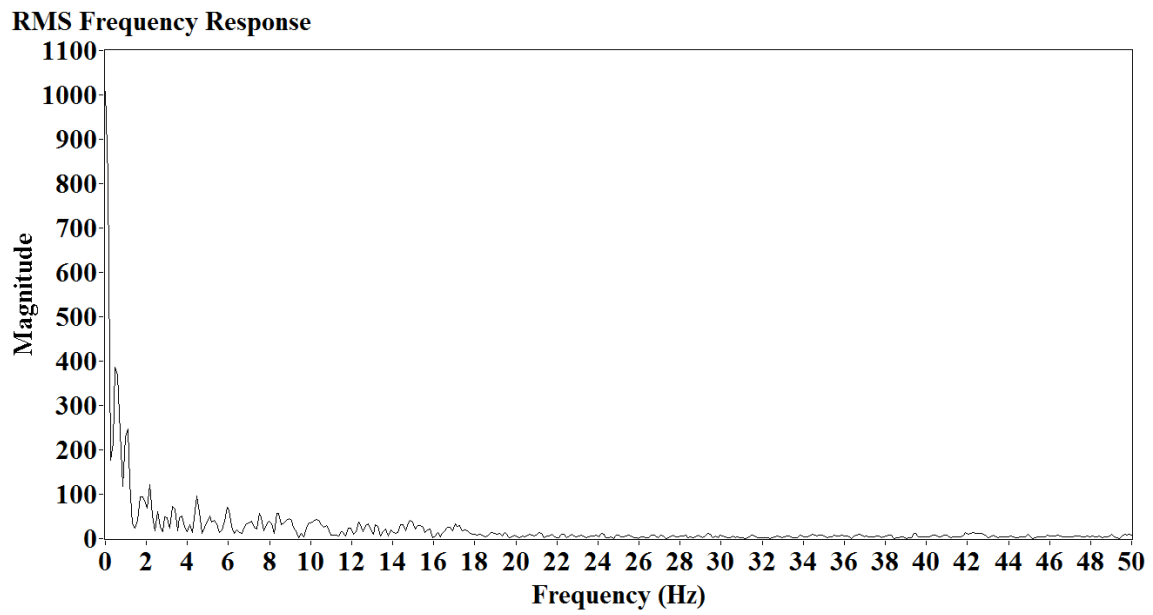


Figure 3.29. Réponse de magnitude du RMS pour la première expérience.

Expérience2 :

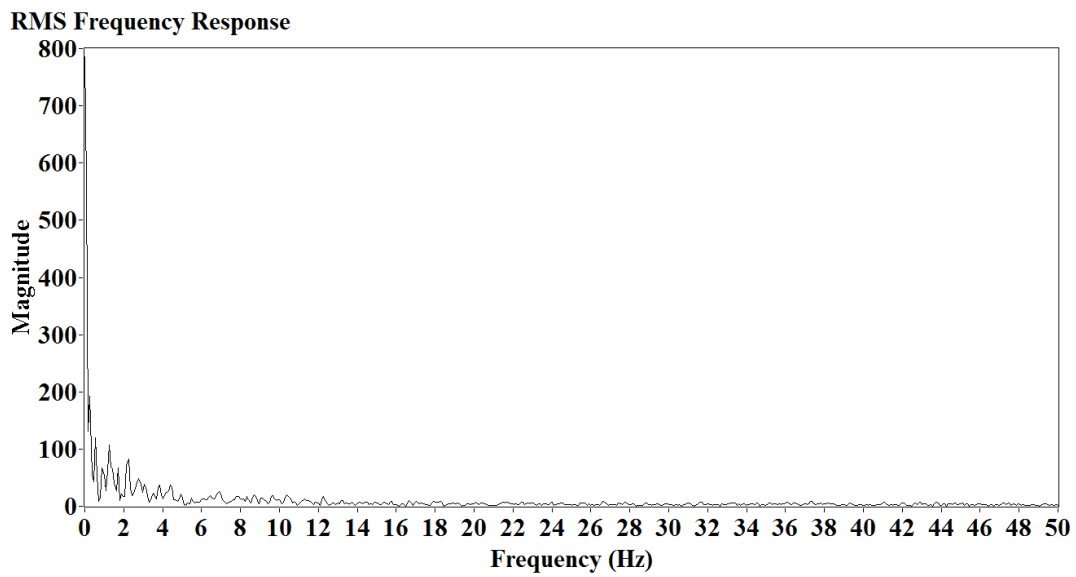


Figure 3.30. Réponse de magnitude du RMS pour la deuxième expérience.

Expérience 3 :

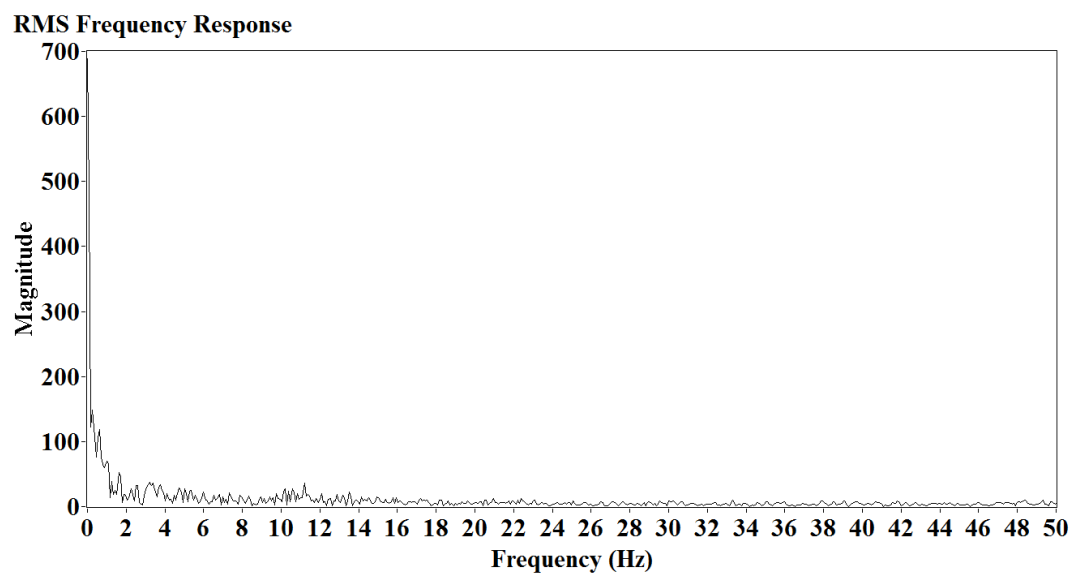


Figure 3.31. Réponse de magnitude du RMS pour la troisième expérience.

3.6 Conception du filtre

Dans cette partie, nous allons concevoir quatre filtres IIR, et nous leur donnerons le même ordre et les mêmes fréquences de coupure. Ensuite, nous allons tracer leur réponse de fréquence pour voir les résultats des différentes conceptions. L'ordre du filtre est choisi pour être 12 puisque nous ne voulons pas augmenter l'ordre du filtre jusqu'à ce qu'il perde sa stabilité. Les fréquences de coupure sont de 2Hz à 20Hz ($[2; 20]$ Hz) parce que les mouvements fœtaux existent dans cet intervalle [14].

a. Conception du filtre Butterworth

Dans cette partie, nous allons concevoir le filtre Butterworth, afin de le faire, nous allons ouvrir le Butterworth Design VI sous la palette De conception de filtre numérique et il est montré dans Figure 3.20. Fondamentalement, ce VI crée un numérique Butterworth réponse impulsion infinie (IIR) filtre pour nous.

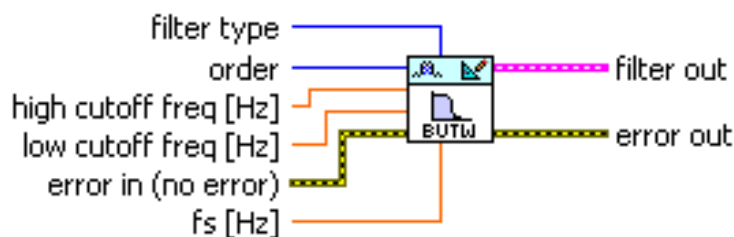


Figure 3.32. Butterworth Design VI

Après nous avons que VI, nous avons besoin de définir les entrées ou les contrôles pour que VI ainsi que les sorties ou les indicateurs. Pour les contrôles, nous allons créer le type de filtre, la fréquence de découpe de conception et la fréquence de coupure faible à la fois dans le hertz.

Ensuite, nous allons attacher la sortie de filtre de la conception VI Butterworth à plusieurs blocs afin que nous puissions obtenir plusieurs pièces d'information et de mesures à partir de lui. Ces blocs se trouvent sous la palette d'analyse de filtre : le premier bloc est la réponse de fréquence VI afin que nous puissions tracer la réponse de fréquence de ce filtre particulier, nous pouvons tracer la réponse de magnitude et la réponse de phase, et nous l'avons mis en place à Linear-Spaced au lieu de Logarithmic-Spaced ou de fréquence vectorielle. Le second bloc est le pôle Zéro VI, et qui nous aide à tracer les pôles et les zéros de ce filtre dans le Z-Plane. Le troisième bloc est l'Impulse Réponse VI, et fondamentalement qui nous a aidé à tracer la réponse impulsion de ce filtre particulier. Le quatrième est le Step Réponse VI et il nous aide à tracer la réponse de l'étape du filtre Butterworth. Après cela, nous avons besoin d'ajouter pour chacun des quatre blocs précédents indicateurs graphique, de sorte que nous finirons avec cinq différents graphiques chacun a son but.

Une fois que nous avons fini d'attacher les blocs, alors nous pouvons mettre tout cela ensemble à l'intérieur d'une boucle while et ajouter la condition de la boucle ainsi. Le résultat final peut être vu dans Figure 3.21.

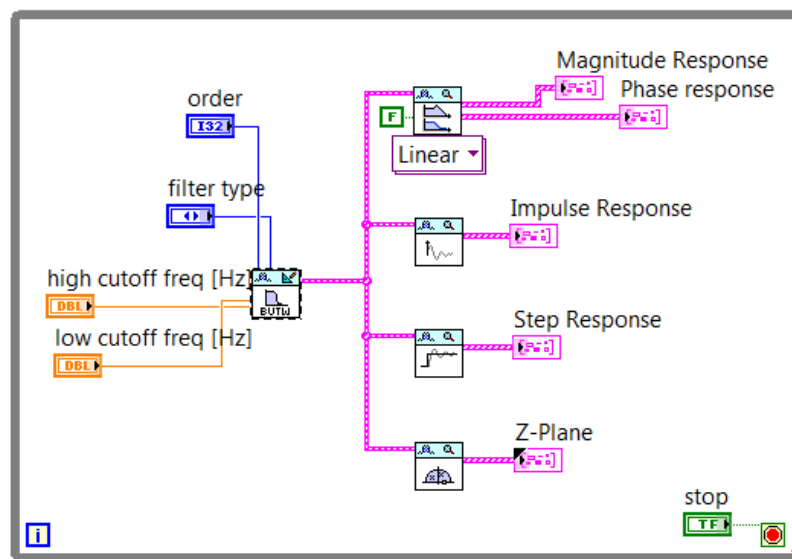


Figure 3.33. Diagramme de bloc de conception de filtre

Puisque nous sommes intéressés par une gamme de fréquences ou une bande de fréquences, alors nous pouvons concevoir un filtre passe-bande. Nous allons garder les fréquences que nous voulons et de se débarrasser des autres fréquences en rendant l'ampleur de ces fréquences zéro. Donc nous allons concevoir un filtre passe-passe avec un faible coupure de 2 Hz et une fréquence de coupure élevée de 20 Hz.

Nous remarquons que la conception de filtre Butterworth n'est pas efficace dans le département de transition d'aller de la 1 à 0 magnitude comme le chiffre de réponse de magnitude montre. Les chiffres suivants montrent la réponse de magnitude, la réponse d'impulsion, la réponse d'étape ainsi que les pôles et le graphe zéros sur le Z-plan.

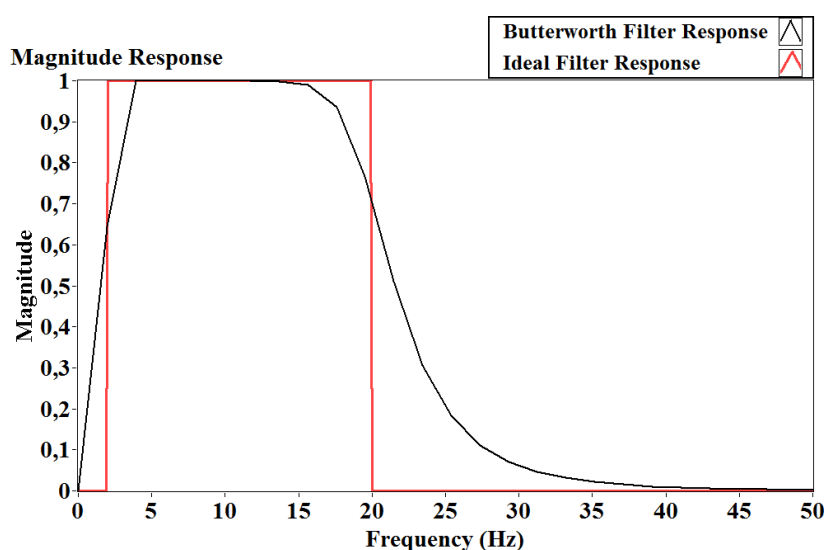


Figure 3.34. Réponse de l'ampleur de Butterworth

Comme nous pouvons le voir à partir de la réponse de magnitude, les résultats ont été les suivants: un col-bande fréquences de débit variant comme suit [4; 14] Hz et Stop-Band Fréquences allant comme suit [0; 50] Hz. La transition de la passe-bande à la bande d'arrêt est très douce et lisse, mais nous voulons qu'il soit aussi proche de la transition parfaite que possible. Bien que, nous pouvons remarquer que la conception de filtre Butterworth a une bonne transition de la bande d'arrêt à la passe-bande, et nous pouvons conclure que les filtres passe-haut Butterworth peut être très efficace.

Impulse Response

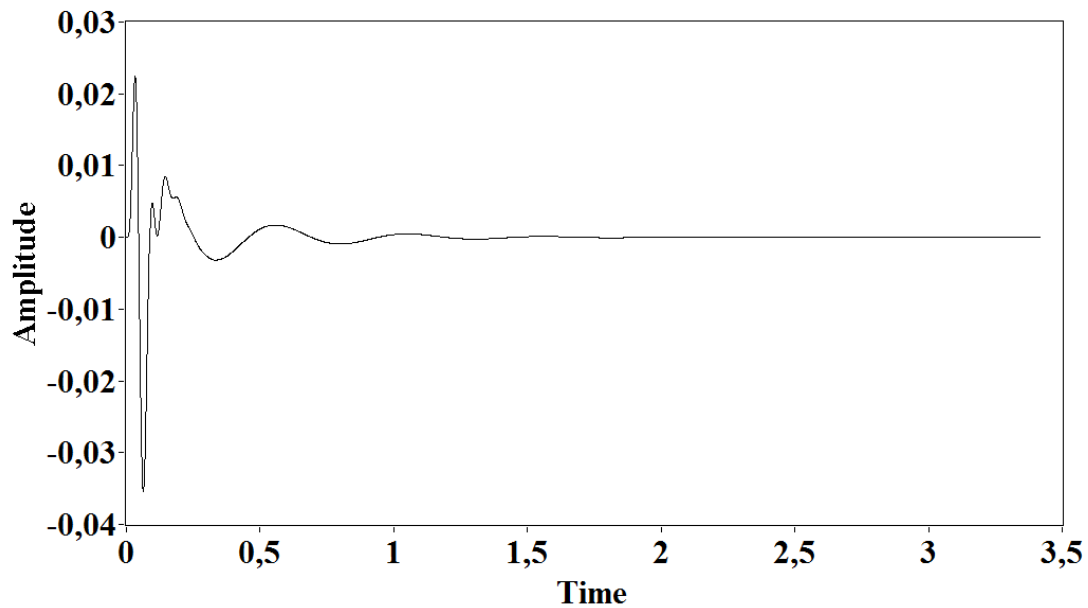


Figure 3.35. Réponse de Butterworth Impulse.

Step Response

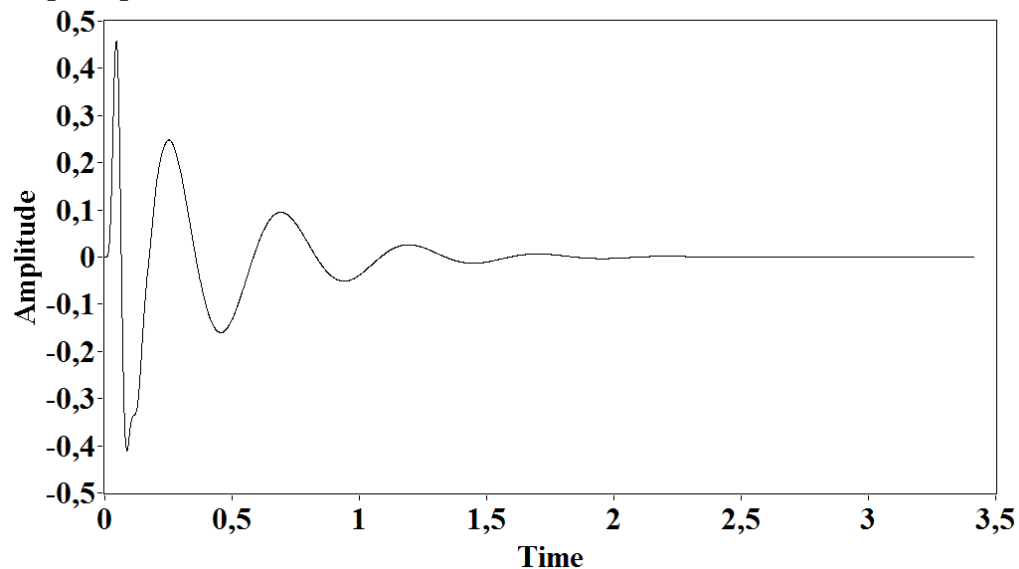


Figure 3.36. Réponse de Butterworth Step.

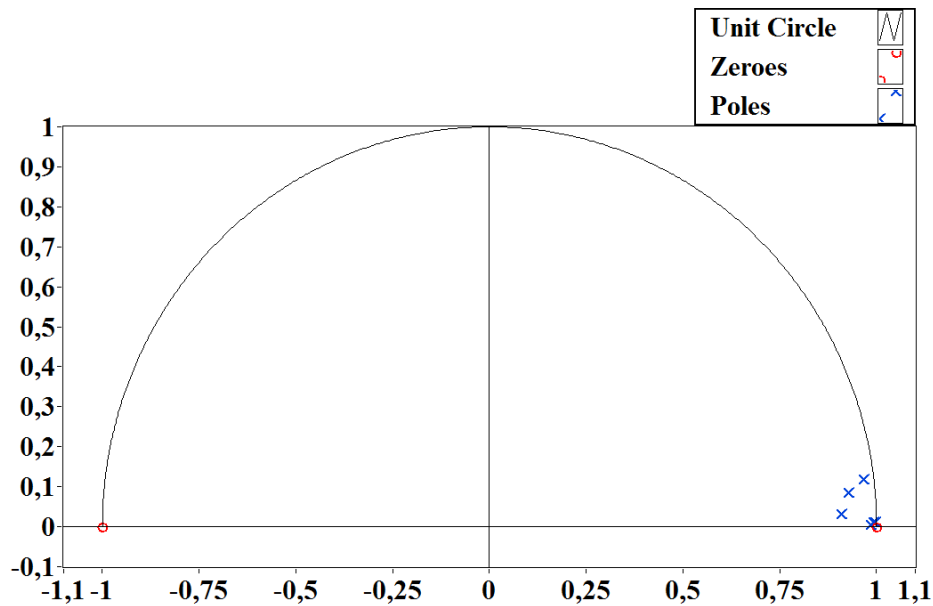


Figure 3.37. Placement de pôle de Butterworth.

b. Conception de filtres Chebyshev

Dans cette partie, nous allons concevoir le filtre Chebyshev tout comme nous l'avons fait pour la conception précédente du filtre. Pour ce faire, nous allons ouvrir le Chebyshev Design VI sous la même palette, et il est montré dans Figure 3.26. Fondamentalement, que VI crée une réponse numérique d'impulsion infinie de Chebyshev (IIR) pour nous.

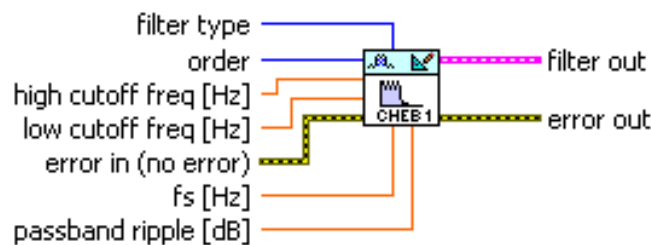


Figure 3.38. Chebyshev Design VI

Et après nous avons que VI, nous avons besoin de définir les entrées ou les contrôles pour que VI ainsi que les sorties ou les indicateurs. Pour les contrôles, nous allons créer le type de filtre, l'ordre de conception, haute fréquence de coupure et basse fréquence de coupure à la fois dans le hertz juste comme nous l'avons fait avant, et nous n'allons pas modifier l'ondulation du pass-band parce que nous n'avons pas accès à ce paramètre dans le Filtre Express VI, donc nous allons le laisser sur la valeur par défaut de 1dB.

Les chiffres suivants montrent la réponse de magnitude, la réponse d'impulsion, la réponse d'étape ainsi que les pôles et le graphe zéros sur le Z-plan

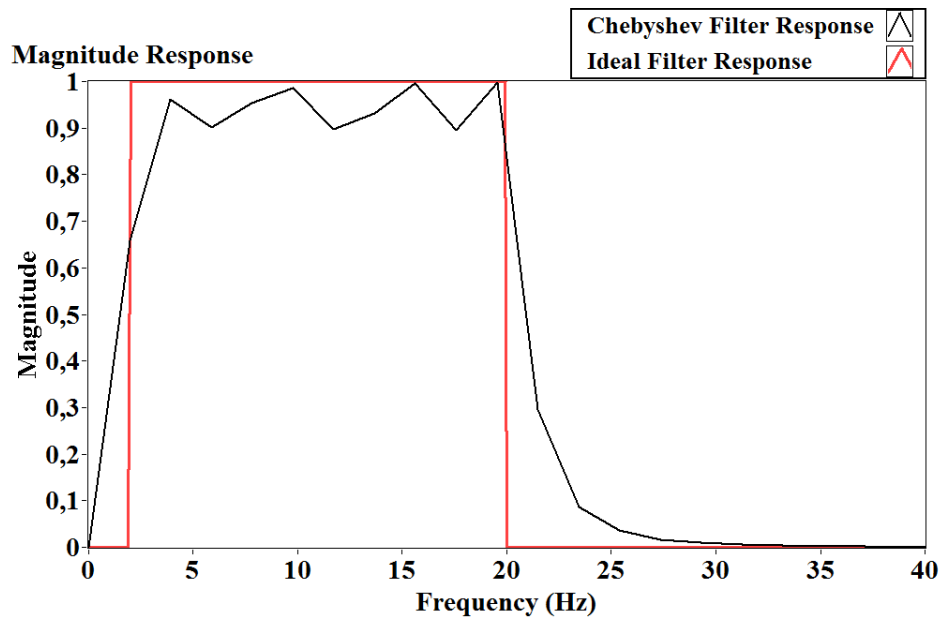


Figure 3.39. Réponse de magnitude de Chebyshev.

Comme nous pouvons le voir à partir de la réponse de magnitude, les résultats ont été les suivants: un col-bande fréquences de débit variant comme suit [4; 20] Hz et Stop-Band Fréquences allant comme suit [0; 33] Hz. La transition entre la bande passe et la bande d'arrêt est plus nette que celle du filtre Butterworth. Cependant, l'utilisation d'un filtre de Chebyshev est le Pass-band Ripples, puisque les ondulations dans le pass-band sont typiquement un effet secondaire indésirables de la production d'un filtre de haute commande qui a un raide roll-off si les ondulations sont trop grandes, nous les compter comme une chose mineure dans ce cas. Parce qu'ils vont apporter des changements dans la réponse de magnitude au lieu de la fréquence des fréquences sera multiplié par un, ils seraient multipliés par intervalle de [0,9;1].

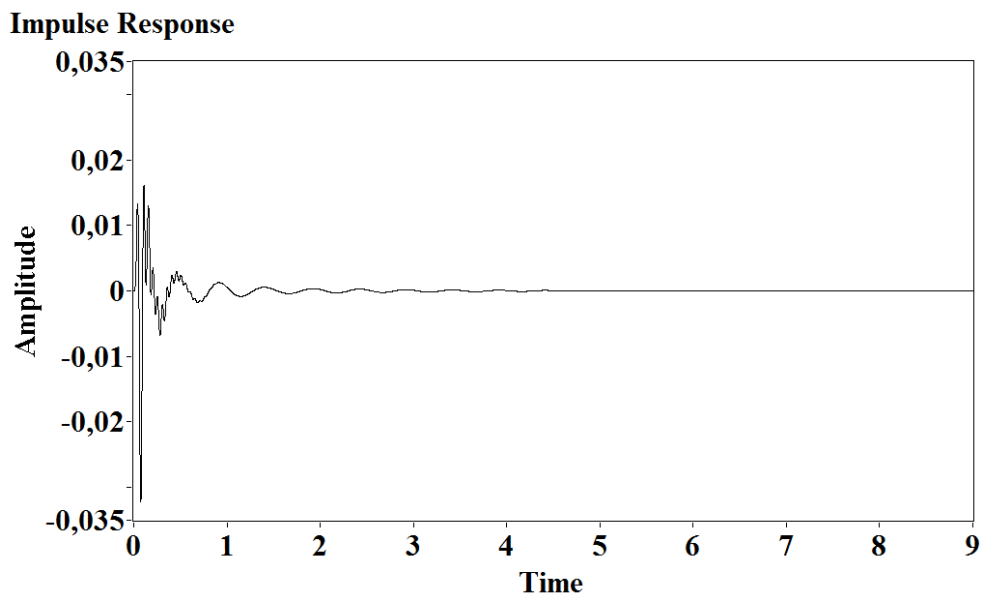


Figure 3.40. Réponse de Chebyshev Impulse.

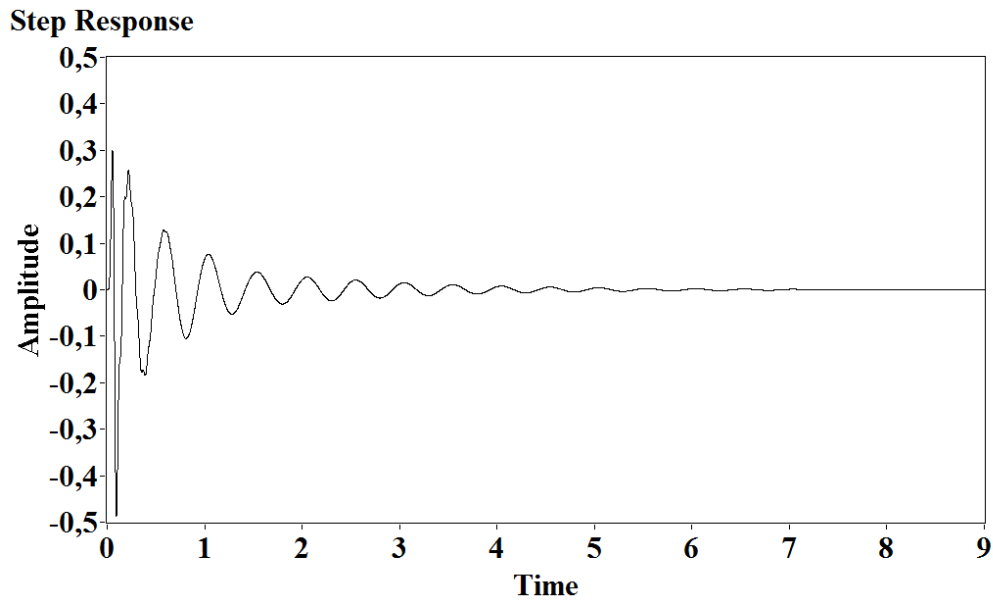


Figure 3.41. Réponse de Chebyshev Step.

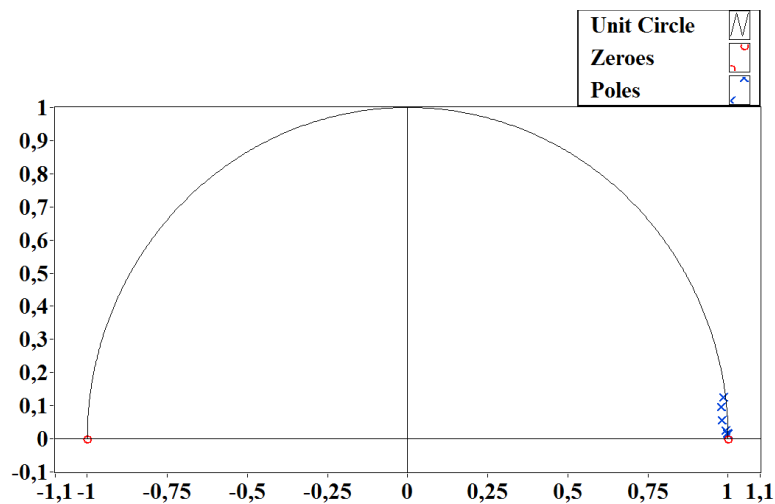


Figure 3.42. Placement de pôle de Chebyshev.

c. Conception de filtre elliptic

Dans cette partie, nous allons concevoir le filtre elliptic pour obtenir le meilleur ou la plus forte réponse de la fréquence que nous pouvons avoir tout comme nous l'avons fait pour les conceptions de filtres précédents. Pour ce faire, nous allons ouvrir le Elliptic Design VI sous la même palette, et il est affiché en Figure 3.31. Fondamentalement, que VI crée une réponse digitale infinie d'impulsion numérique infini (IIR) filtre pour nous.

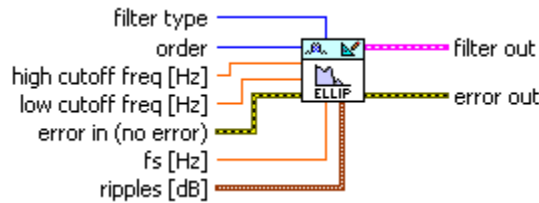


Figure 3.43. Elliptic Design VI.

Après nous avons que VI, nous avons besoin de définir les entrées ou les contrôles pour que VI ainsi que les sorties ou les indicateurs. Pour les contrôles, nous allons créer le type de filtre, l'ordre de conception, haute fréquence de coupure et basse fréquence de coupure à la fois dans le hertz juste comme nous l'avons fait avant, mais nous n'allons pas modifier les ondulations parce que nous n'avons pas accès à ce paramètre dans le Filtre Express VI, donc nous allons le laisser sur la valeur par défaut de 1dB pour l'ondulation de la bande-passe et 60 dB pour l'atténuation de la bande d'arrêt.

Les chiffres suivants montrent la réponse de magnitude, la réponse d'impulsion, la réponse d'étape ainsi que les pôles et le graphe zéros sur le Z-plan.

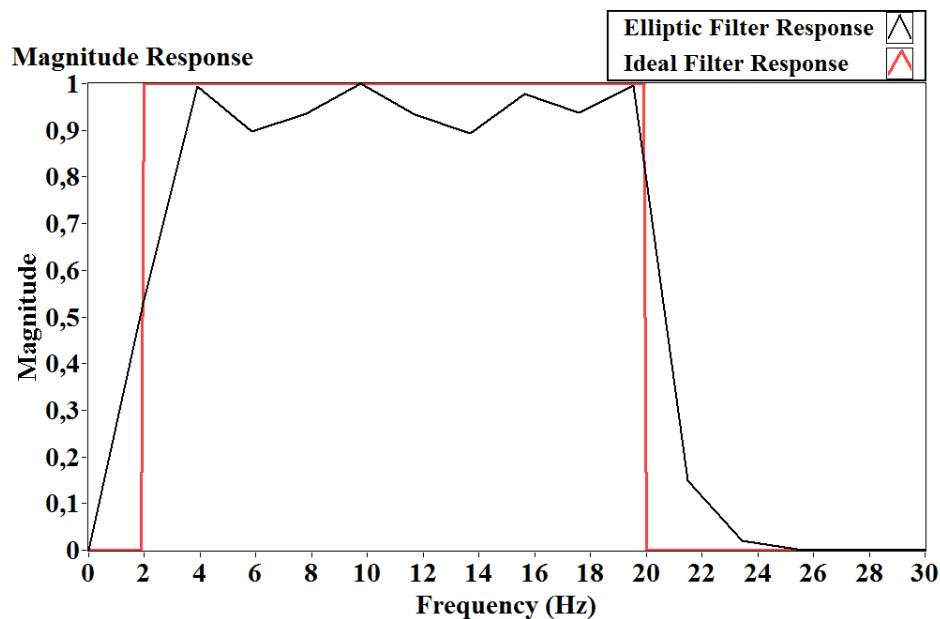


Figure 3.44. Réponse de magnitude elliptic.

Comme nous pouvons le voir sur la réponse de magnitude, les résultats ont été les suivants: un col-bande fréquences d'Edge allant comme suit [4; 20] Hz; et Stop-Band Fréquences allant comme suit [0; 26] Hz. La transition entre la bande passante et la bande d'arrêt est plus nette que celle du filtre Butterworth, mais elle est presque la même que celle du filtre Chebyshev.

Impulse Response

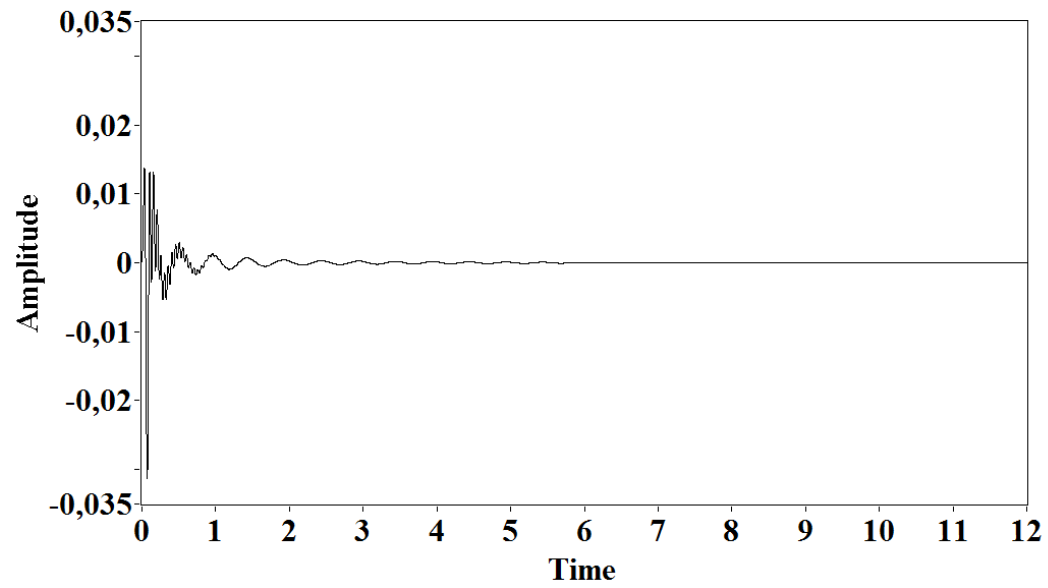


Figure 3.45. Réponse d'Elliptic Impulse.

Step Response

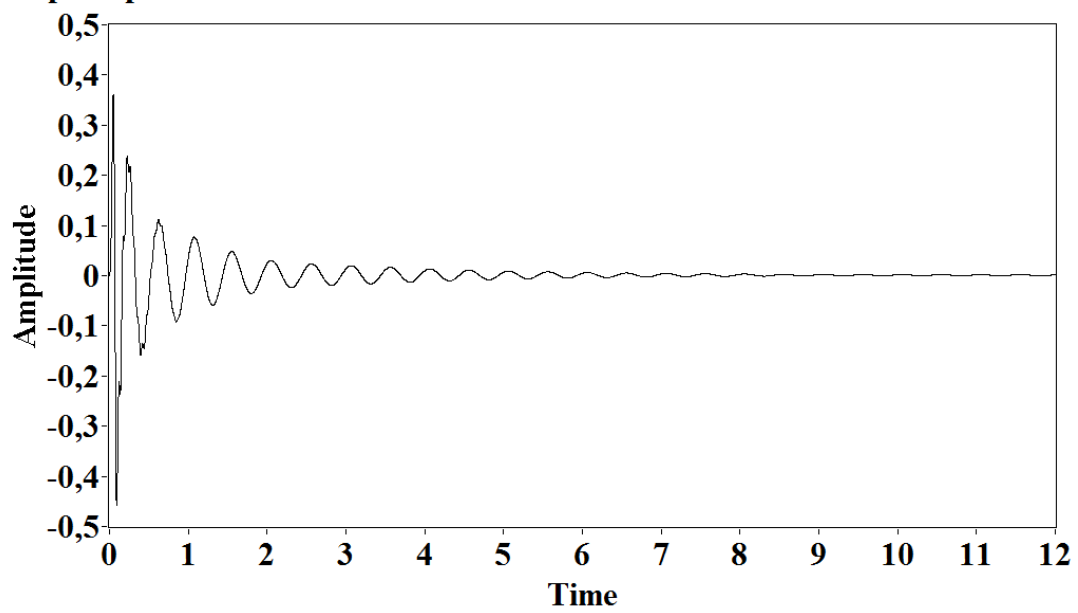


Figure 3.46. Réponse elliptic step.

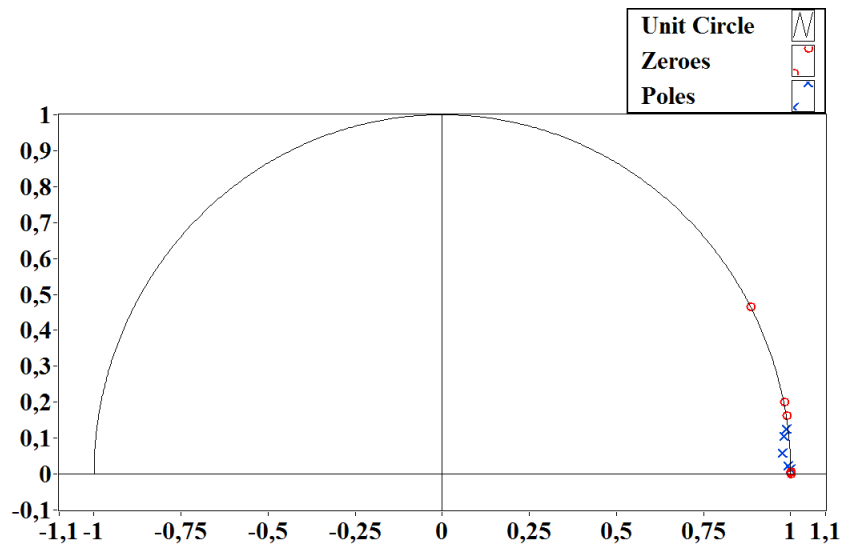


Figure 3.47. Placement de pôle elliptic

d. Conception inverse du filtre Chebyshev

Dans cette partie, nous allons concevoir le filtre Inverse Chebyshev. Pour ce faire, nous allons ouvrir l'InvChebyshev Design VI sous la même palette, et il est montré dans Figure 3.36. Fondamentalement, ce VI crée un numérique InvChebyshev réponse impulsion infinie (IIR) filtre pour nous.

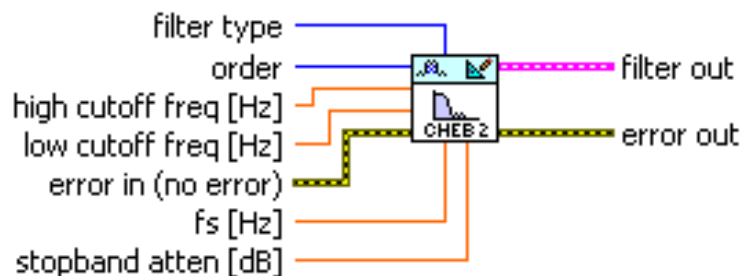


Figure 3.48. Inverse Chebyshev Design VI

Après nous avons que VI, nous avons besoin de définir les entrées ou les contrôles pour que VI ainsi que les sorties ou les indicateurs. Mais nous n'allons pas modifier l'atténuation du stopband parce que nous n'avons pas accès à ce paramètre dans le Filtre Express VI, donc nous allons le laisser sur la valeur par défaut de 60dB. Les chiffres suivants montrent la réponse de magnitude, la réponse d'impulsion, la réponse d'étape ainsi que les pôles et le graphe zéros sur le Z-plan.

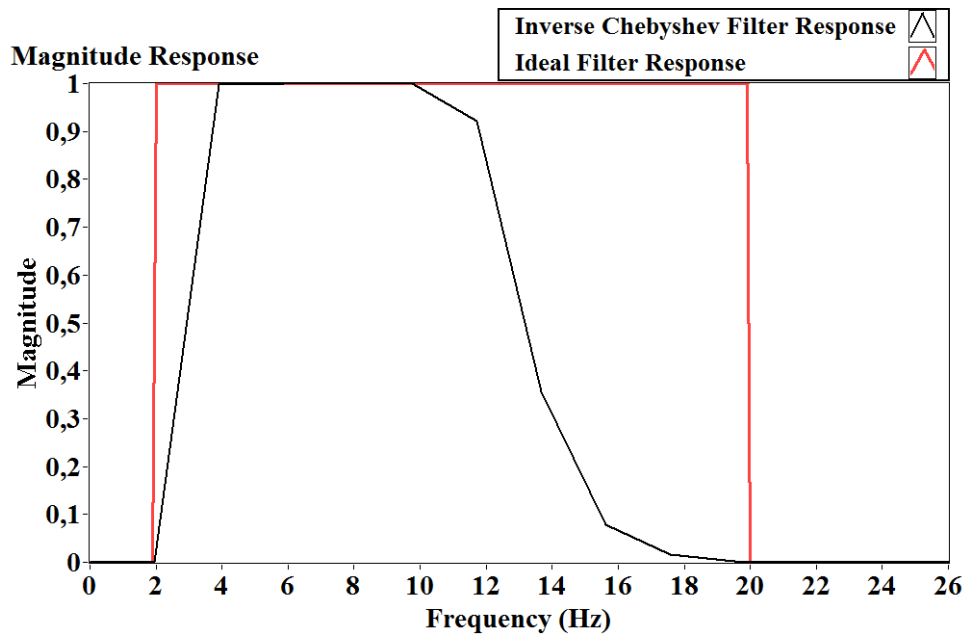


Figure 3.49. Réponse inverse de l'ampleur de Chebyshev.

Comme nous pouvons le voir sur la réponse à la magnitude, les résultats ont été les suivants: un col-bande fréquences de bande allant comme suit [4; 10] Hz; et des fréquences d'arrêt-bande de bord s'étendant comme suit [2 ; 20] Hz, qui est nettement meilleur que les filtres précédents. La transition du stopband à la transition de passband commencera après la basse fréquence de coupure contrairement aux filtres précédents ; tous essaient d'atteindre l'ampleur de l'un juste quand la basse fréquence entre. En outre, dans la bande passe à la transition de la bande d'arrêt, nous pouvons voir une pré-transition similaire. L'amplitude du filtre passe à zéro à droite sur la fréquence de coupe élevée. Dans l'ensemble, nous avons une très bonne réponse.

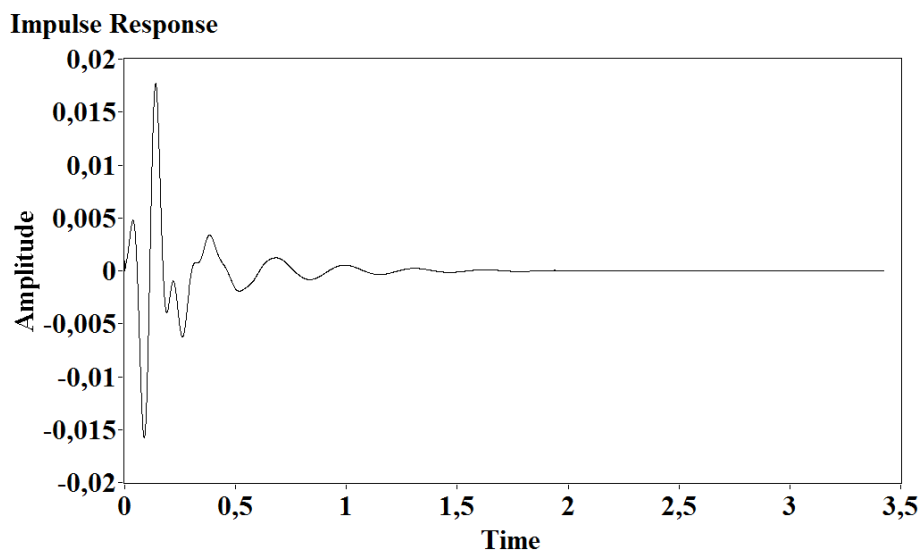


Figure 3.50. Réponse inverse de Chebyshev Impulse.

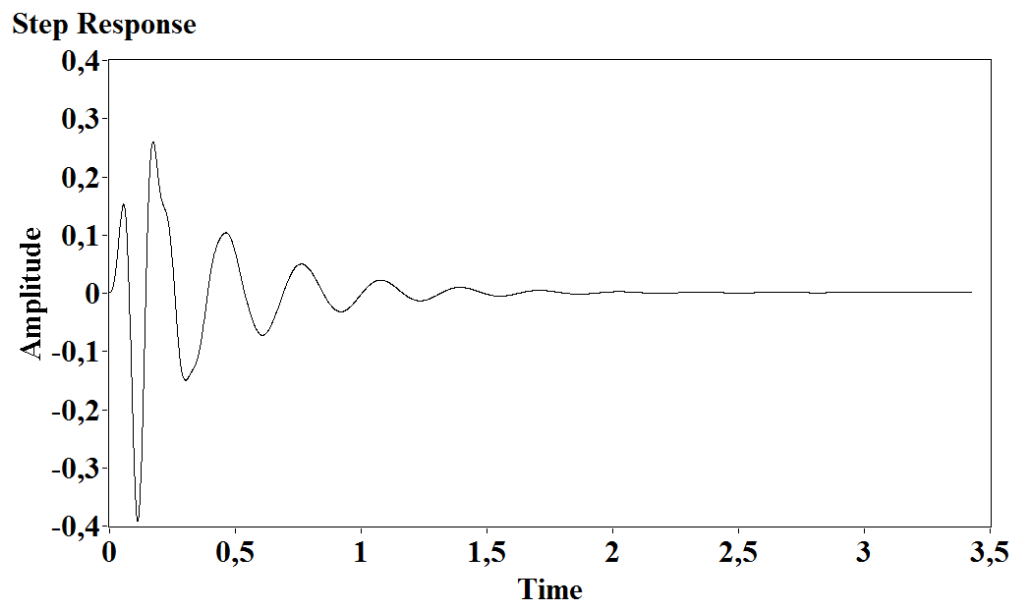


Figure 3.51. Réponse inverse de Chebyshev Step.

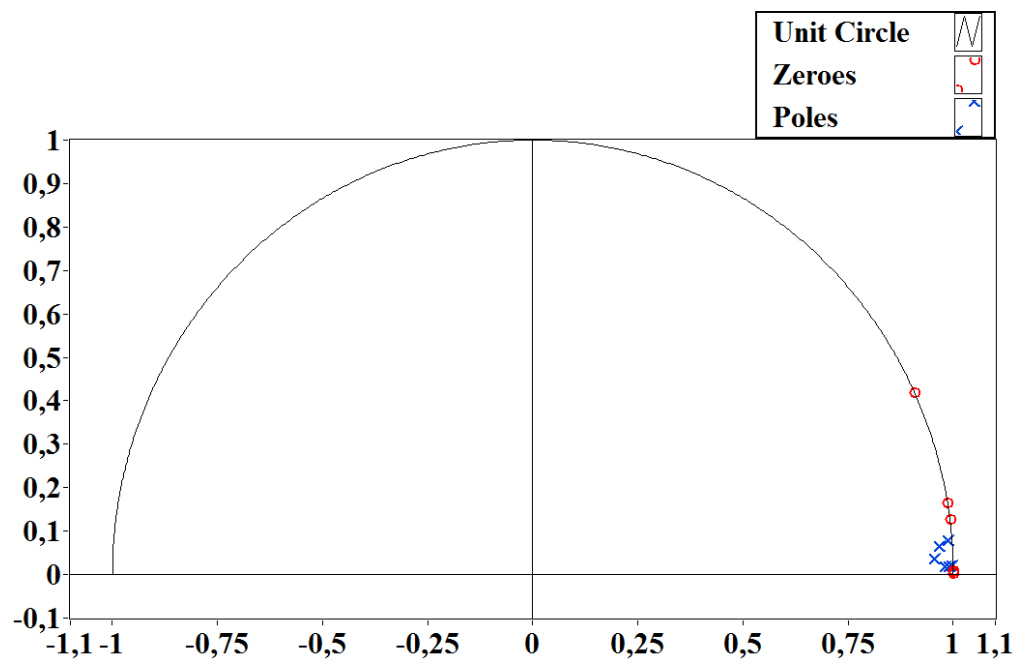


Figure 3.52. Placement inverse de pôle de Chebyshev.

3.7 Implémentation de filtre

Après avoir conçu les filtres IIR, nous pouvons les implémenter sur notre signal. Pour ce faire, nous avons besoin d'obtenir un filtre Express VI, trouvé sous la palette d'Analyse Signal Express.

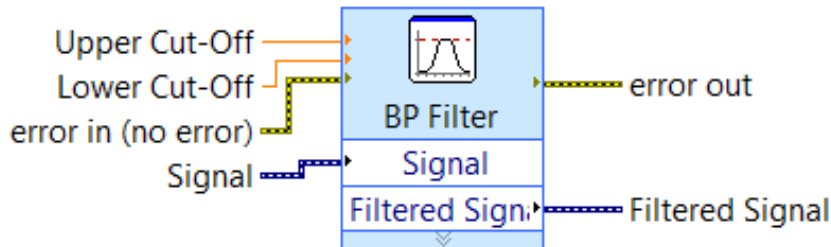


Figure 3.53. Filtre Express VI.

Une fois que nous avons ouvert ou double-cliqué dessus, une boîte de dialogue (Figure 3.42) nous sera montrée avec plusieurs arrangements dans lui. ces paramètres sont le type de filtrage qui peut être un filtre de bande passante, filtre passe-bas, filtre passe-haut, ou bande-stop filtre, et il contient les spécifications du filtre. Dans ces spécifications, nous pouvons définir la basse fréquence de coupure et la fréquence de coupure élevée aussi bien que si nous voulons employer un filtre de réponse d'impulsion finie (FIR) ou un filtre infinie d'impulsion de réponse (IIR). Nous voulons choisir un filtre infinie d'impulsion de réponse ; sous cela, nous pouvons choisir Butterworth, Chebyshev, elliptique ou Inverse Chebyshev topologie; nous pouvons également choisir l'ordre de notre filtre.

Puisque nous savons déjà que, la bande passante de fréquence (Range) des mouvements foetus va être entre 2 Hz et 20Hz, et notre fréquence d'échantillonnage est de 1000 Hz, en plus, la fréquence d'échantillonnage sera plus tard normalisée à 1Hz parce que nous avons acquis les signaux à partir d'un fichier texte. Par conséquent, nous avons besoin de diviser notre gamme désirée de fréquences sur la fréquence d'échantillonnage, de sorte que la plage de fréquences désirée sera alors forme 0.002 Hz à 0.02 Hz.

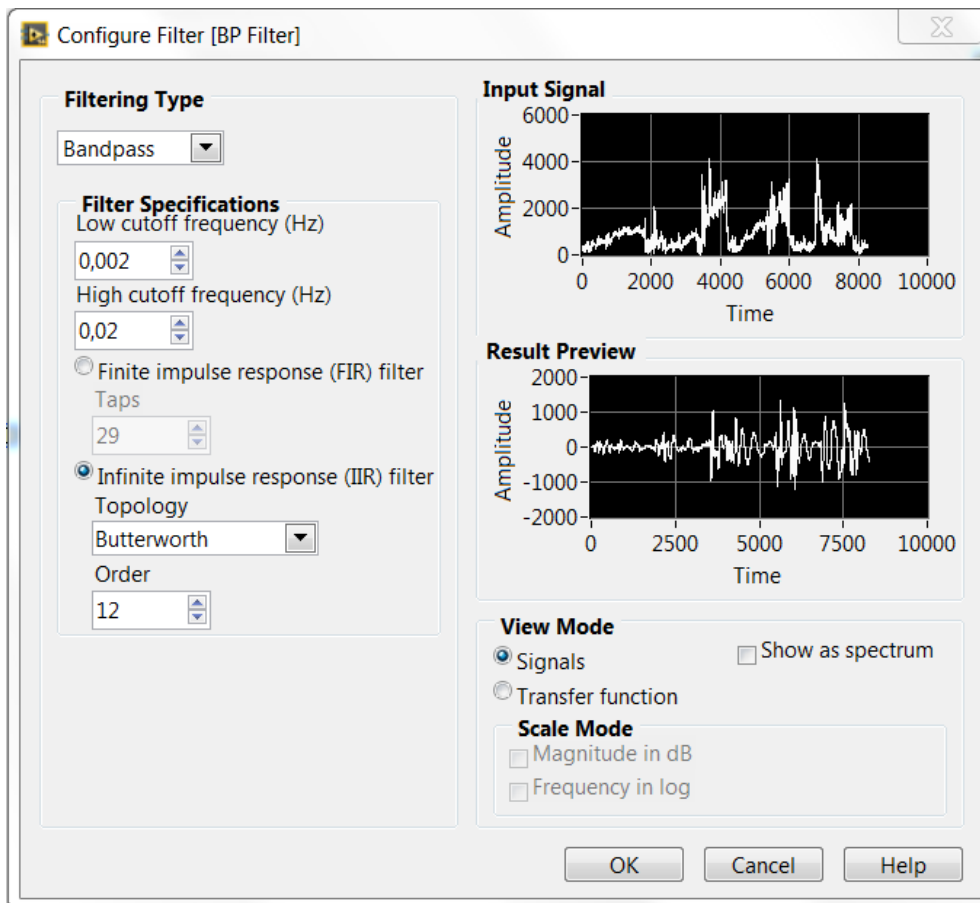


Figure 3.54. Boîte de dialogue Filter Express VI.

Après cela, nous avons besoin de tracer le signal filtré et effectuer la Transformation de Fourier rapide sur ce signal filtré pour vérifier l'effet du filtre choisi sur le signal.

a. Implémentation de filtre Butterworth

Dans cette section, nous appliquerons le filtre Butterworth pour nos signaux et tracer le signal filtré dans le domaine du temps ainsi que dans le domaine de fréquence (Magnitude Response) en utilisant la Transformation de Fourier rapide pour valider l'implémentation du filtre. Nous pouvons voir les résultats de la première expérience dans les graphiques suivants.

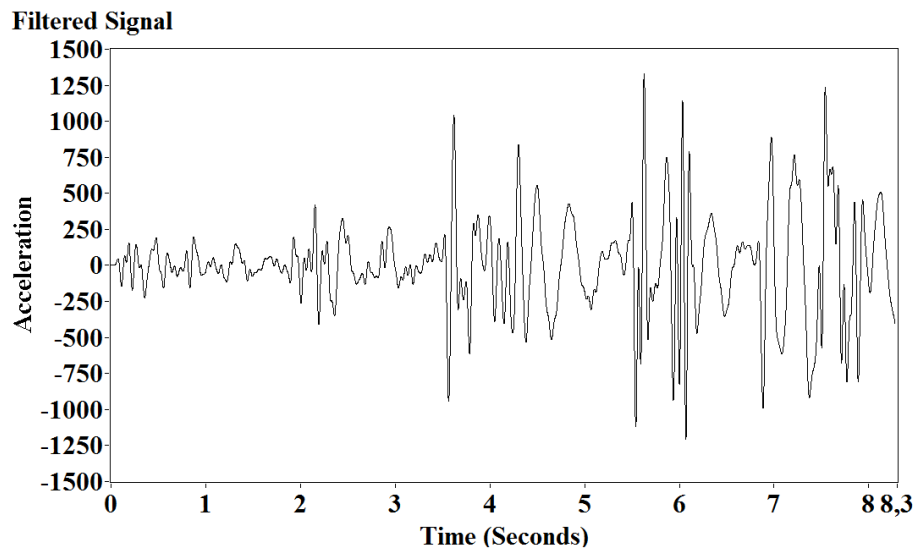


Figure 3.55. RMS filtré à l'aide du filtre Butterworth.

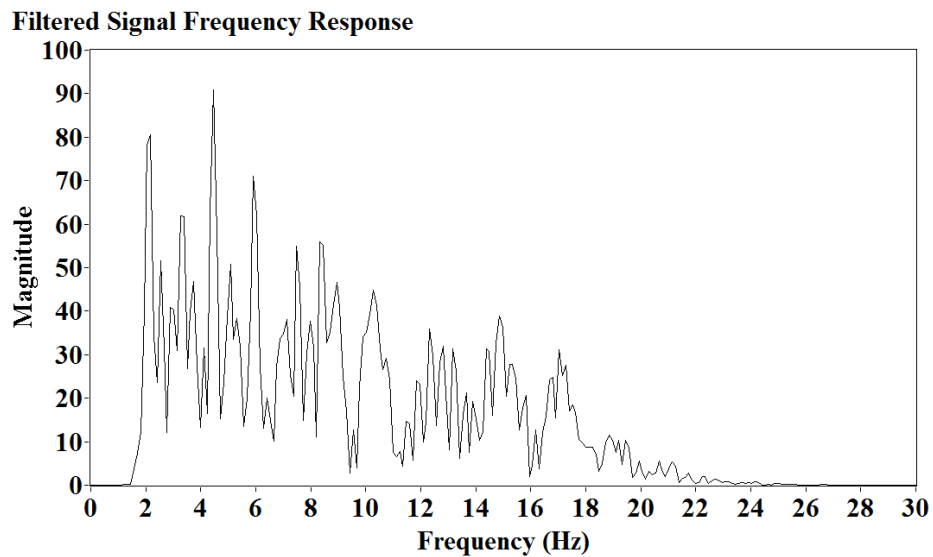


Figure 3.56. Réponse de fréquence de signal filtré (Butterworth).

b. Implémentation de filtre Chebyshev

Dans cette section, nous appliquerons le filtre Chebyshev pour nos signaux comme nous l'avons fait pour le filtre Butterworth et tracer le signal filtré dans le domaine temps ainsi que dans le domaine de fréquence.

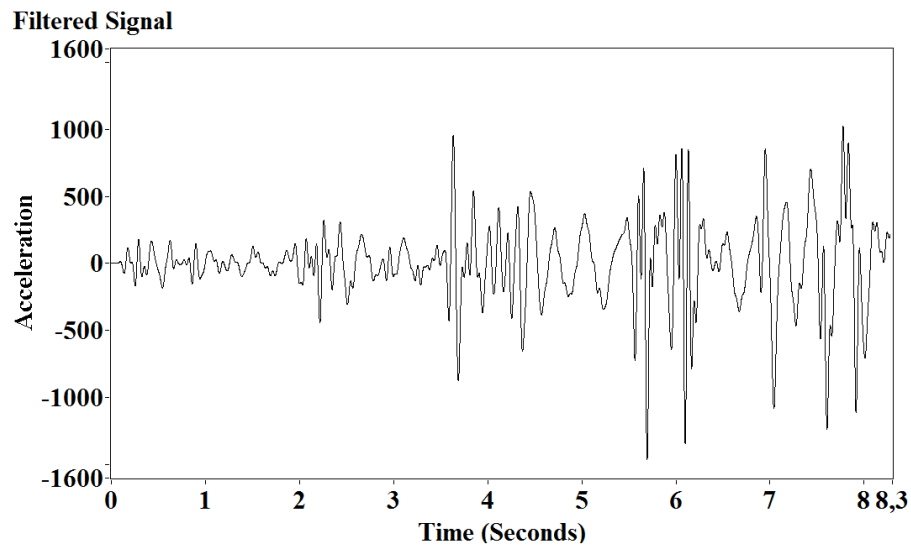


Figure 3.57. RMS filtré à l'aide du filtre Chebyshev.

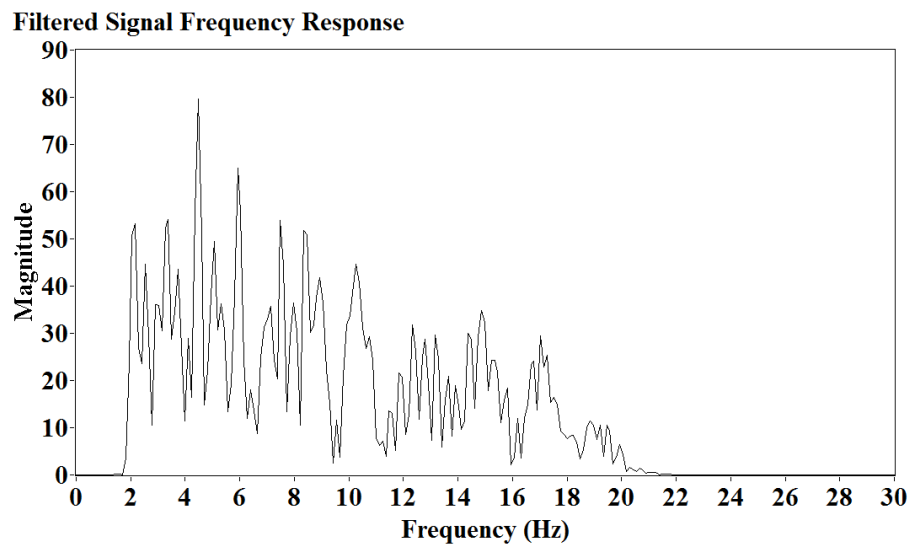


Figure 3.58. Réponse de fréquence de signal filtré (Chebyshev).

c. Implémentation de filtre elliptique

Dans cette partie, nous appliquerons le filtre elliptique pour nos signaux et tracer le signal filtré dans le domaine de temps ainsi que dans le domaine de fréquence comme nous l'avons fait avec les filtres précédents.

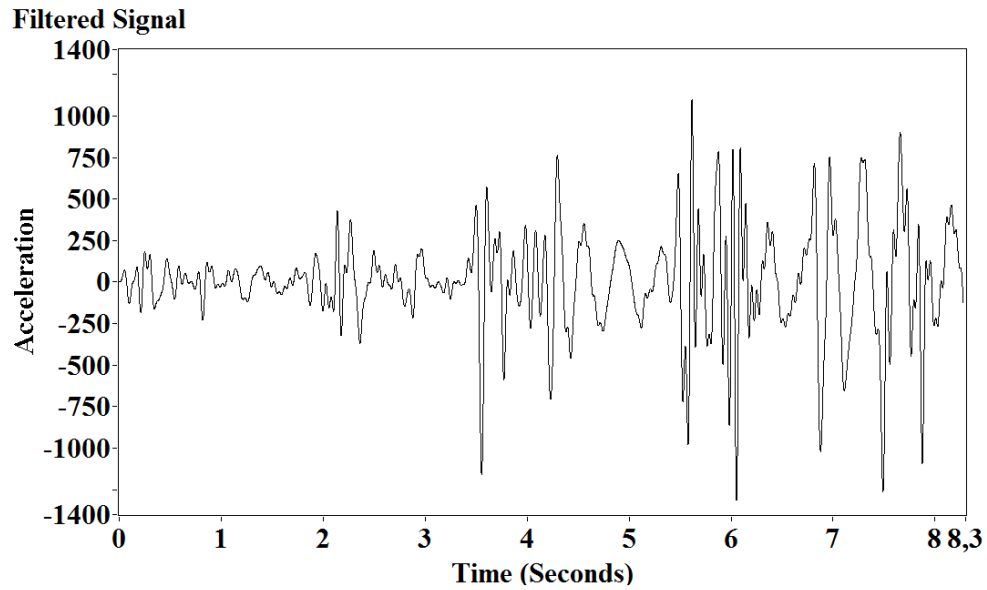


Figure 3.59. RMS filtré à l'aide du filtre elliptic.

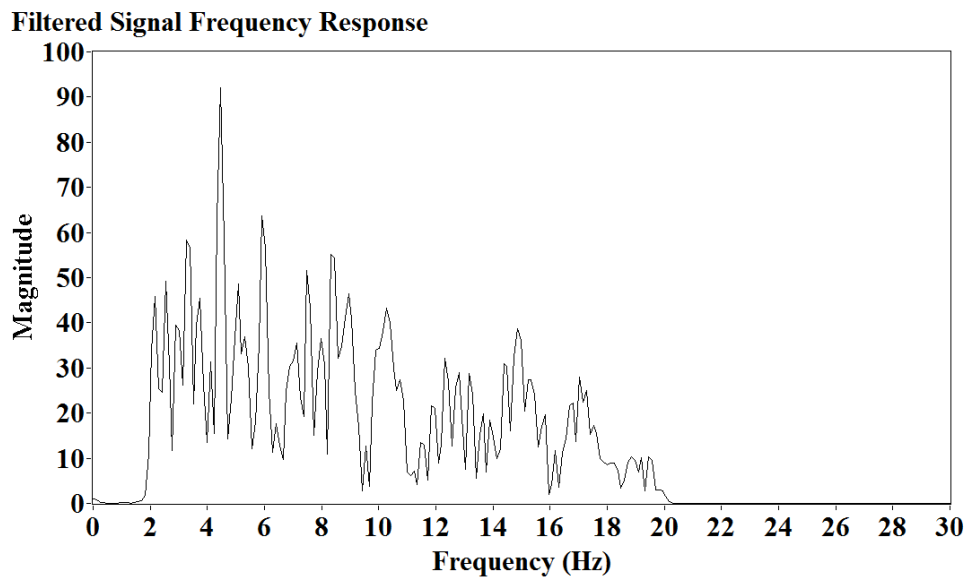


Figure 3.60. Réponse de fréquence de signal filtré (Elliptic).

d. Implémentation de filtre inversée de Chebyshev

Comme nous l'avons fait avant, nous appliquerons le filtre Inverse Chebyshev pour nos signaux et tracer le signal filtré dans le domaine de temps ainsi que dans le domaine de fréquence.

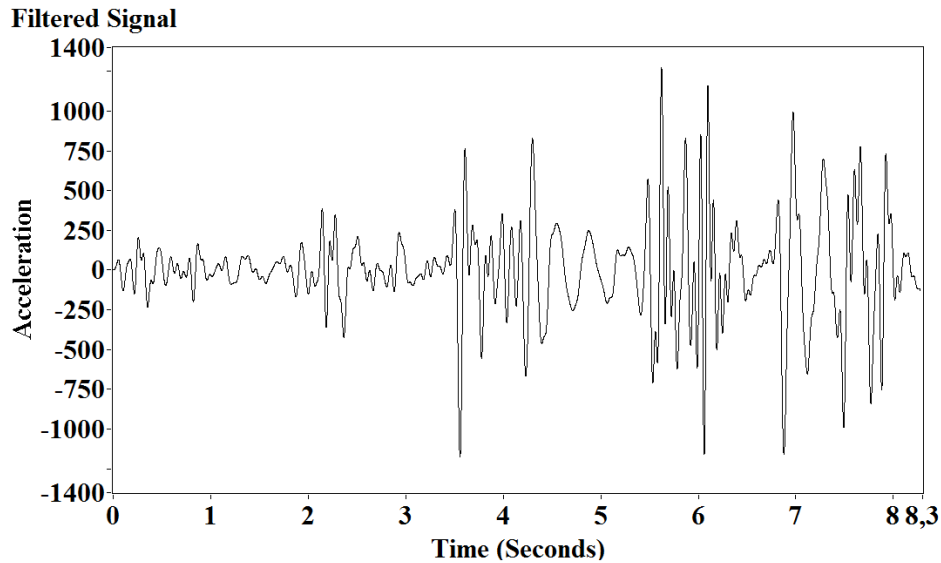


Figure 3.61. RMS filtré à l'aide du filtre Inverse Chebyshev.

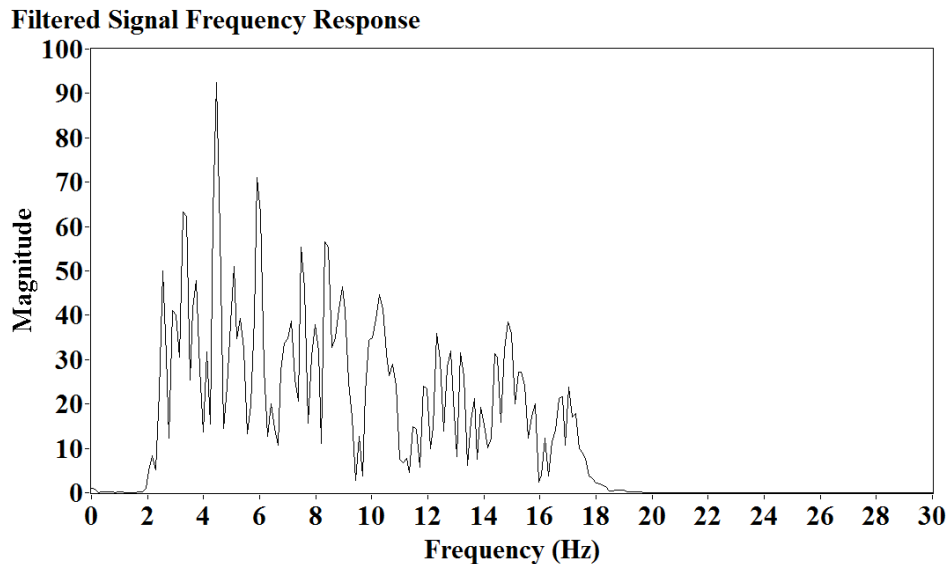


Figure 3.62. Réponse de fréquence de signal filtré (Inverse Chebyshev).

Nous pouvons remarquer de légères différences dans les parcelles de domaine du temps entre le filtre quatre, cependant, la différence est plus perceptible dans la réponse de magnitude. Dans la réponse de magnitude du filtre Butterworth, la décoloration des fréquences est lisse par rapport à la réponse de magnitude Inverse Chebyshev, qui se caractérise d'être la plus forte des quatre, tandis que le Chebyshev et la chute elliptique entre les deux.

3.7.1 Computation de valeur absolu et détection de Seuil

Une fois que nous obtenons le signal filtré, nous pouvons appliquer la valeur absolue à ce signal pour augmenter sa résolution pour la détection de seuil. Pour obtenir la valeur absolue du signal, il suffit de le passer à travers le bloc Valeur Absolue trouvé sous la palette numérique. Le signal sortant du bloc sera désigné comme le signal final ; après cela, la phase de détection du seuil commence.

Un algorithme de détection de seuil sera appliqué au signal final. Afin d'effectuer cela, nous allons utiliser le bloc supérieur ou égal trouvé sous la palette booléenne ; l'entrée de ce bloc sera le signal final que nous venons d'avoir et la valeur seuil que nous vérifions si le signal est au-dessus ou au-dessous. La sortie de ce bloc sera une matrice booléenne qui sera transformée en un masque binaire de zéros et de ceux utilisant le bloc "booléen" à (0,1). Figure 3.51 représente cette partie du code.

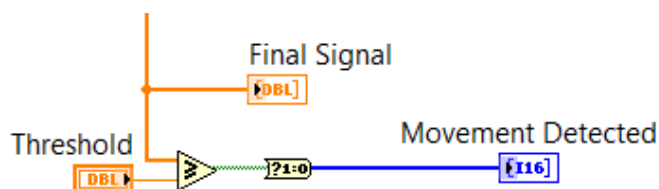


Figure 3.63. Diagramme de bloc d'algorithme de détection de seuil.

Le masque binaire qui est la sortie du bloc "booléen" à (0,1), sera le mouvement détecté par la méthode que nous appliquons ; il sera tracé et comparé au masque binaire généré par le Button0 .

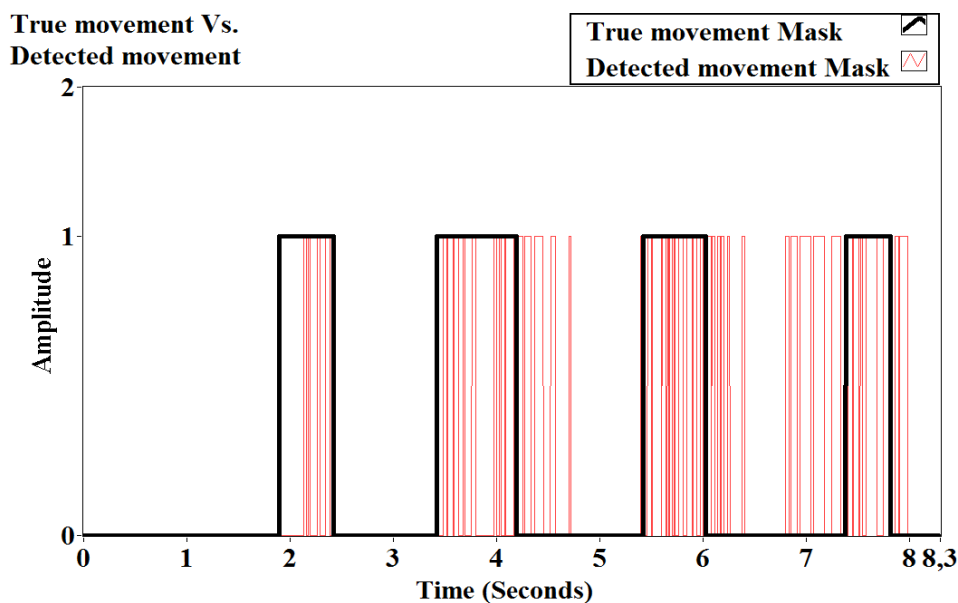


Figure 3.64. Masque binaire du mouvement détecté contre le vrai mouvement.

3.7.2 Évaluation de la détection du mouvement féministe algorithme

Une fois le graphe sur Figure 3.52 est obtenu, nous pouvons développer une méthode pour évaluer les performances de notre méthode de filtrage. Nous allons le faire en utilisant le test de classification binaire ainsi que la mesure statistique.

a. Test de classification binaire

Pour générer les graphiques True Positive (TP), True Negative (TN), False Positive (FP) et False Negative (FN), nous n'avons besoin que de deux entrées l'une d'entre elles est le vrai mouvement, qui est un masque binaire qui vient du button0 et l'autre est le masque binaire détecté par le mouvement qui provient de l'algorithme de détection de seuil. Ces deux entrées seront ensuite comparées par des blocs booléens pour générer les graphiques recherchés.

True Positive graphique: Afin de tracer le graphique du vrai positif, nous avons besoin de prendre le vrai mouvement binaire masque entrée et le comparer à un (True), et prendre le masque binaire mouvement détecté et le comparer avec un (True). Pour ce faire, nous devons prendre le bloc égal de la palette de comparaison et insérer en elle la première entrée, qui est le vrai masque binaire de mouvement, et pour la deuxième entrée, nous avons besoin d'ajouter une véritable constante booléenne. Il en va de même pour le mouvement détecté. Ensuite, nous allons prendre la sortie des deux blocs égaux et en faire l'entrée de la, « Et » bloc de porte logique trouvé sous la palette booléenne. La sortie de cette porte logique est notre véritable positif; nous devons le convertir du formulaire booléen en formulaire (1,0) Figure 3.53 montrera le diagramme de bloc associé à cette partie.

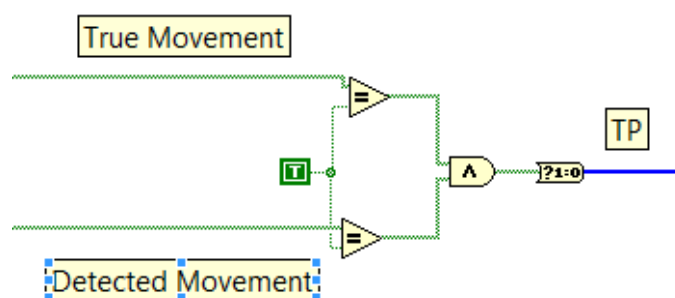


Figure 3.65. Diagramme de bloc de calcul positif vrai.

Graphique true Négatif : Afin de tracer le graphique du vrai négatif, nous avons besoin de prendre le mouvement véritable masque binaire entrée et de le comparer à zéro (False) et de prendre le mouvement détecté masque binaire et par rapport à zéro (False). Figure 3.54 montrera le diagramme de bloc associé à cette partie.

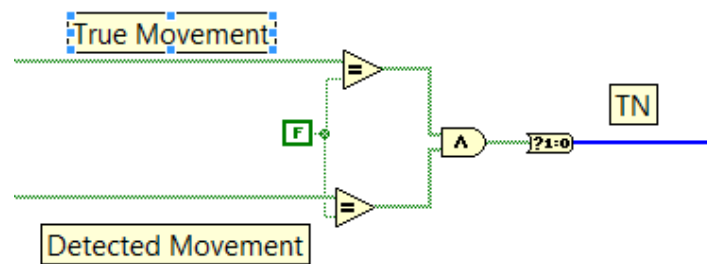


Figure 3.66. Diagramme de bloc de calcul négatif vrai.

Graphique false positif: Afin de tracer le graphique du faux positif, nous avons besoin de prendre le mouvement réel masque binaire entrée et de le comparer à zéro (False) et de prendre le masque détecté mouvement binaire et de le comparer avec un (True). Figure 3.55 montrera le diagramme de bloc associé à cette partie.

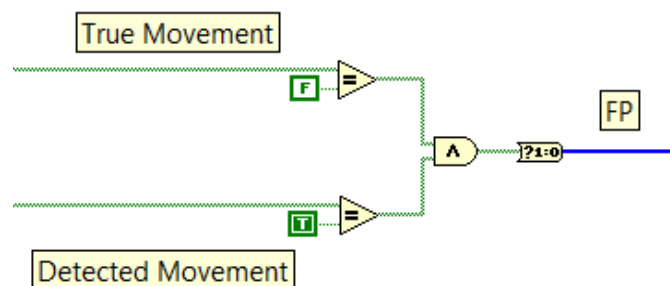


Figure 3.67. Diagramme de bloc de calcul faux positif.

Graphique false négatif: Afin de tracer le graphique du faux négatif, nous avons besoin de prendre le mouvement réel masque binaire entrée et de le comparer à un (True) et de prendre le mouvement détecté masque binaire et il a comparé avec zéro (False). Figure 3.56 montrera le diagramme de bloc associé à cette partie.

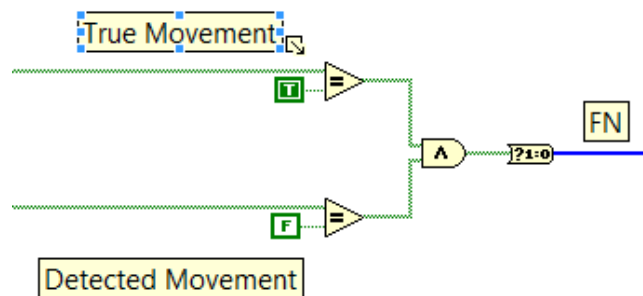


Figure 3.68. Diagramme de bloc de calcul faux négatif.

Puisque nous avons les tableaux, nous avons maintenant besoin de calculer ces variables (TP, TN, FP, FN) directement. Pour ce faire, nous avons besoin de compter chaque échantillon où la valeur est une. Parce que les graphiques seront également un masque binaire, une fois que la valeur est une qui signifie que la valeur existe.

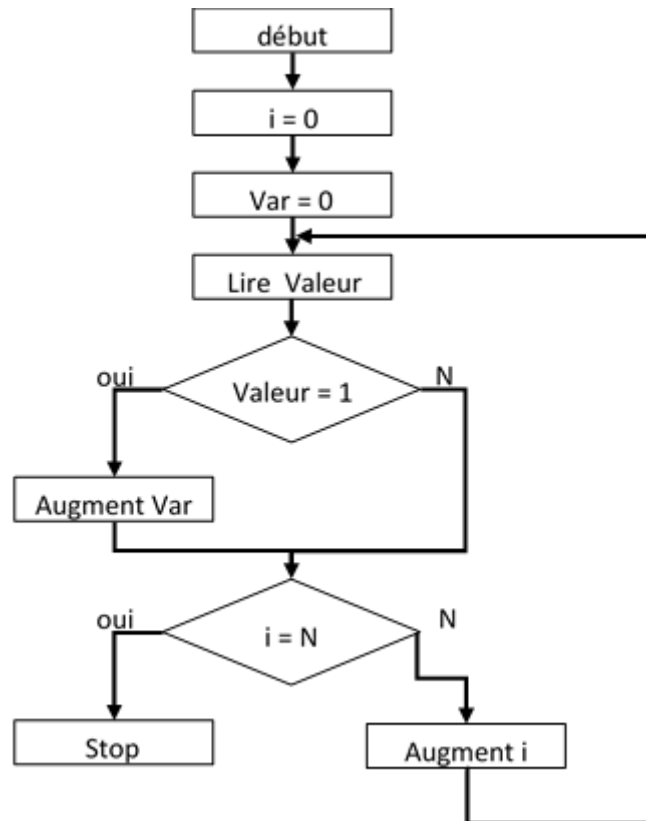


Figure 3.69. Algorithme de calcul des valeurs de test de classification binaire.

La Figure 3.57 montre l’algorithme que nous avons utilisé pour calculer chacune des variables précédentes. L’algorithme commence par l’initialisation de la variable (TP, TN, FP, FN) ainsi que l’itération de la boucle qui représente les échantillons de signaux. Après cela, nous lisons la valeur de la variable et vérifions si elle est égale à une, si oui, nous augmentons la variable. Si non, nous procédons dans la boucle. Ensuite, nous vérifions si l’itération est égale à "N" qui représente le dernier échantillon du signal.

Pour appliquer cet algorithme sur LabVIEW, la première chose que nous devons faire est de créer des variables globales. Une fois que nous faisons cela, nous pouvons choisir si nous voulons lire la variable.

Après cela, nous devons créer une structure de séquence. La première séquence consiste à initialiser la variable à zéro. La deuxième séquence est pour le calcul; la troisième séquence est pour indiquer la valeur de la variable.

Pour calculer l’une des valeurs FN ou FP TN TP, nous devons d’abord attacher le tableau correspondant au bloc ' "Array size" trouvé sous la palette Array, ce qui nous aidera à déterminer le nombre d’itérations pour la boucle pour.

Ensuite, nous créons une boucle pour à l’intérieur de la deuxième séquence et attachons la sortie du bloc de taille Array à l’entrée N de la « for loop », Ensuite, nous prenons la valeur d’itération qui existe à l’intérieur de la « for Loop » et l’attacher au bloc “Index Array” trouvé sous la palette Array ainsi. Nous devons attacher notre tableau à l’entrée "n-dimension

array" et attacher également l'itération de la boucle "for loop" l'entrée index du bloc Index Array. La sortie de ce bloc retournera l'élément de ce tableau, qui, nous le savons, pourrait être zéro ou un. Après cela, nous devons le passer par un bloc « égal » pour vérifier s'il est égal à un. La sortie du bloc « equal » nous donnera une valeur booléenne; cette valeur booléenne sera le sélecteur de cas de notre structure de cas que nous allons créer. La structure de cas peut être trouvée sous la palette de structures, nous laissons le cas false vide, mais dans le cas vrai, nous devons mettre notre variable globale et augmenter sa valeur d'une dans chaque itération de la « boucle » lorsque la valeur du tableau d'entrée est égale à une. Cette procédure entière sera montrée dans Figure 3.58 avec le TP pris comme exemple.

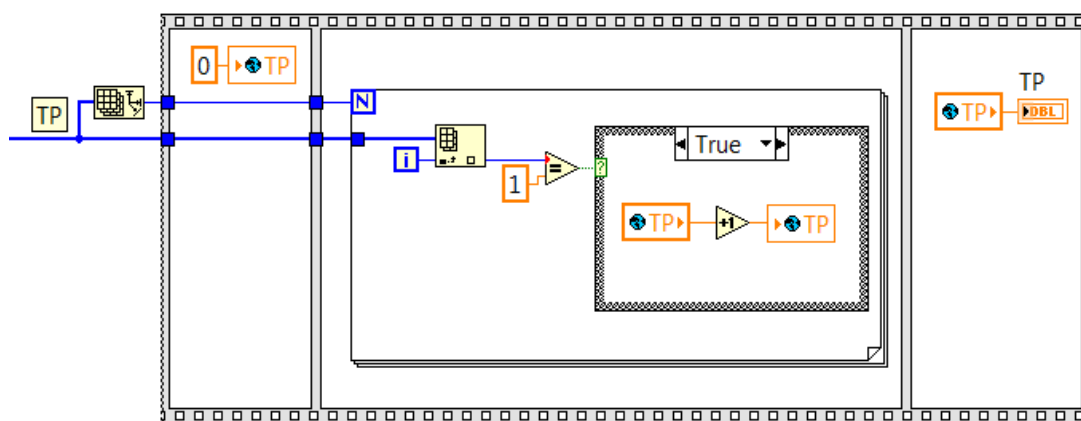


Figure 3.70. Diagramme de bloc de calcul des valeurs de test de classification binaire.

b. Mesures statistiques

Une fois que nous obtenons notre TN, TP, FN, et FP, tout ce qu'il nous reste est de calculer la sensibilité, la spécificité, et la précision de nos expériences. Le diagramme de bloc pour cette étape sera à la recherche comme suit:

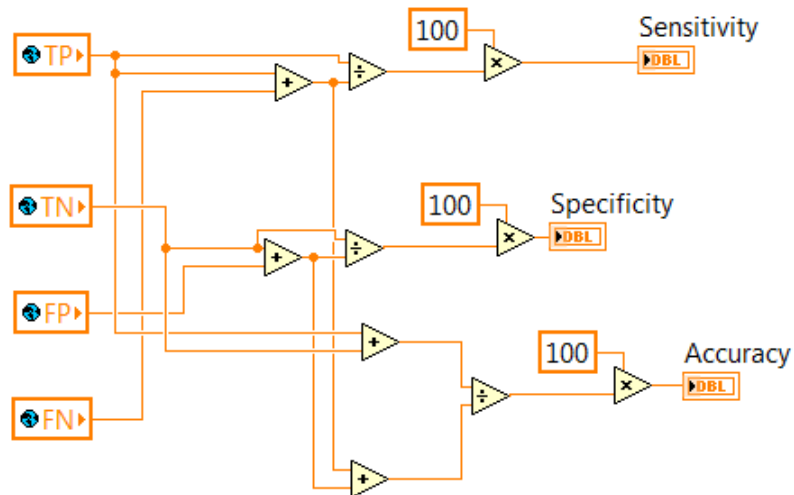


Figure 3.71. Diagramme de bloc de calcul des mesures statistiques.

Ce diagramme de bloc utilisera les variables préalablement calculées (TP, TN, FP, FN), et il retournera les valeurs de mesure statistique en pourcentage.

Une fois que nous appliquons toutes les étapes et méthodes précédentes, nous nous retrouvons avec un système qui ressemble à Figure 3.60. Ce chiffre résume le système complet de détection du mouvement foetal.

Conception du filtre

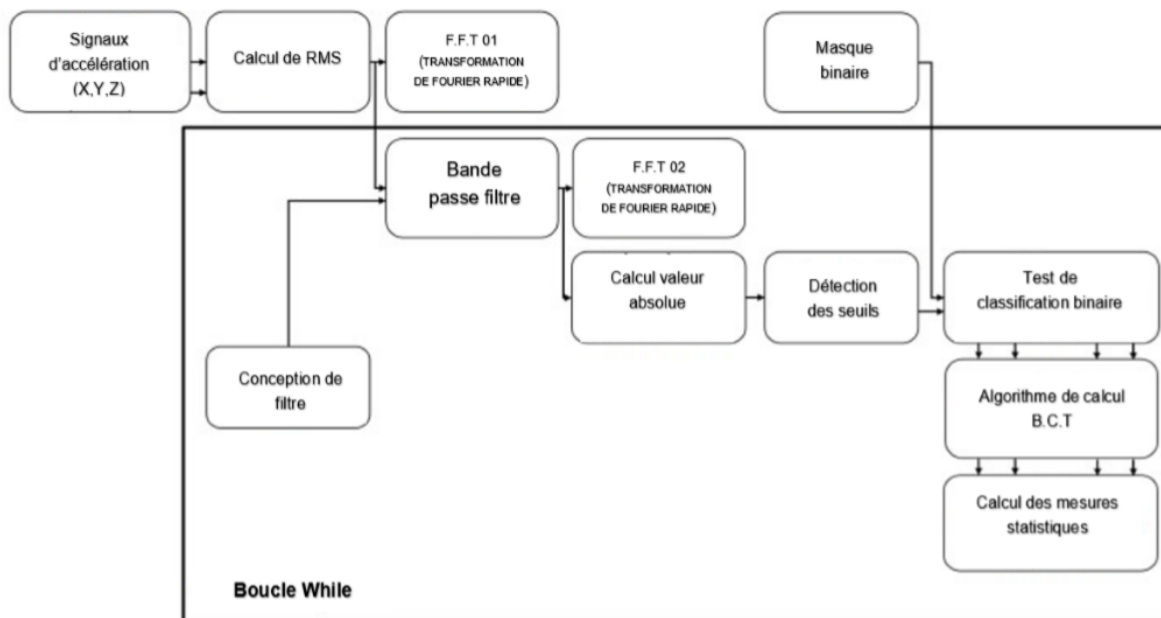


Figure 3.72. Diagramme de bloc de calcul des mesures statistiques.

3.8 Résultats de l'évaluation du rendement de la méthode de détection des mouvements fœtaux

Dans cette section discutera des résultats provenant de trois expériences différentes, Dans chacune de ces expériences, nous avons différents signaux acquis, et pour chacun, nous avons appliqué quatre filtres. Pour chaque filtre, nous choisirons un seuil différent qui permet au filtre d'avoir une bonne performance. Les performances de ce filtre seront jugées en fonction de la spécificité et de la précision.

Ces résultats seront résumés dans le tableau suivant, recueillant la sensibilité, la spécificité et l'exactitude. Le tableau d'évaluation complet figurera à l'Annexe.

Filters	Mesures de statistiques		
	Sen. (%)	Spéc. (%)	Pré. (%)
Butterworth	36.60	80.19	68.23
Chebyshev	36.41	80.96	68.47
Elliptic	39.24	82.97	70.76
Inv. Chebyshev	39.93	83.34	71.14

Tableau 3.6. L'IIR filtre les performances pour la détection des mouvements fœtaux avec différents seuils.

A partir des résultats précédents ainsi que des seuils précédents utilisés, nous allons mettre en place un seuil qui convient à toutes les expériences. Nous avons calculé la moyenne des seuils précédents utilisés et nous les appliquons aux trois expériences précédentes sur tous les filtres. Nous pouvons voir comment ce seuil moyen se produira dans le tableau suivant.

Filtres	Mesures statistiques		
	Sen. (%)	Spéc. (%)	Pré. (%)
Butterworth	37.50	79.12	67.37
Chebyshev	34.04	80.17	67.06
Elliptic	38.67	81.70	69.24
Inv. Chebyshev	37.42	84.38	71.21

Tableau 3.7. L'IIR filtre les performances pour la détection des mouvements fœtaux avec un seuil moyen.

3.9 Conclusion

Dans ce chapitre, nous avons parlé sur le Système et les méthodes d'acquisition en suite nous allons passer par Méthodologie de détection de mouvement fetale

on a aussi présenté Résultats de l'évaluation du rendement de la méthode de détection des mouvements foetaux et Nous avons également donné le schéma général et la réalisation. Nous avons terminé le chapitre en effectuant différents tests.

Conclusion générale

La mise en œuvre des filtres Infinité Impulse Response pour détecter les mouvements fœtaux a montré des résultats prometteurs. Si nous prenons un coup d'œil au Tableau 3.2, nous pouvons voir que le filtre Inverse Chebyshev avait une précision de 71,21% ainsi que 84,38% de spécificité; c'est pourquoi nous le considérons comme le meilleur filtre dans une telle application. Les autres filtres n'ont pas eu de mauvais résultats non plus; prenant le filtre elliptique comme exemple, nous pouvons voir qu'il a eu à peu près les mêmes résultats que le filtre Inverse Chebyshev. Ces résultats étaient attendus en raison de la netteté de la réponse d'ampleur des filtres excellents.

À partir de ces résultats, nous avons montré que cette méthode est très prometteuse; cependant, nous pouvons améliorer cette méthode en appliquant des méthodes de lissage du signal pour obtenir un « masque binaire de mouvement détecté » cohérent, spécifiquement pour éliminer la commutation entre la valeur « un » et « zéro » dans le binaire comme, si cela est réalisé, nous aurons un intervalle plat de « un » ou « zéro ». En outre, nous pouvons utiliser l'un des capteurs d'accéléromètre pour enregistrer le mouvement maternel et les soustraire du signal principal en utilisant le filtrage adaptatif pour se retrouver avec une meilleure identification du mouvement fœtal.

Références

- [1] Royal College of Obstetricians and Gynaecologists, “Reduced Fetal Movement,” Green-top Guideline No. 57, pp. 1-3, 2011.
- [2] J. Bryant, R. T. Jamil and J. Thistle., “Fetal Movement,” The National Center for Biotechnology Information, 2019.
- [3] Royal College of Obstetricians and Gynaecologists, “Your baby’s movements in pregnancy,” Health and care information that you can trust, pp. 1-4, 2019.
- [4] Enas W. Abdulhay, Rami J. Oweis, Asal M. Alhaddad, Fadi N. Sublaban, Mahmoud A. Radwan, Hiyam M. Almasaeed, “Review Article: Non-Invasive Fetal Heart Rate Monitoring Techniques,” Biomedical Science and Engineering, vol. 2, no. 3, pp. 53-54, 2014.
- [5] C. S. BERGER AND P. TRIGG, “The Measurement of Fetal Movement Using a Strain-Gauge Transducer,” IEEE Transactions on Biomedical Engineering, Vols. BME-28, no. 11, pp. 787-790, 1981.
- [6] Pathara Norasethasopon, Kitiphol Chitsakul, Suradej Tretriluxana, “A Development of Real-Time Fetal Movement Detector,” IEEE EMBS Conference on Biomedical Engineering and Sciences, pp. 637-640, 2016.
- [7] Rania Hussien Al-Ashwal and Amirul Ridhwan bin Sazali, “Foetal movement detection and characterization based on force sensing,” CURRENT SCIENCE, vol. 115, no. 4, pp. 629-632, 2018.
- [8] Lvchen Zhao, Wei Wu, Xianyi Zeng, Ludovic Koehe, Guillaume Tartare, “A new method for Fetal Movement Detection Using an Intelligent T-shirt Embedded Physiological Sensors,” Proceedings of ICCT, pp. 563-567, 2015.
- [9] Eiji Ryo, Kyoko Nishihara, Sachiyo Matsumoto, Hideo Kamata, “A new method for long-term home monitoring of fetal movement by pregnant women themselves,” Medical Engineering & Physics, p. 566– 572, 2012.
- [10] Eiji Ryo, Kyoko Nishihara, Sachiyo Matsumoto, Hideo Kamata, “A long-term monitoring of fetal movement at home using a newly developed sensor: An introduction of maternal micro-arousals evoked by fetal movement during maternal sleep,” ScienceDirect , p. 595–603, 2008.

- [11] Eiji Ryo, Kyoko Nishihara, Sachiyo Matsumoto, Hideo Kamata, “Automated Software Analysis of Fetal Movement Recorded during a Pregnant Woman’s Sleep at Home,” PLOS ONE, pp. 1-15, 2015.
- [12] SOPHIE RIBES, JEAN-MARC GIRAULT, FRANCK PERROTIN, and DENIS KOUAM, “Multidimensional Ultrasound Doppler Signal Analysis for Fetal Activity Monitoring,” Ultrasound in Med. & Biol., p. 1–10, 2015.
- [13] Jonathan Lai, Richard Woodward, Yuriy Alexandrov, Qurratul ain Munnee, Christoph C. Lees, Ravi Vaidyanathan, Niamh C. Nowlan, “Performance of a wearable acoustic system for fetal movement discrimination,” PLOS One, pp. 1-14, 2018.
- [14] Girier Thomas, O 'Toole John, Mesbah Mostefa, Boashash Boualem, Clough Ian, Wilson Stephen, Fuentes Miguel, Callan Susan, East Christine, Colditz Paul, “Detecting Fetal Movements using Non-invasive Accelerometers: A Preliminary Analysis,” 10th International Conference on Information Science, Signal Processing and their Applications, pp. 508-511, 2010.
- [15] M. Mesbah, M. S. Khlif, C. East, J. Smeathers, P. Colditz and B. Boashash, “Accelerometer-based Fetal Movement Detection,” 33rd Annual International Conference of the IEEE EMBS, pp. 7877-7880, 2011.
- [16] M. S. Khlif, B. Boashash, S. Layeghy, T. Ben-Jabeur, M. Mesbah, C. East, and P. Colditz, “Time-Frequency Characterization of Tri-Axial Accelerometer Data for Fetal Movement Detection,” IEEE, pp. 466-471, 2011.
- [17] Siamak Layeghy, Ghasem Azemi, Paul Colditz, and Boualem Boashash, “Classification of Fetal Movement Accelerometry Through Time-Frequency Features,” Crown, pp. 1-6, 2014.
- [18] Nor Diana binti Zakaria, Mohd Najeb Jamaludin and M. B. Malarvili, “Fetal Activity Recognition Using 3-Axis Accelerometer Sensor,” IEEE Conference on Biomedical Engineering and Sciences, pp. 343-346, 2014.
- [19] Nor Diana Zakaria, Paulson Eberechukwu Numan and M. B. Malarvili, “Fetal Movements Recording System Using Accelerometer Sensor,” ARPN Journal of Engineering and Applied Sciences, vol. 13, no. 3, pp. 1022-1032, 2018.
- [20] “Arduino,” Arduino, 2019. [Online]. Available: <https://www.arduino.cc/>. [Accessed 13 11 2019].
- [21] “Matlab,” MathWorks, 1994-2019. [Online]. Available: <https://www.mathworks.com/products/matlab.html>. [Accessed 18 11 2019].
- [22] Marco Altini, Patrick Mullan, Michiel Rooijakkers, Stefan Gradl, Julien Penders,

- “Detection of Fetal Kicks Using Body-Worn Accelerometers During Pregnancy: Trade-offs Between Sensors Number and Positioning,” IEEE, pp. 5319-5322, 2016.
- [23] “LabVIEW,” Wikipedia, 22 09 2019. [Online]. Available: <https://en.wikipedia.org/wiki/LabVIEW>. [Accessed 18 11 2019].
- [24] University of Debrecen, 2019. [Online]. Available: <https://www.edu.unideb.hu/>.
- [27] “Accelerometer,” Wikipedia, 16 11 2019. [Online]. Available: <https://en.wikipedia.org/wiki/Accelerometer>. [Accessed 18 11 2019].
- [28] D. Nosovitz, “Testing Helmets With Impact-Sensing Accelerometers for Concussion Analysis,” Popular Science, 2014.
- [29] “MPU6050 - Accelerometer and Gyroscope Module,” Componants 101, 2019. [Online]. Available: <https://components101.com/sensors/mpu6050-module>. [Accessed 18 11 2019].
- [30] “I²C,” Wikipedia, 14 11 2019. [Online]. Available: <https://en.wikipedia.org/wiki/I%C2%B2C>. [Accessed 18 11 2019].
- [31] “I2C Background,” TOTALPHASE, 2019. [Online]. Available: <https://www.totalphase.com/support/articles/200349156-I2C-Background>. [Accessed 18 11 2019].
- [32] “Jean Baptiste Joseph Fourier,” Mathshistory, 1 1997. [Online]. Available: <http://mathshistory.st-andrews.ac.uk/Biographies/Fourier.html>. [Accessed 18 11 2019].
- [33] “Joseph Fourier,” Wikipedia, 12 11 2019. [Online]. Available: https://en.wikipedia.org/wiki/Joseph_Fourier#cite_note-The_MacTutor_History_of_Mathematics_Archive-4. [Accessed 18 11 2019].
- [34] J. Stillwell, “Logic and the philosophy of mathematics in the nineteenth century,” Ten, C. L, vol. 7, p. 204, 2013.
- [35] P. G. Lejeune-Dirichlet, “On the convergence of trigonometric series which serve to represent an arbitrary function between two given limits,” für die reine und angewandte Mathematik , p. 157–169, 1829.
- [36] D. Mascré and B. Riemann, “Posthumous Thesis on the Representation of Functions by Trigonometric Series,” Elsevier , p. 49, 2005.
- [37] R. Remmert, “Theory of Complex Functions: Readings in Mathematics,” Springer, p. 29, 1991.

- [38] “Fourier series,” Wikipedia, 11 11 2019. [Online]. Available: https://en.wikipedia.org/wiki/Fourier_series. [Accessed 18 11 2019].
- [39] “The Fourier Transform,” 2011, [Online]. Available: <http://www.thefouriertransform.com/transform/fourier.php>. [Accessed 18 11 2019].
- [40] “Fast Fourier transform,” Wikipedia, 16 11 2019. [Online]. Available: https://en.wikipedia.org/wiki/Fast_Fourier_transform. [Accessed 18 11 2019].
- [41] H. Zumbahlen, “Chap.8 Analog Filters,” in Linear Circuit Design Handbook, Newnes/Elsevier, 2008, pp. 8.1-8.3.
- [42] C. L. Clark, “Chap.4 Digital Filters,” in LabVIEW Digital Signal Processing and Digital Communications, McGraw-Hill, 2015, p. 61.
- [43] “Infinite impulse response,” Wikipedia, 12 10 2019. [Online]. Available: https://en.wikipedia.org/wiki/Infinite_impulse_response. [Accessed 18 11 2019].
- [44] “Low-pass filter,” Wikipedia, 24 10 2019. [Online]. Available: https://en.wikipedia.org/wiki/Low-pass_filter. [Accessed 18 11 2019].
- [45] “Lesson 6. Digital Filtering,” in LabVIEW Signal Processing Course Manual, National Instruments Corporation, 1997, p. 6.3.
- [46] “High-pass filter,” Wikipedia, 20 09 2019. [Online]. Available: https://en.wikipedia.org/wiki/High-pass_filter. [Accessed 18 11 2019].
- [47] “Band-pass filter,” Wikipedia, 18 04 2019. [Online]. Available: https://en.wikipedia.org/wiki/Band-pass_filter. [Accessed 18 11 2019].
- [48] “Band-stop filter,” Wikipedia, 21 09 2019. [Online]. Available: https://en.wikipedia.org/wiki/Band-stop_filter. [Accessed 18 11 2019].
- [49] C. B. Rorabaugh, in Digital Filter Designer’s Handbook, TAB Books Division of McGraw-Hill, Inc., pp. 65-99.
- [50] “Butterworth Filters,” 12 09 2018. [Online]. Available: <https://tttapa.github.io/Pages/Mathematics/Systems-and-Control-Theory/Analog-Filters/Butterworth-Filters.html>. [Accessed 18 11 2019].
- [51] J. Stensby, “SCRIBD,” 31 08 2011. [Online]. Available: <https://www.scribd.com/document/264586900/Chebyshev-analog-filter>. [Accessed 18 11 2019].
- [52] B. Mihai, “LabVIEW Modeling and Simulation of The Digital Filters,” ResearchGate,

2015.

- [53] L. D. Paarmann, "Chap.5 Chebyshev Type II Filters," in Design and Analysis of Analog Filters , Kluwar Academic , 2003, pp. 155-159.
- [54] "Inverse Chebyshev approximation," Matheonics Technology inc., 2009. [Online]. Available: <http://www.matheonics.com/Tutorials/InverseChebyshev.html>. [Accessed 18 11 2019].
- [55] "Sensitivity and specificity," Wikipedia, 21 10 2019. [Online]. Available: https://en.wikipedia.org/wiki/Sensitivity_and_specificity#cite_note-BMJ-6. [Accessed 18 11 2019].
- [56] "Accuracy and precision," Wikipedia, 20 10 2019. [Online]. Available: https://en.wikipedia.org/wiki/Accuracy_and_precision. [Accessed 18 11 2019].

Annexe

	Filters	Measurements							
		Threshold	TP	TN	FP	FN	Sen. (%)	Spec. (%)	Acc. (%)
Experiment1	Butterworth	265.7(20%)	1018	4489	1445	1326	43,43	75,65	66,53
	Chebyshev	292,8(20%)	825	4947	987	1519	35,19	83,37	69,73
	Elliptic	262,2(20%)	1025	4873	1061	1319	43,73	82,12	71,25
	Inv. Chebyshev	254,4(20%)	1027	4948	986	1317	43,81	83,38	72,18
Experiment2	Butterworth	232,9(33%)	538	7383	1004	2232	19,42	88,03	70,99
	Chebyshev	190,4(33%)	909	7027	1360	1861	32,82	83,78	71,13
	Elliptic	195,3(33%)	816	7235	1152	1954	29,46	86,26	72,16
	Inv. Chebyshev	169,9(33%)	902	7152	1235	1868	32,56	85,27	72,19
Experiment3	Butterworth	142.1(10%)	1701	5800	1744	1923	46,94	76,88	67,17
	Chebyshev	140,3(10%)	1494	5714	1830	2130	41,23	75,74	64,54
	Elliptic	158,0(10%)	1614	6076	1468	2010	44,54	80,54	68,86
	Inv. Chebyshev	167,8(10%)	1573	6138	1406	2051	43,41	81,36	69,05

Tableau A.1 L'IIR filtre les performances détaillées pour la détection des mouvements fœtaux avec différents seuils

	Filters	Measurements							
		Threshold	TP	TN	FP	FN	Sen. (%)	Spec. (%)	Acc. (%)
Experiment1	Butterworth	206	1270	4165	1769	1074	54,18	70,19	65,66
	Chebyshev	206	1117	4282	1652	1227	47,65	72,16	65,22
	Elliptic	206	1205	4355	1579	1139	51,41	73,39	67,17
	Inv. Chebyshev	206	1202	4626	1308	1142	51,28	77,96	70,40
Experiment2	Butterworth	206	666	7093	1294	2104	24,04	84,57	69,54
	Chebyshev	206	728	7203	1184	2042	26,28	85,88	71,09
	Elliptic	206	759	7329	1058	2011	27,40	87,39	72,49
	Inv. Chebyshev	206	681	7612	775	2089	24,58	90,76	74,33
Experiment3	Butterworth	206	1243	6231	1313	2381	34,30	82,60	66,92
	Chebyshev	206	1022	6222	1322	2602	28,20	82,48	64,86
	Elliptic	206	1240	6375	1182	2384	34,22	84,33	68,07
	Inv. Chebyshev	206	1319	6370	1174	2305	36,40	84,44	68,89

Tableau A.2. L'IIR filtre les performances détaillées pour la détection des mouvements fœtaux avec un seuil moyen