



المدرسة الوطنية المتعددة التخصصات
Ecole Nationale Polytechnique

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT D'ELECTRONIQUE

MEMOIRE DE FIN D'ETUDES
D'INGENIEUR D'ETAT EN ELECTRONIQUE

Sujet :

**Développement d'un système embarqué
d'aide aux personnes handicapées moteur,
basé sur l'exploitation du signal EEG
(Implémentation sur carte DSP)**

Présenté par :

Mohamed Rahid DJAFRI & Ahmed Rafik HANNACHE

Membres du jury :

H. BOUSBIA	Chargé de cours, Président	(E.N.P)
A. MOKRAOUI	Docteur d'Etat, Examinateur	(E.N.P)
A. BELOUCHRANI	Maître de conférence, Promoteur	(E.N.P)
R. SADOON	Chargé de cours, Co-promoteur	(E.N.P)

- Juillet 2005 -

Dédicaces

Je dédie ce modeste travail à mes parents Sou, Zou, Chafou et Wahibou ; à tous mes frères et à Chafika. Sans eux, je n'aurai jamais pu devenir celui que je suis.

Je vous dis merci

Rahid

Je dédie ce travail

à mes parents,

à mes grands parents,

à mes oncles et tantes Mohamed, Khebbab, Hamza, Khedoudja, Souad,

Halima, Yacine, Hichem et mon regretté oncle Brahim.

à tous mes amis, ils se reconnaîtront.

à toute ma famille.

à la mémoire de mon oncle Omar.

Rafik

Remerciements

Nous tenons à exprimer notre gratitude et notre profond respect à nos promoteurs Mr A. Belouchrani et Mr R. Sadoun pour leurs conseils avisés et leurs remarques pertinentes. Nous les remercions vivement pour les efforts et le temps qu'ils nous ont consacrés.

Nous témoignons aussi notre grande reconnaissance à Mr H. Bousbia pour l'honneur qu'il nous fait en présidant notre jury et à Mr A. Mokraoui qui a bien voulu juger notre travail.

Nous adressons également nos remerciements à Mr Haddadi pour l'aide précieuse qu'il nous a apportée.

Nous remercions également, tous les enseignants qui ont contribué à notre formation et tous ceux qui ont participé d'une façon ou d'une autre à l'aboutissement de ce projet.

Table des matières

Introduction générale.....	1
 1. Notions sur les ICO	
1.1. Introduction.....	3
1.2. L'observation de l'activité cérébrale.....	3
1.3. Les types d'activités mentales utilisées dans les ICO.....	4
1.3.1. Les ICO basées sur le conditionnement.....	5
1.3.1.1. Les variations lentes du potentiel cortical	5
1.3.1.2. L'oscillation de l'activité sensorimotrice.....	5
1.3.2. Les ICO basées sur les réponses évoquées.....	6
1.3.2.1. Les potentiels évoqués P300.....	6
1.3.2.2. La réponse liée à un stimulus visuel permanent.....	6
1.4. Les signaux EEG.....	7
1.4.1. Définitions et généralités.....	7
1.4.2. L'activité EEG.....	8
1.4.2.1. Les ondes bêta.....	8
1.4.2.2. Les ondes alpha.....	8
1.4.2.3. Les ondes thêta.....	9
1.4.2.4. Les ondes delta.....	9
1.4.3. Les électrodes.....	9
1.4.4. Le système 10-20.....	9
1.5. L'algorithme général d'une ICO.....	10
1.6. Conclusion.....	11

2 Développement et mise en œuvre du système

2.1. Introduction.....	13
2.2. L'approche de développement.....	13
2.2.1. Le principe général du système développé.....	13
2.2.1.1. Le dispositif d'excitation.....	14
2.2.1.2. La fonction visuelle du sujet.....	14
2.2.1.3. L'acquisition et la mise en forme du signal EEG.....	15
2.2.1.4. La détection et l'envoi de la commande.....	15
2.2.1.5. La gestion de la commande et du dispositif de pointage.....	16
2.2.2. Les outils de développement.....	16
2.2.2.1. Le bloc de traitement et de décision.....	17
2.2.2.1.1. L'approche logicielle.....	17
2.2.2.1.2. L'approche matérielle.....	18
2.2.2.2. Le dispositif de pointage.....	19
2.2.2.3. Le bloc analogique.....	19
2.3. Le développement et la mise en œuvre.....	19
2.3.1 Le traitement et la commande.....	19
2.3.1.1. Le principe.....	19
2.3.1.2. Le développement sur Simulink-MatLab.....	20
2.3.1.2.1. La simulation.....	20
2.3.1.2.2. Les tests avec un signal réel.....	22
2.3.1.2.3. L'utilisation des S-Functions.....	25
2.3.1.3. Développement pour la carte DSK 5402 sous CCS.....	25
2.3.1.3.1. Description de la carte.....	25
2.3.1.3.2. Le Code Composer Studio CCS.....	26
2.3.1.3.3. La mise en œuvre.....	26
2.3.1.3.4. L'élaboration du programme.....	28
2.3.1.3.5. Le programme.....	28
2.3.1.3.5.1. L'initialisation de la carte.....	28

2.3.1.3.5.2. L'initialisation et la configuration du port série...	30
2.3.1.3.5.3. L'initialisation du codec.....	30
2.3.1.3.5.4. La lecture de N échantillons sur l'entrée analogique	30
2.3.1.3.5.5. L'opération de bit reverse sur les adresses des données.....	31
2.3.1.3.5.6. L'application de la FFT sur N points.....	31
2.3.1.3.5.7. Le calcul du spectre.....	31
2.3.1.3.5.8. La détermination de l'indice de l'élément max	33
2.3.1.3.5.9. Le test et l'envoi de la commande sur le port série	33
2.3.1.3.6. Commentaires.....	33
2.3.2. L'interface utilisateur.....	34
2.3.2.1. Le bloc d'excitation.....	34
2.3.2.2. Le développement du curseur.....	35
2.3.2.3. Le bloc d'exécution.....	37
2.3.3. La conception de l'interface d'adaptation et de mise en forme.....	40
2.3.3.1. Le filtre.....	40
2.3.3.2. L'amplificateur d'instrumentation.....	41
2.3.3.3. L'amplificateur de gain.....	42
2.3.3.4. L'isolation galvanique.....	42
2.4. Conclusion.....	44
Conclusion générale.....	45
Références bibliographiques.....	47
Annexes.....	49

Introduction générale

''Can these observable electrical brain signals be put to work as carriers of information in man computer communication or for the purpose of controlling such external apparatus as prosthesis devices or space ships ?

Even on the sole basis of the present states of the art of computer science and neurophysiology, one may suggest that such a feat is potentially around the corner.''

Jacques Vidal, 1973

Depuis l'article publié par Jacques Vidal [1], la communauté scientifique n'a cessé de rechercher les voies et moyens pour exploiter l'activité encéphalographique dans le but de fournir un nouveau canal non musculaire pour l'envoi de messages vers le monde extérieur ; ce qui constitue l'interface cerveau ordinateur (ICO). Ces messages peuvent être utilisés aussi bien pour la commande que pour la communication.

Durant les 15 dernières années, des programmes de recherche [2] ont émergé et ont été portés par la nouvelle compréhension de la fonction cérébrale, par l'arrivée d'équipements informatiques avec une puissance de calcul appréciable, à des coûts de plus en plus réduits et par la prise de conscience du besoin d'autonomie des personnes handicapées. Ces programmes sont focalisés sur le développement de nouvelles technologies de la communication pour les individus ayant des troubles neuromusculaires sévères, tels que la sclérose en plaque, l'attaque au niveau du pédoncule cérébral -brainstem stroke-, les lésions de la moelle épinière, et qui ne peuvent donc pas utiliser les technologies conventionnelles. Le but immédiat est de proposer à cette catégorie sociale, une aptitude à communiquer de façon à lui donner le moyen d'exprimer ses désirs à son entourage, d'utiliser l'interface graphique d'un ordinateur, d'écrire sur des logiciels simples de traitement de texte, de commander un fauteuil roulant ou une quelconque prothèse. Les systèmes ICO actuels déterminent les intentions de l'utilisateur à partir d'une variété de différents signaux électrophysiologiques.

Ces signaux sont captés par des électrodes implantées ou pas. Ils sont traduits en temps réel, en commandes qui agissent sur l’affichage d’un ordinateur ou tout autre appareillage. Il est évident que l’utilisateur et le système ICO doivent être adaptés l’un à l’autre. [3]

L’activité mentale produit une variété de phénomènes qui peuvent être mesurés avec les capteurs adéquats, et qui présentent une utilité potentielle pour les ICO.

L’objectif de notre étude est l’exploitation des signaux électroencéphalogrammes (EEG) pour le développement d’une ICO basée sur les réponses évoquées par un stimulus visuel. Sa mise en application sera illustrée par le contrôle du déplacement d’un curseur de souris sur deux dimensions dans l’interface graphique de l’environnement Windows, ainsi que le contrôle du click. Notre mémoire est structuré en deux chapitres.

Dans le premier chapitre, nous aborderons des notions générales sur les interfaces cerveau machine et les signaux EEG.

Le deuxième chapitre sera consacré au développement du système, tout en expliquant notre approche et les outils de développement utilisés.

Nous conclurons par une synthèse de ce qui a été réalisé. Nous évoquerons également les perspectives à court et à long terme.

Notions sur les ICO

1.1. Introduction

Une interface cerveau ordinateur ou ICO est un système de communication ou de contrôle qui ne dépend pas des chemins de sortie normaux du cerveau vers les nerfs et les muscles périphériques [4]. Elle a pour entrées, les signaux électrophysiologiques résultant de l'observation de l'activité électrique du cerveau et pour sortie, l'une des actions effectuées par l'appareillage cible [5] .

Le sujet contrôle l'application cible en effectuant des activités mentales, qui sont associées à des actions. Celles-ci varient selon l'application.

L'association d'activités mentales (AM) à des actions, nécessite la sélection d'un ensemble d'AM de contrôle auxquelles l'ICO va répondre et l'identification de signatures dans l'activité du cerveau qui caractérise chaque AM. Ces signatures sont identifiées par l'analyse des signaux EEG enregistrés pendant l'exécution des activités mentales précitées [5].

1.2. L'observation de l'activité cérébrale

L'activité cérébrale produit une variété de phénomènes qui peuvent être mesurés avec les capteurs adéquats et présentent une utilité potentielle pour une ICO.

Parmi les méthodes actuelles d'observation de l'activité mentale, nous citons : la mesure de potentiels électriques (électroencéphalogramme et méthodes électrophysiologique

invasives), l'imagerie par résonance magnétique (IRM), le magnétoencéphalogramme (MEG) [5].

Les systèmes à base d'IRM et de MEG sont volumineux et requièrent un environnement magnétiquement blindé. Les mesures de potentiels électriques constituent pour l'instant le choix le plus convenable pour sonder l'activité du cerveau pour des ICO.

Les changements observables dans ces signaux électriques, résultent du fait que le sujet effectue une activité mentale donnée ; et mis à part les méthodes invasives, il est facile d'acquérir le matériel nécessaire et de le mettre en oeuvre.

L'électroencéphalogramme enregistré au niveau du cuir chevelu constitue un choix intéressant pour les systèmes à ICO, grâce à sa méthode non invasive d'acquisition, sa relative simplicité et son faible coût. Nous concentrerons donc notre attention sur la conception et le développement d'ICO basées sur des électroencéphalogrammes enregistrés au niveau du cuir chevelu.

1.3. Les types d'activités mentales utilisées dans les ICO

Les types d'activités mentales utilisées dans les ICO actuelles ont été choisis selon les études sur les spécialisations hémisphériques qui suggèrent que les deux hémisphères du cerveau sont spécialisés dans différentes fonctions de cognition [5]. L'hémisphère gauche semble être spécialement sollicitée durant les fonctions analytiques et verbales, l'hémisphère droit l'est durant les activités spatiales et holistiques. Les activités mentales typiques sont : réponse évoquée par un stimulus extérieur, mouvement imaginaire d'un membre et opérations géométriques, arithmétiques ou verbales. Suivant le type d'activité mentale qu'elles utilisent, les ICO peuvent être classés en :

- ICO à Réponse Evoquée ;
- ICO basées sur le conditionnement de l'utilisateur.

Les ICO basées sur le conditionnement de l'opérateur réagissent à des activités mentales AM particulières (AM de contrôle) effectuées par le sujet. Les ICO à réponse évoquée

s'appuient sur l'attention focalisée du sujet sur des stimuli associés à des actions. Ces AM sont reconnues par l'ICO et continuellement relues [5].

1.3.1. Les ICO basées sur le conditionnement

Le sujet acquiert des aptitudes de contrôle par la pensée grâce à un apprentissage et à l'utilisation d'un feedback. Il doit apprendre à moduler son activité mentale dans le but que l'ICO exécute ses intentions. Le processus d'apprentissage peut prendre beaucoup de temps (jusqu'à plusieurs semaines). En même temps, l'ICO s'adapte à l'utilisateur par des algorithmes d'apprentissage. Donc l'opération s'appuie sur l'adaptation des deux parties : le sujet et la machine [5].

1.3.1.1. Les variations lentes du potentiel cortical

Les variations lentes du potentiel cortical -Slow Cortical Potential Shifts- (SCPS), durent de quelques centaines de millisecondes à quelques secondes et indiquent le niveau global d'excitation d'un réseau cortical et sont universellement présentes dans le cerveau humain. Les SCPS négatives sont associées au mouvement et autres activités provoquant une activité corticale alors que les SCPS positives sont liées à une diminution de l'activité corticale.

Cette activité est exploitée après un apprentissage du sujet par feedback. Elle lui permet de produire des signaux positifs ou négatifs pour un contrôle binaire [5].

1.3.1.2. L'oscillation de l'activité sensorimotrice

Les populations de neurones forment des réseaux complexes qui sont à l'origine d'une activité oscillatoire. Généralement, leur fréquence diminue avec l'augmentation du nombre d'ensembles de neurones synchronisés [5].

Une stimulation sensorielle, l'exécution d'un mouvement ou une image mentale, peuvent changer la connectivité fonctionnelle à l'intérieur du cortex et mener à la suppression

d'une amplitude (Event Related Desynchronisation : ERD), ou l'augmentation d'une amplitude (Event Related Synchronisation : ERS). Ces phénomènes apparaissent principalement pour les ondes μ (7-13 Hz) et les ondes β (>13 Hz). Par exemple la préparation à un mouvement de la main donne une courte désynchronisation au niveau des ondes μ et β [5].

1.3.2. Les ICO basées sur les réponses évoquées

Les réponses évoquées sont liées aux méthodes cognitives en psychologie qui considèrent l'esprit comme un outil de traitement de signal dont la sortie dépend de la relation entre les stimuli et l'activation de processus cognitifs.

Des événements visuels ou auditifs extérieurs provoquent, dans l'EEG, des signaux transitoires caractérisés par des modifications du potentiel. Ils sont connus sous l'appellation de *potentiels liés à un événement* (Event Related Potentials : ou ERP). Quand le sujet prête attention à un stimulus particulier, un ERP, temporellement lié à ce stimulus, apparaît dans l'EEG. Si des actions sont associées au stimulus, le sujet peut prendre le contrôle d'une ICO en concentrant son attention sur le stimulus correspondant à l'action désirée [5].

1.3.2.1. Les potentiels évoqués P300

Cette onde est une composante retardée d'un potentiel lié à un événement (excitation), qui peut être visuel ou auditif. Il possède un retard d'environ 300ms et est provoqué par un stimulus rare avec un niveau de surprise important ou significatif en terme d'énergie, quand celui-ci est mêlé à un stimulus fréquent [5].

1.3.2.2. La réponse liée à un stimulus visuel permanent

Un stimulus scintillant à une fréquence variant entre 2 et 90 Hz provoque *une réponse liée à un stimulus visuel permanent* -Steady State Visual Evoked Response- (SSVER) dans l'EEG. Cette réponse est caractérisée par une oscillation à la même fréquence que celle du stimulus ou à des multiples de cette fréquence. Donc, une SSVER peut être détectée en examinant le contenu spectral du signal enregistré dans le cortex visuel. Précisément au niveau des électrodes O1 et O2 du système international 10-20 que nous aborderons en 1.4.4.

Quand des actions sont associées à des objets scintillant à des fréquences différentes, le sujet peut contrôler l'ICO en fixant la cible correspondant à l'objet désiré.

La détection du stimulus fixé par le sujet, peut se faire malgré la présence simultanée des autres stimuli [6]

Les avantages de telles ICO résident dans le fait que peu ou pas d'apprentissage est requis pour un sujet afin de contrôler l'interface [7], Un autre avantage est qu'une seule électrode peut suffire, ce qui implique une mise en œuvre facile et une simplicité dans le traitement puisque nous avons un seul signal. Les avantages de cette dernière approche nous ont amenés à l'adopter.

1.4. Les signaux EEG

1.4.1. Définitions et généralités

L'électroencéphalogramme est un enregistrement de l'activité électrique du cerveau à partir du cuir chevelu. Les premiers enregistrements ont été effectués par Hans Berger en 1929 même si des études similaires ont été menées sur des animaux déjà en 1870 [8].

Les ondes enregistrées reflètent l'activité électrique à la surface du cerveau : le cortex. Celle-ci est influencée par l'activité des structures neuronales en dessous du cortex.

Les cellules nerveuses dans le cerveau produisent des signaux appelés *potentiels d'action*. Ces potentiels se déplacent d'une cellule à l'autre via un espace appelé *synapse*. Des médiateurs chimiques appelés *neurotransmetteurs* permettent au signal de passer à travers les synapses.

Il y a deux types de neurotransmetteurs. L'un permet au potentiel d'action de passer à la prochaine cellule nerveuse et l'autre l'empêche de passer à un autre neurone. Les neurotransmetteurs jouent un rôle d'aiguilleur. Le cerveau fournit beaucoup d'efforts pour

garder les bonnes quantités de chacun de ces neurotransmetteurs. C'est cette activité électrique qui est sondée lors d'un EEG [8].

La gamme de fréquence de l'EEG commence à 0.5 Hz et dépasse rarement les 70 Hz ou même les 40 Hz. En général, l'amplitude des ondes est inversement proportionnelle à leur fréquence. Cette amplitude varie de quelques microvolts à quelques centaines de microvolts [9].

1.4.2. L'activité EEG

L'activité EEG peut être classée en 4 bandes de fréquences : l'activité β ($>13\text{Hz}$), l'activité α (8-13 Hz), l'activité θ (4-7 Hz) et l'activité δ ($< 4\text{ Hz}$).

1.4.2.1. Les ondes β

L'activité β est l'activité normale présente quand les yeux sont ouverts ou fermés. Elle a tendance à apparaître dans les canaux enregistrés à partir du centre ou l'avant de la tête. Certaines drogues augmentent l'activité β (Fig.1).



Fig.1 : L'activité β [8]

1.4.2.2. Les ondes α

L'activité α est aussi une activité normale chez un sujet adulte éveillé. Elle est généralement enregistrée par les canaux qui sont placés à l'arrière de la tête et possède une amplitude variant de 40 à 100 μV . Elle est uniquement observée quand les yeux sont fermés et doit disparaître ou du moins diminuer quand les yeux sont ouverts (Fig.2).

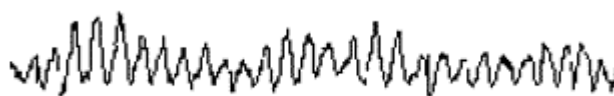


Fig.2 : L'activité α [8]

1.4.2.3. Les ondes θ

L'activité θ apparaît chez les enfants et les adultes somnolant, dans les autres cas elle indique un dysfonctionnement (Fig.3).



Fig.3 : L'activité θ [8].

1.4.2.4. Les ondes δ

L'activité δ est normale, uniquement pendant le sommeil profond chez l'adulte (Fig.4).



Fig.4 : L'activité α [8].

1.4.3. Les électrodes

Ce sont de petits disques métalliques qui sont placés sur le cuir chevelu à des positions normalisées. Elles peuvent être en acier inoxydable, en zinc, en or ou en argent plaqué avec une couche de chlorure d'argent [8].

1.4.4. Le système 10-20

Ce système fournit une normalisation internationale du placement des électrodes EEG sur le cortex humain (Fig.5).

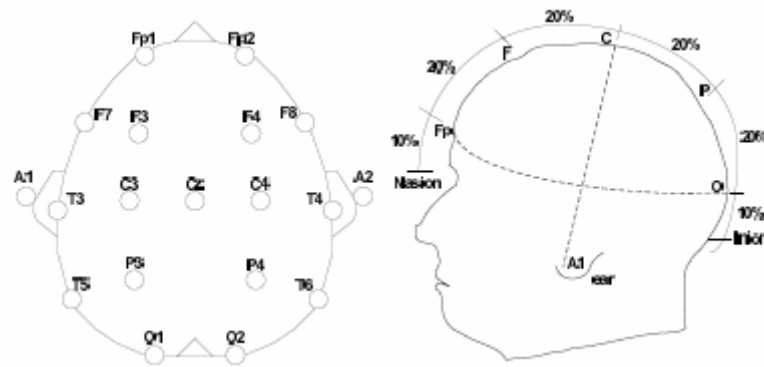


Fig.5 : Système international 10-20 [5].

Les nombres pairs indiquent les électrodes qui se trouvent sur l'hémisphère droit et les nombres impairs celles qui se trouvent sur l'hémisphère gauche.

Les lettres majuscules indiquent la zone frontale (F), la zone centrale (C), la zone pariétale (P), la zone temporelle (T), la zone occipitale (O), le pôle frontal (Fp) et les zones auriculaires (A) [5].

1.5. L'algorithme général d'une ICO

L'algorithme général d'une ICO est représenté à la figure 6.

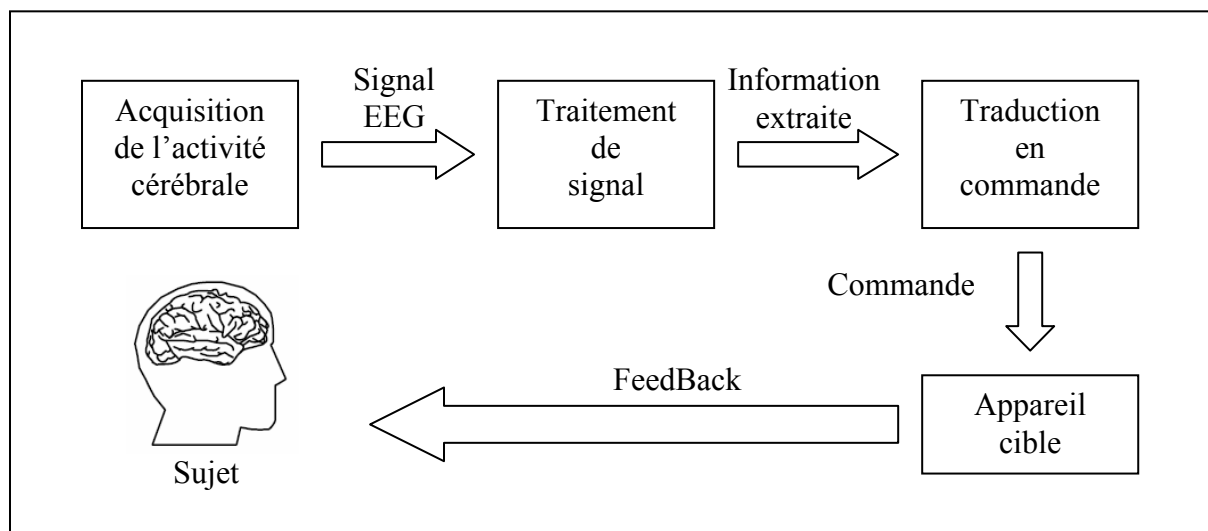


Fig.6 : Algorithme général d'une ICO.

L'observation de l'activité du cerveau du sujet fournit des signaux électriques qui sont analysés par le bloc de traitement de signal. Ce dernier envoie l'information extraite au bloc de traduction en commandes. A son tour, ce bloc envoie une commande, qui va générer l'action correspondante au niveau de l'appareil cible. Celle-ci, qui est observable par l'utilisateur, constitue un feedback qui lui permet de moduler son activité mentale pour obtenir le résultat désiré.

1.6. Conclusion

Après avoir introduit les ICO, et fait une distinction entre les différentes AM sur lesquelles elles se basent, nous avons présenté les signaux EEG et leurs caractéristiques générales. Car ces signaux sont à l'origine de telles interfaces.

Il ressort de ce chapitre que l'exploitation des SSVER présente une solution bien adaptée au temps de développement qui nous est imparti.

Développement et mise en œuvre du système

2

2.1. Introduction

Le document qui a suscité ce projet [7] présente une ICO expérimentale basée sur la détection des SSVER.

Ces réponses ont l'avantage d'être caractérisées par la présence de pics importants à la fréquence d'excitation [7].

Dans ce chapitre, toutes les étapes du développement du système seront détaillées. Nous commencerons par présenter notre approche de développement et les différents outils que nous avons utilisés. Ensuite, nous décrirons un par un les modules du système, en indiquant, chaque fois que cela est utile, l'interaction du module en question avec les autres.

2.2. L'approche de développement

2.2.1. Le principe général du système développé

Le but de notre travail est de concevoir un système qui procurera une aide aux personnes handicapées moteur. L'interactivité avec le système s'effectuera en temps réel et exclusivement par le biais d'une excitation oculaire en exploitant les SSVEP (fig.7).

A partir de ce schéma de principe (fig.7), nous pouvons concevoir un système gérant N commandes, équivalentes à N stimuli.

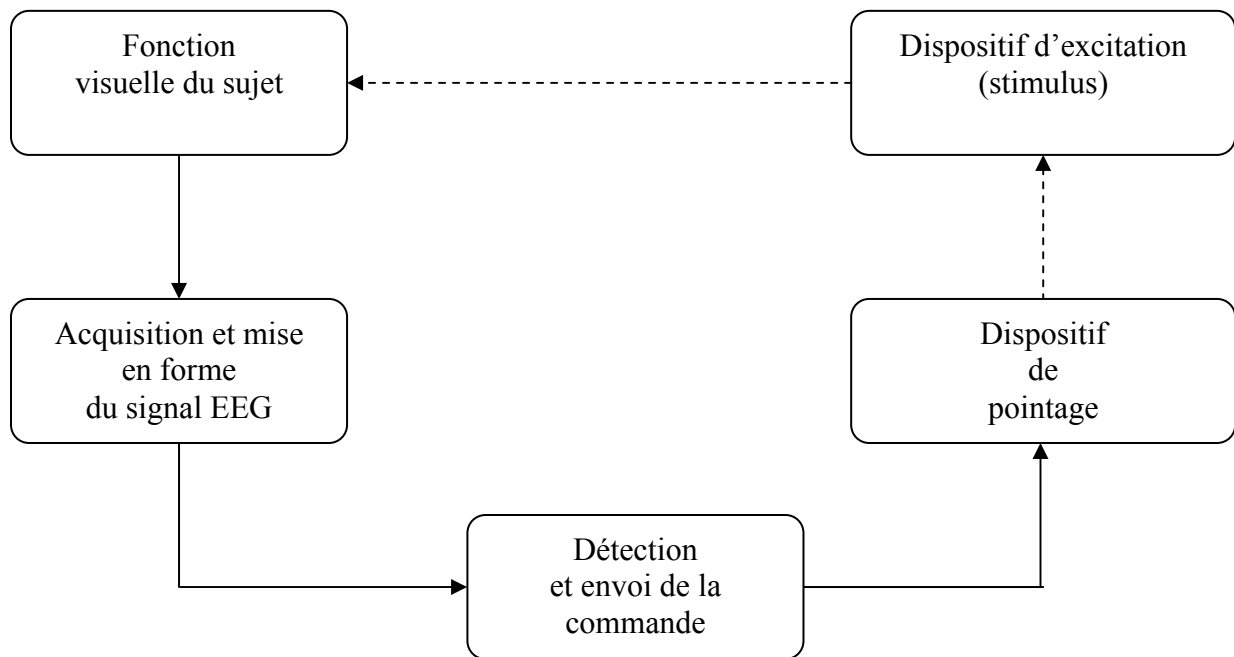


Fig.7 : Principe général du système à réaliser

2.2.1.1. Le dispositif d'excitation

Connaissant les caractéristiques de la réponse du cerveau à un stimulus visuel, ce module doit fournir une fréquence de clignotement qui varie en moyenne de 4Hz à 40Hz. Néanmoins, la bande dans laquelle on peut récupérer des signaux d'amplitudes exploitables (de l'ordre de 50 μ V) va de 6 à 20 Hz.

2.2.1.2. La fonction visuelle du sujet

L'utilisateur est considéré comme un module à part entière du système car nous allons exploiter sa fonction visuelle (excitation et réponse du cortex visuel). C'est ce même utilisateur qui fermera, pour ainsi dire, la boucle du système. Les traits en pointillés désignent l'interaction de l'utilisateur avec les autres blocs.

2.2.1.3. L'acquisition et la mise en forme du signal EEG

Le signal acquis via le capteur doit être amplifié à un niveau convenable. Le capteur sera constitué d'une sonde EEG fixée sur le cortex visuel (O2). Un gel spécial améliore l'adaptation entre le capteur et la peau. Un autre point (le lobe de l'oreille généralement : A2) sera pris comme masse pour permettre une mesure différentielle.

Un amplificateur d'instrumentation s'impose dans ce cas vu les caractéristiques du signal à traiter (tension différentielle de faible niveau).

Notons également qu'un filtre passe bas et de préférence passif pour ne pas introduire de bruit supplémentaire, placé en amont de l'ampli, contribuera à diminuer le bruit susceptible de causer des problèmes lors de l'amplification (intermodulation dû à la non linéarité des composants actifs, amplification de composantes fréquentielles indésirables, ondes électromagnétique captées par le fil de la sonde). Ce filtre permettra également de supprimer la raie à 50 Hz du réseau électrique et ses harmoniques, ainsi qu'une composante avoisinant les 50 Hz qui est quant à elle due à la fréquence de rafraîchissement du moniteur.

Une autre fonction de ce bloc est l'isolation galvanique. Elle doit fournir à l'utilisateur une protection contre tout risque d'accident d'ordre électrique.

2.2.1.4. La détection et l'envoi de la commande

Ce bloc est chargé de faire les traitements nécessaires sur le signal acquis, après sa mise en forme, pour détecter la fréquence d'excitation 'f' et activer le cas échéant le signal de commande correspondant. Une analyse spectrale s'impose ici. Une algorithmes rapide pour le calcul de la transformée de Fourier discrète -Fast Fourier Transform- (FFT) permet d'avoir le spectre du signal. Ensuite une procédure de décision permet de déclencher la commande. L'algorithme de Goertzel est une autre solution. Il permet un gain de temps car contrairement à une FFT, il ne calcule que la composante fréquentielle désirée. Un seuil ou une simple comparaison de niveau permet ou non de valider la commande. Signalons toutefois que c'est ce bloc qui est susceptible de poser le problème du temps réel.

2.2.1.5. La gestion de la commande et du dispositif de pointage

Pour le système à concevoir, la gestion du clignotement et l'exécution de la commande peuvent être gérées par un même processus.

Cette gestion étant programmable, il est possible de changer la correspondance commande/fréquence. Ceci permet de créer des menus comme ce qui se fait dans la téléphonie mobile. Une fréquence f permettrait par exemple de sélectionner un menu, cette même fréquence correspondrait à une autre commande dans le menu sélectionné. Cette gestion de la commande offre une grande souplesse à ce genre de système. En effet, il suffit de reprogrammer pour modifier le comportement voire la tâche du système.

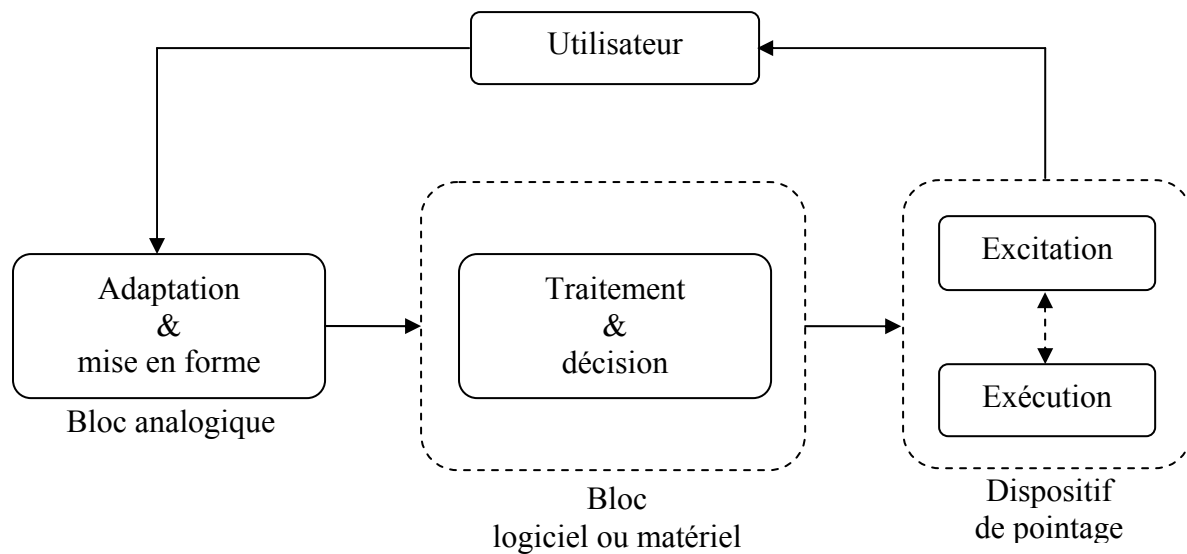
D'autres applications, hormis la notre, sont évidemment possibles telles que la commande d'un fauteuil roulant ou la synthèse vocale de mots ou de phrases significatives (demande d'aide, désir de manger ou de boire,...). Là, la seule limitation, est celle de l'imagination.

2.2.2. Les outils de développement

Il paraît clair que chaque bloc de l'algorithme, nécessite des approches et des outils de conception et de réalisation adaptés.

Pour développer notre système, conformément à l'algorithme général de la figure 7, nous avons opté pour deux solutions. Une approche logicielle qui permet de valider le concept et une approche matérielle qui permet d'avoir le système abouti. Il va de soi que pour avoir le produit fini, c'est-à-dire une solution embarquée du système, nous devons passer par des simulations pour valider la faisabilité du projet. Ceci pour dire que les deux solutions de développement vont de paire.

Le système à réaliser doit interagir avec l'utilisateur en temps réel. Cette contrainte influe considérablement sur l'orientation et le choix des outils de développement. Pour mieux cerner notre approche, il est nécessaire d'introduire l'algorithme suivant :

**Fig.8 : Structure du système**

Hormis l'utilisateur, le système est constitué de trois modules principaux (Fig.8). Cette subdivision en blocs est faite sur la base des outils de développement afin d'expliquer notre manière de procéder.

Pour le développement du système, le bloc central de traitement et de décision apparaît comme une vraie passerelle entre les autres blocs. C'est pourquoi nous avons commencé par développer ce bloc en premier. Nous nous sommes ensuite intéressés au système graphique gérant l'excitation et l'exécution, pour enfin arriver au dernier bloc, qui recueille et met en forme le signal.

2.2.2.1. Le bloc de traitement et de décision

Les deux approches logicielle et matérielle concernent seulement le bloc de traitement et de décision ; qui demande une certaine attention, vu que c'est à ce niveau que nous trouvons la contrainte de temps réel.

2.2.2.1.1. L'approche logicielle

Nous avons opté pour l'environnement de simulation et de développement Simulink et MatLab 6.5 de MathWorks. Cet environnement nous est familier, il intègre beaucoup de fonctionnalités du point de vue traitement de signal et interface système et il permet une relative rapidité de développement. D'autant plus qu'il nous permet de travailler en temps réel moyennant une bonne machine. La machine utilisée est de type COMPAQ EVO à base d'un processeur Intel P4 cadencé à 2GHz avec 256 Mo de RAM.

2.2.2.1.2. L'approche matérielle

La contrainte du temps réel étant imposée, nous avons le choix entre développer un système sur FPGA (Field Programmable Gate Array : circuit intégré re/programmable) ou bien un système à base de DSP (Digital Signal Processor : Processeur de traitement de signal). Nous avons retenu le deuxième choix pour les deux raisons suivantes :

- un système à base de FPGA nous aurait pris plus de temps pour l'étudier et l'élaborer alors que nous avons déjà travaillé sur les DSPs et leur environnement de développement ;
- des cartes de développement DSP étaient disponibles au niveau de notre laboratoire.

Quant au choix de la carte, nous avons le choix entre les cartes de développement suivantes :

- une carte DSP de Analog Device interne (PCI) ;
- une carte DSP de Texas Instruments de la famille C3x ;
- une autre carte de Texas Instruments de la famille C54.

Notre choix s'est porté sur la dernière carte, car elle satisfait mieux à l'idée de solution embarquée en comparaison avec la carte DSP de Analog Device, qui doit quant à elle être enfichée dans un port PCI d'un PC de bureau, ce qui est contraire à l'idée de système embarqué. La carte C3x a été écartée pour la non disponibilité d'outils de développement adéquat comparativement à la C5x.

La carte choisie est la TMS320VC5402 DSK de Texas Instruments, elle sera décrite dans 2.3.2.1. Il en sera de même pour son environnement de développement qu'est le Code Composer Studio V2.

2.2.2.2. Le dispositif de pointage

Pour développer le système graphique, nous avons opté pour le langage C++ sous l'environnement de développement Microsoft Visual C++. Celui-ci gère les API Windows (Application Programming Interface).

2.2.2.3. Le bloc analogique

Pour développer le bloc analogique, nous avons choisi ORCAD, qui intègre Capture, PSpice et Layout. Cet environnement bien complet, permet de concevoir les schémas des circuits électroniques, d'en simuler le fonctionnement et de tracer le circuit imprimé.

2.3. Le développement et la mise en oeuvre

2.3.1 Le traitement et la commande

Ce bloc gère la partie traitement du signal du système et pour des raisons pratiques, il inclut également la partie numérisation. Dans le dispositif final, ce bloc reçoit un signal analogique amplifié et filtré. Il en extrait les informations utiles qui déterminent la décision. Ensuite il traduit cette décision en commande reconnaissable par l'application cible.

2.3.1.1. Le principe

Le principe est relativement simple. Plusieurs stimuli visuels sont présentés au sujet. Il doit en fixer un. Chaque stimulus consiste en un scintillement à une fréquence prédéfinie. Le signal EEG recueilli contient une oscillation à la même fréquence, il contient également des harmoniques. Il suffit d'appliquer une transformée de Fourier au signal pour connaître l'énergie de ses composantes spectrales, et par conséquent, de déterminer quel est le stimulus fixé par le sujet.

2.3.1.2. Le développement sur Simulink-MatLab

2.3.1.2.1. La simulation

Dans un premier temps, le développement de ce bloc s'est fait sur un signal simulé, composé d'une sommation de 4 sinusoïdes (Fig.9).

Ce signal subit une FFT sur 1024 points, dont le résultat est exploité par un algorithme de traitement qui recherche les fréquences des sinusoïdes présentes à l'entrée. Il établit un seuil de décision dont la valeur est égale à deux fois la moyenne spectrale. Il extrait les raies spectrales correspondant aux sinusoïdes, leur fait subir un test préliminaire par rapport à un seuil afin d'ignorer les pics non significatifs puis compare celles qui dépassent le seuil. La composante fréquentielle la plus grande parmi celles testées sera choisie. Une variable en sortie nous renseigne sur la décision.

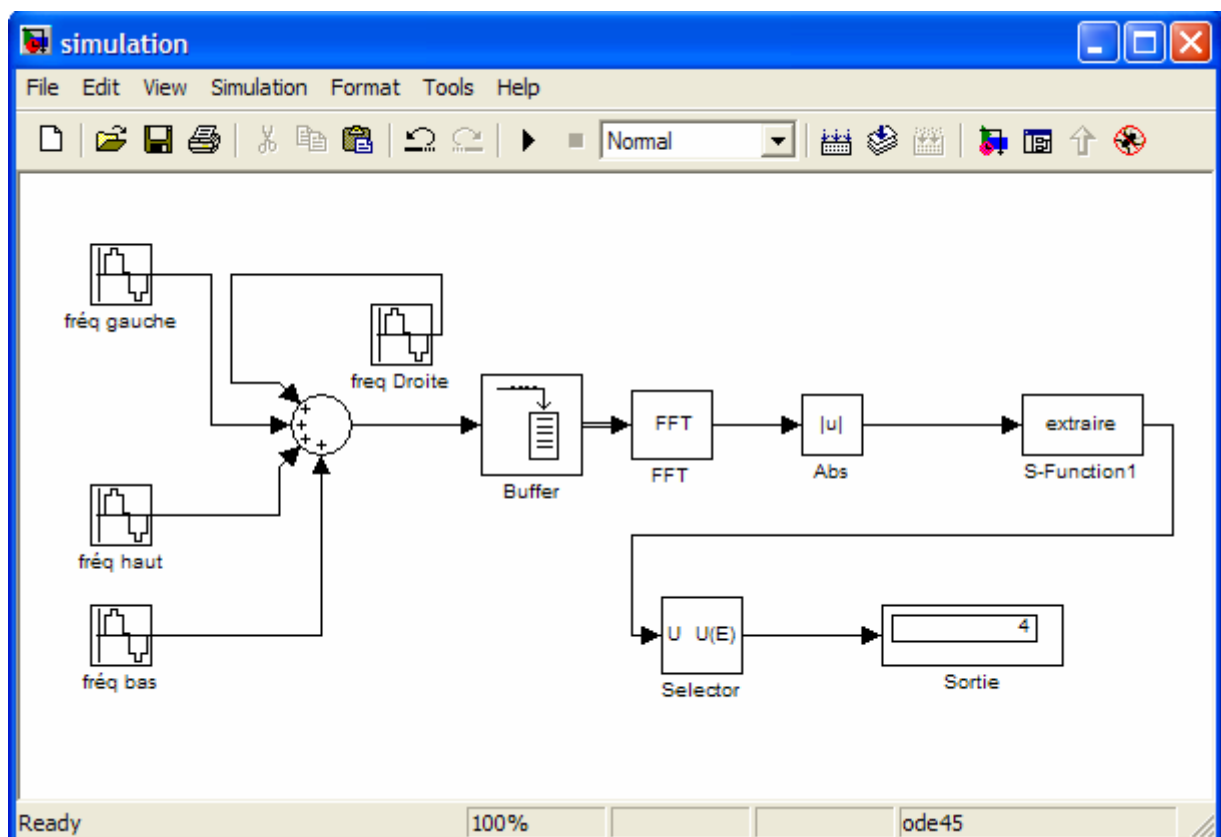


Fig.9 : Système de traitement et de décision développé sous Simulink, avec un signal simulé à l'entrée

Description

- Il y a quatre sinusoïdes numériques simulées à l'entrée du sommateur. Elles ont des fréquences respectives de 6, 7, 8 et 9 Hz. Chacune est échantillonnée à 200 Hz. La résolution spectrale du vecteur de sortie est de $200\text{Hz}/1024 = 0.195\text{Hz}$. Nous modulons leurs amplitudes à souhait, pour tester l'algorithme.
- Le 'buffer' ou tampon sert à mémoriser et à encapsuler un nombre donné d'éléments dans un vecteur qui servira d'argument d'entrée à la FFT (Fig.10).

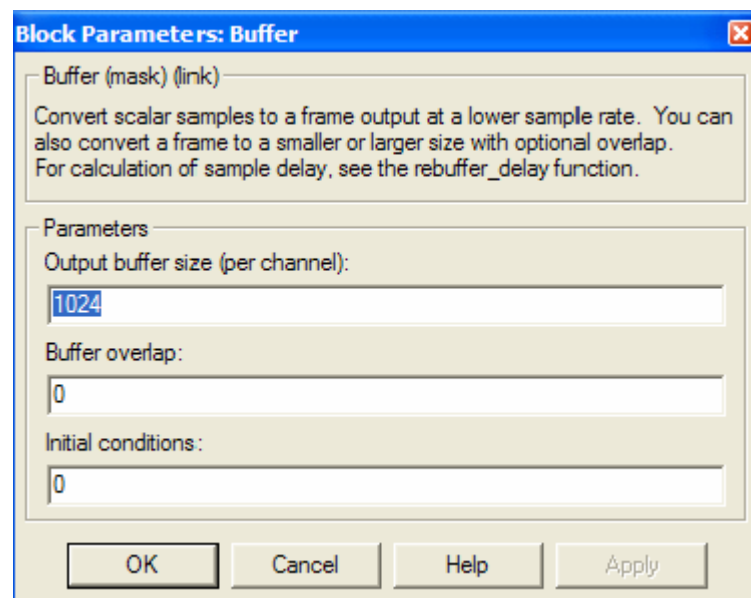


Fig.10 : Paramétrage du bloc buffer

- Le bloc FFT applique l'algorithme 'Fast Fourier Transform' au vecteur qui arrive sur son entrée. Il suppose que le signal comporte des échantillons temporels et tire les informations sur la fréquence d'échantillonnage automatiquement.
- Le bloc *extraire* est une fonction que nous avons développée. Elle recherche les composantes fréquentielles voulues, vérifie le niveau de leurs amplitudes, puis les compare et opte pour la plus grande (Fig.11).

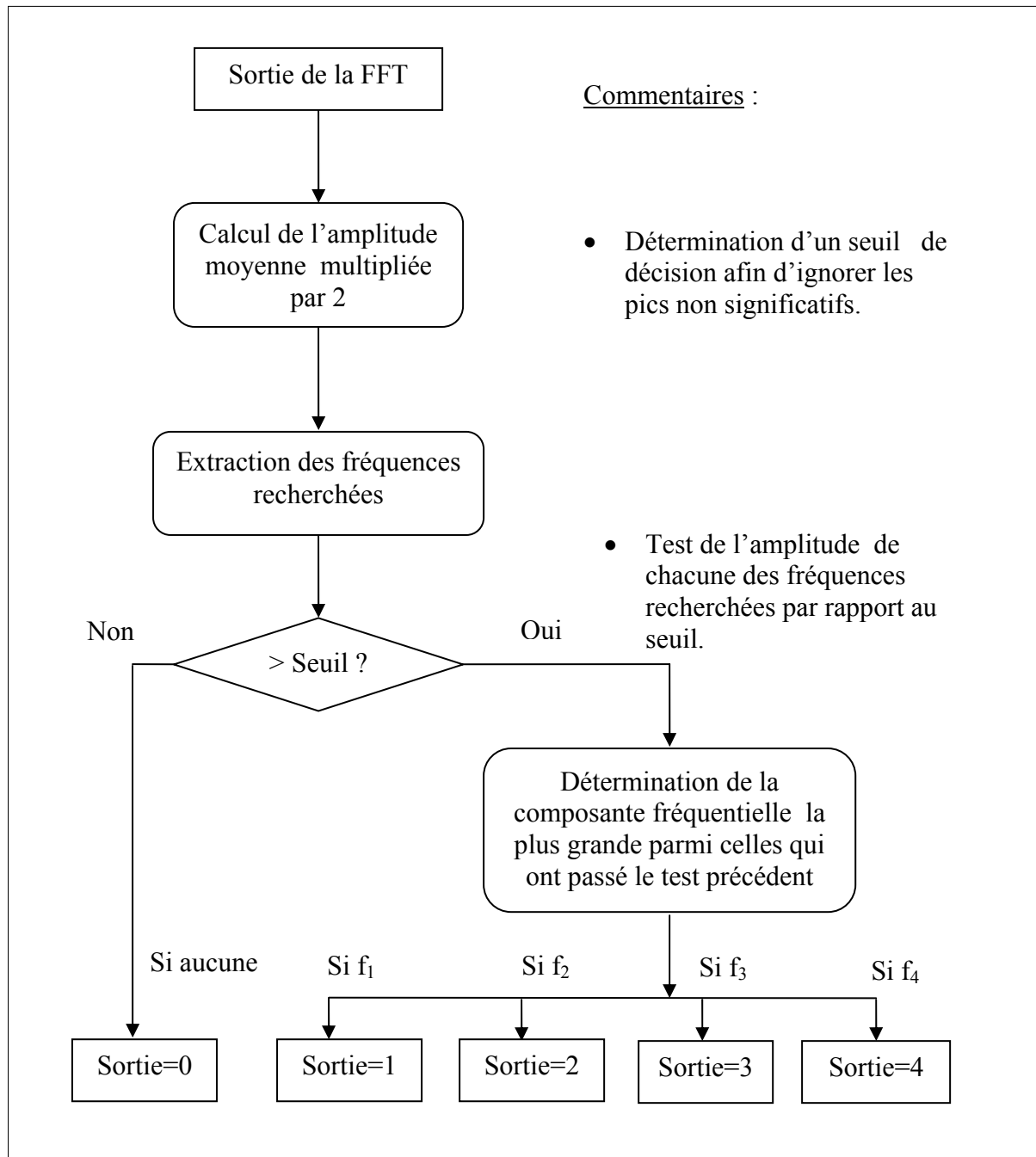


Fig.11 : Algorithme de la fonction extraire

2.3.1.2.2. Les tests avec un signal réel

Ce même programme sera étendu à une acquisition analogique via la carte son du PC, la numérisation étant paramétrée à l'aide de fonctionnalités de Simulink que nous verrons plus loin. Le même algorithme est utilisé pour le traitement du résultat de la FFT.

A la sortie du programme de traitement, Nous ajoutons une partie de traduction. Elle traduit la décision en commande spécifique à l'application cible. Cette partie gère l'écriture d'une commande correspondant à la décision sur le port série (Fig.12).

La liaison série a été choisie dans le but d'une communication avec l'application cible qu'est le curseur de souris. Celui-ci sera commandée par les décisions de ce programme et donc par le signal analogique soit directement par le cerveau.

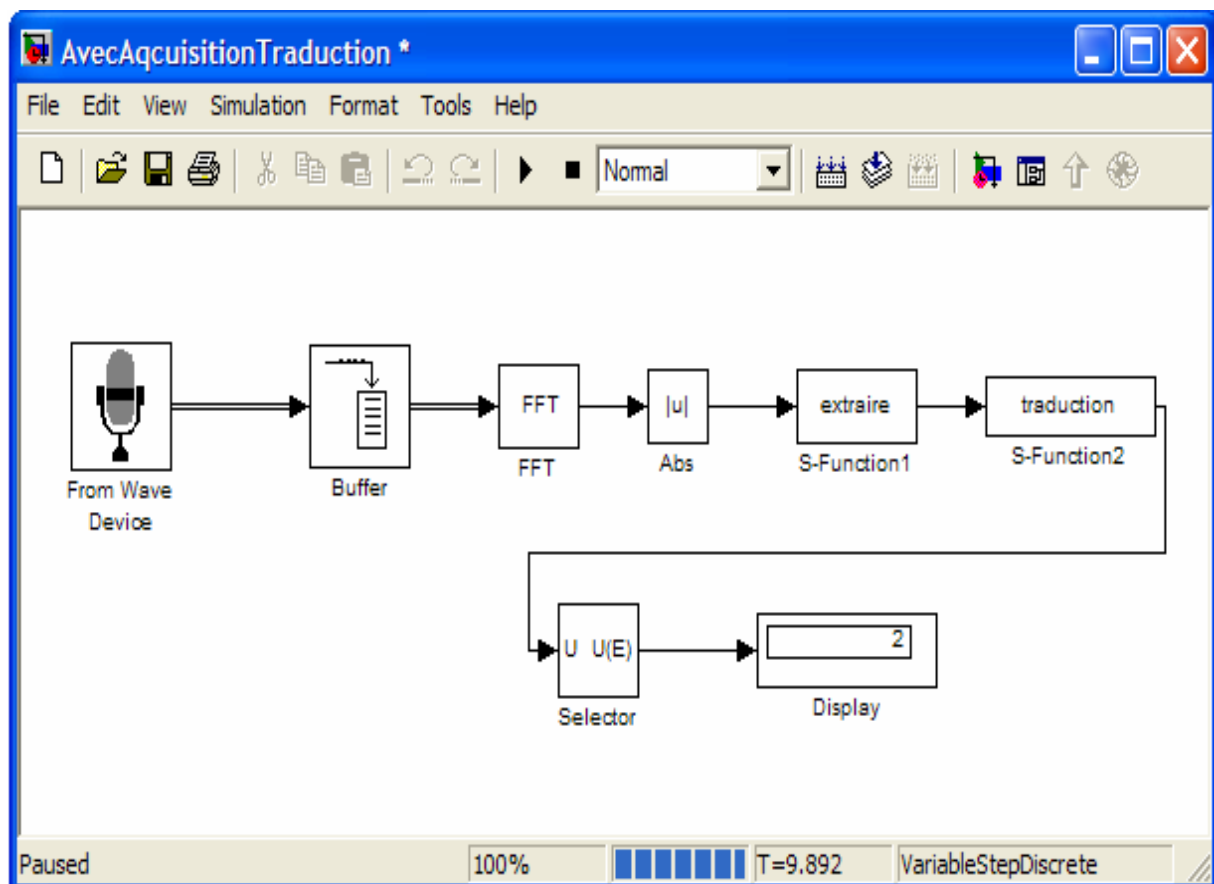


Fig.12 : Système de traitement et de décision développé sous Simulink, avec un signal réel à l'entrée et la traduction en commande

Description

- A l'entrée, un signal analogique est acquis par la carte son. Les paramètres de l'acquisition sont configurés grâce au bloc 'From Wave Device' de Simulink (Fig.13).

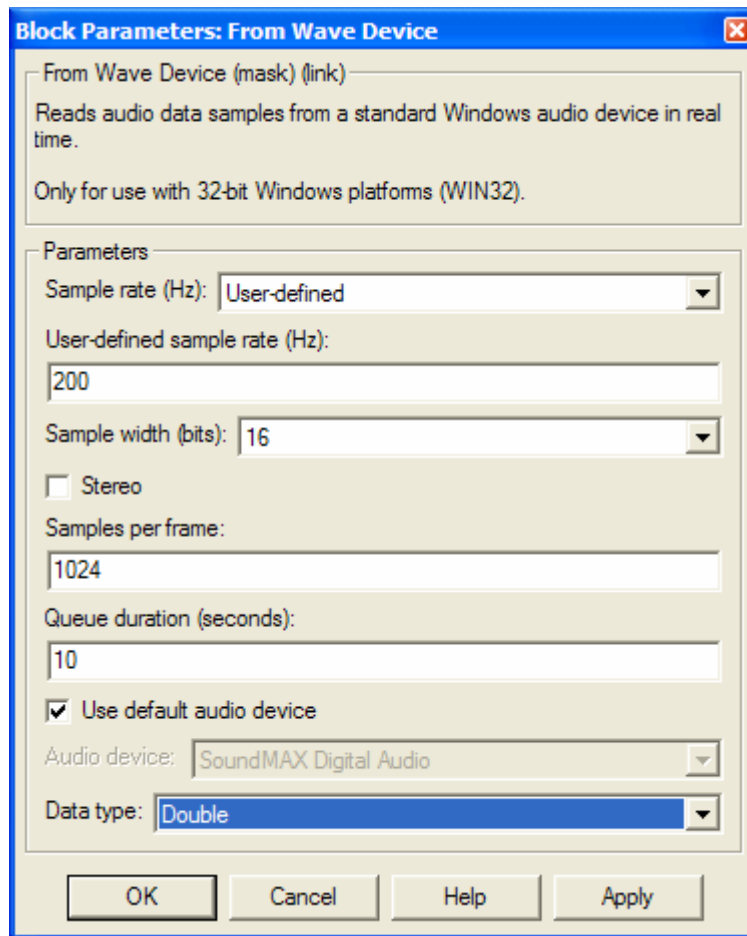


Fig.13 : Configuration des paramètres de l'acquisition

Paramètres

1. fréquence d'échantillonnage ;
2. codage des échantillons en mémoire ;
3. nombre d'échantillons par trame ;
4. durée de l'exécution ;
5. type de donnée.

- La traduction en commande sur le port série est contenue dans le bloc traduction. Elle comporte trois étapes :

1. numérisation ;
2. extraction de l'information ;
3. traduction de l'information en commande.

2.3.1.2.3. L'utilisation des S-Functions

L'environnement Simulink est composé de bloc prédéfinis, mais pour développer un programme spécifique, il est nécessaire de l'écrire en langage de programmation, comme le langage MatLab dont les codes sources sont sauvegardés dans des fichiers nommés m-files. Un fichier m-file ne peut être ni lu ni exécuté par Simulink. Il est possible de régler ce problème en compilant le fichier m-file en question, avec le compilateur de Matlab, en S-function. Sachant qu'une S-function est reconnaissable par Simulink. Cette compilation se fait par l'instruction suivante : *mcc -S file*

Il faut que le fichier '*.m' à compiler soit une fonction (qui possède une entrée et une sortie) et non pas un simple script MatLab sinon, la compilation s'arrêtera avec un message d'erreur.

L'instruction précédente génère une S-function dans un fichier source du type *file_simulink.c*.

Ensuite, dans notre environnement Simulink, il suffit d'ajouter un bloc appelé S-function, et de spécifier dans ses paramètres la S-function que nous souhaitons utiliser, en l'occurrence, celle que nous venons de compiler.

2.3.1.3. Développement pour la carte DSK 5402 sous Code Composer Studio

2.3.1.3.1. Description de la carte

La carte TMS320VC5402 DSK (Developer Starter Kit) (annexe 1) est relativement peu chère. Elle permet de développer des applications pour les DSP de la famille 54x. Elle est destinée au domaine de la parole et possède [10]:

- le DSP VC5402 qui peut être cadencé jusqu'à 100 MHz ;
- une mémoire externe SRAM de 64K mots ;
- une mémoire FLASH de 256K mots ;
- un contrôleur de bus JTAG pour l'émulation ;

- un port HPI host port interface pour la connexion à un ordinateur via le port parallèle ;
- une interface téléphonique analogique avec un connecteur RJ11 ;
- une interface audio dotée d'entrée/sortie pour connecter un Microphone/Haut parleur via des prises jack standard de 3.5mm et du codec audio AD50 ;
- une interface de données asynchrone RS-232 ;
- une interface pour carte d'extension (mémoire ou périphérique) ;
- des régulateurs linéaires qui génèrent les tensions de 1.8, 3.3 et 5V à partir du 5V de l'alimentation externe connectée à la prise jack de 2.5mm ;
- quatre indicateurs LED (3 contrôlables par l'utilisateur + 1 témoin de mise sous tension) ;
- huit interrupteurs DIP Switches, dont deux sont à définir par l'utilisateur ;
- un bouton de réinitialisation (Reset Manuel) ;
- des cavaliers (jumpers) pour la configuration de la carte.

2.3.1.3.2. Le Code Composer Studio CCS

Le CCS est un environnement de développement et de simulation très élaboré. Il permet d'exécuter les programmes C/C++ ou assembleur, sur des simulateurs de DSP de la famille C5x ou bien directement sur la carte.

Nous avons utilisé les deux options pour les raisons suivantes : le simulateur est facile à manipuler, ne plante pas, il n'y a pas d'erreur lors de la transmission ou du chargement du programme vers la mémoire, ce qui n'est pas toujours le cas pour la carte.

Une fois le programme mis au point sur le simulateur, nous l'avons modifié de sorte qu'il soit exécutable sur la carte, en ajoutant les bibliothèques et headers adéquats. Cela n'a pas fonctionné bien évidemment dès la première fois.

2.3.1.3.3. La mise en oeuvre

Pour développer une application sur CCS, au moins un fichier source en C ou en assembleur doit être ajouté au projet. Le projet porte l'extension .pjt . Des bibliothèques doivent être

ajoutés si des fonctions spéciales sont requises. Les librairies que nous avons utilisées sont :

- Rts.lib : cette librairie est nécessaire car elle fournit la routine boot.asm, qui permet le lancement du programme.
- Dsk5402.drv et Dsk5402.lib : Ces librairies sont nécessaires pour le bon fonctionnement du programme sur la carte DSK. Elles fournissent aussi un certain nombre de fonctions pour gérer les périphériques de la carte.
- 54xDSP.lib : cette librairie fournit des fonctions de traitement de signal optimisées. Elles sont entièrement écrites en assembleur. Elles peuvent être appelées du C.

Les fichiers header .h sont des librairies très utiles. Ils contiennent quantité d'informations et d'indications sur les fonctions et leurs arguments. Ces librairies spécialisées rendent aisée l'exploitation de la carte et fournissent un gain de temps appréciable.

Le fichier de lien dont l'extension est .cmd contient la répartition de la mémoire. Il permet de placer les sections de programmes aux emplacements mémoires désirés et de procéder à l'alignement des données.

Remarque importante concernant l'utilisation de la FFT

La bande de fréquences qui nous intéresse est celle de 4 à 20 Hz. La résolution en fréquence est définie comme suit :

$$\text{résolution} = F_s / N \quad \text{où : } F_s \text{ est la fréquence d'échantillonnage et } N \text{ est le nombre de points de la FFT.}$$

N est limité à 1024. Car la librairie ne fournit qu'une FFT de 1024 points au maximum. Alors pour augmenter la résolution, nous sommes obligés de diminuer la fréquence d'échantillonnage. Là aussi nous sommes limités par le codec, qui ne peut pas descendre en-dessous de 2KHz. Ce qui donne une résolution d'environ 2 Hz.

Pour améliorer cette résolution, nous pouvons envisager d'augmenter le nombre de point de la FFT en la programmant nous même, mais cela pose les problèmes suivants :

- La programmation sous C n'est pas optimale, et risque de compromettre l'exécution en temps réel. La programmation sous assembleur est optimale mais nécessite un temps de développement important, non permis par nos échéances.
- Même en acceptant les contraintes précitées, une FFT de 2048 ou 4096 points consommerait beaucoup de mémoire et de temps de calcul pour une résolution améliorée d'un facteur de 4 au mieux. L'exécution en temps réel d'une telle FFT, même développée en assembleur, risque d'être décevante.

2.3.1.3.4. L'élaboration du programme

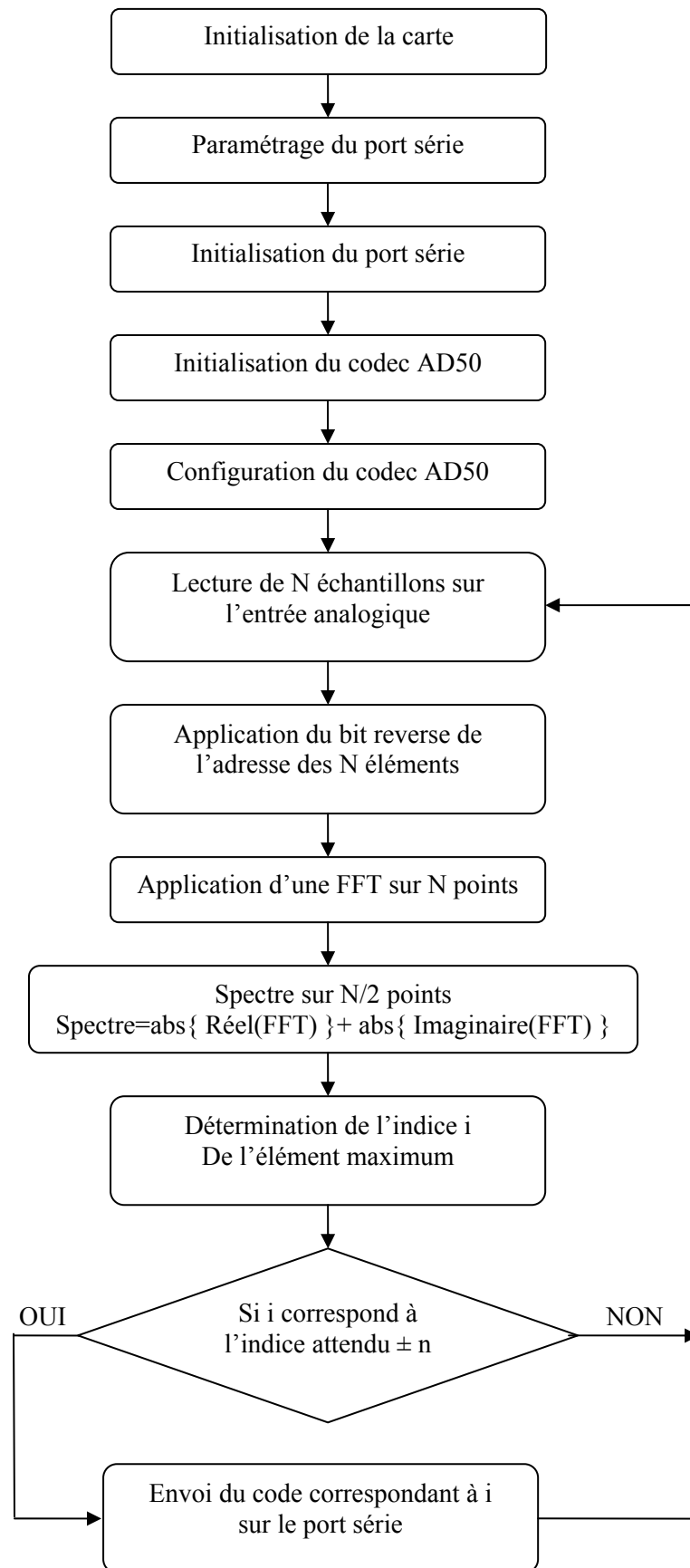
Avec le même principe général que celui utilisé dans la validation sous simulink, nous avons développé un programme qui s'adapte aux contraintes de la carte DSK (Fig.14).

2.3.1.3.5. Le programme

L'algorithme proposé a été implémenté sur la carte DSK, et a donné des résultats satisfaisants. Nous avons établi une liaison série avec un PC sous une session d'Hyper Terminal, nous avons connecté un GBF à l'entrée analogique de la carte, puis nous avons prédéfini quatre fréquences de commande. En calant le GBF sur ces fréquences, nous recevions sur l'écran de l'Hyper Terminal les commandes associées. Ces fréquences allaient de 6Hz jusqu'à 20Hz. Le critère de temps réel était satisfait. Voici maintenant le détail de notre programme :

2.3.1.3.5.1. L'initialisation de la carte

Une fonction d'initialisation fournie par Texas Instrument (TI) permet d'initialiser la carte pour être utilisée et fixe la fréquence de travail du DSP. Cette fréquence peut être 20, 30, 40, 50, 60, 70 ou 100MHz. Nous avons mis la valeur maximale pour satisfaire la contrainte du temps réel.

**Fig.14 :** Algorithme de traitement implémenté sur la carte DSK

2.3.1.3.5.2. L'initialisation et la configuration du port série

Une fonction d'initialisation fournie par TI permet d'initialiser le port série et de le configurer avec les valeurs par défaut. Pour établir une connexion série, les deux équipements doivent avoir la même configuration. La configuration choisie est la suivante :

- vitesse de connexion : 9600 BPS ;
- bits de données : 8 ;
- bits de stop : 1 ;
- parité : paire ;
- contrôle de flux : matériel ;
- bouclage : désactivé.

Ces valeurs sont chargées avant de lancer l'initialisation du port série. De ce fait, le port est initialisé avec les paramètres voulus.

2.3.1.3.5.3. L'initialisation du codec

Une fonction d'initialisation fournie par TI permet d'activer le codec et de lui affecter un 'handle'. Une fois le handle obtenu, la configuration du codec se fait par des fonctions de la librairie. Les paramètres suivants ont été utilisés :

- convertisseur ADC : en mode 15 bits ;
- gain du codec pour l'entrée analogique : 6 dB ;
- fréquence d'échantillonnage : 2 KHz.

2.3.1.3.5.4. La lecture de N échantillons sur l'entrée analogique

En premier lieu, nous avons utilisé une boucle pour remplir les éléments d'un tableau de dimension N. Dans les simulations, le signal acquis (une sinusoïde) présentait un phénomène d'escalier. Ceci était dû au fait que nous perdions des cycles d'horloges dans la boucle. Pour remédier à ce problème, nous avons trouvé une fonction qui peut acquérir directement N échantillons et les mettre en mémoire.

2.3.1.3.5.5. L'opération de bit reverse sur les adresses des données

L'algorithme de la FFT nécessite que les données lui soit transmises en adressage bit reverse. La fonction utilisée est tirée de la librairie 54xdsp.lib, c'est une fonction écrite en assembleur [11].

2.3.1.3.5.6. L'application de la FFT sur N points

La fonction utilisée est la rfft1024. C'est une FFT écrite en assembleur pour des signaux réels. Ces signaux doivent être introduit en adressage bit reverse. La fonction fait du calcul sur place, c'est-à-dire qu'elle écrase le signal fourni en argument. La sortie quant à elle est sous la forme : $\text{Re}(0) \text{Re}(512) \text{Re}(1) \text{Im}(1) \dots \text{Re}(512) \text{Im}(512)$

La fréquence $512 = F_s/2$ correspond à la fréquence de nyquist.

Nous avons rencontré un problème lors de la mise en œuvre de la FFT. En effet, tout était bien écrit et bien agencé mais les résultats de la FFT étaient tout simplement faux. Nous n'avons pas trouvé de solution à ce problème dans la documentation. Après une recherche sur le net, nous avons trouvé un excellent forum traitant des DSP [12], où nous avons pu déceler quelques bribes de réponses, mais la solution au problème a été trouvée sur le site internet de l'université de Stanford. [13]. Le problème venait de l'alignement des données en mémoire. Nous avons procédé à cet alignement (fig.15), et avons obtenu les résultats escomptés.

2.3.1.3.5.7. Le calcul du spectre

L'exécution de la FFT conventionnelle nécessite beaucoup de temps de calcul ($N/2$ (512) fois la racine carrée d'une somme de deux carrés), chose qui compromet le temps réel. Pour remédier à ce problème, nous avons proposé une autre formule moins gourmande en temps de calcul. Nous avons pensé à l'utilisation de la valeur absolue. De ce fait nous calculons la somme des valeurs absolues de la partie réelle et de la partie imaginaire. Le résultat obtenu n'est pas rigoureusement le spectre du signal, mais une sorte d'image distordue (verticalement, si on trace le spectre). Mais l'information qui nous intéresse n'est pas altérée. Les résultats obtenus par la suite le confirment.

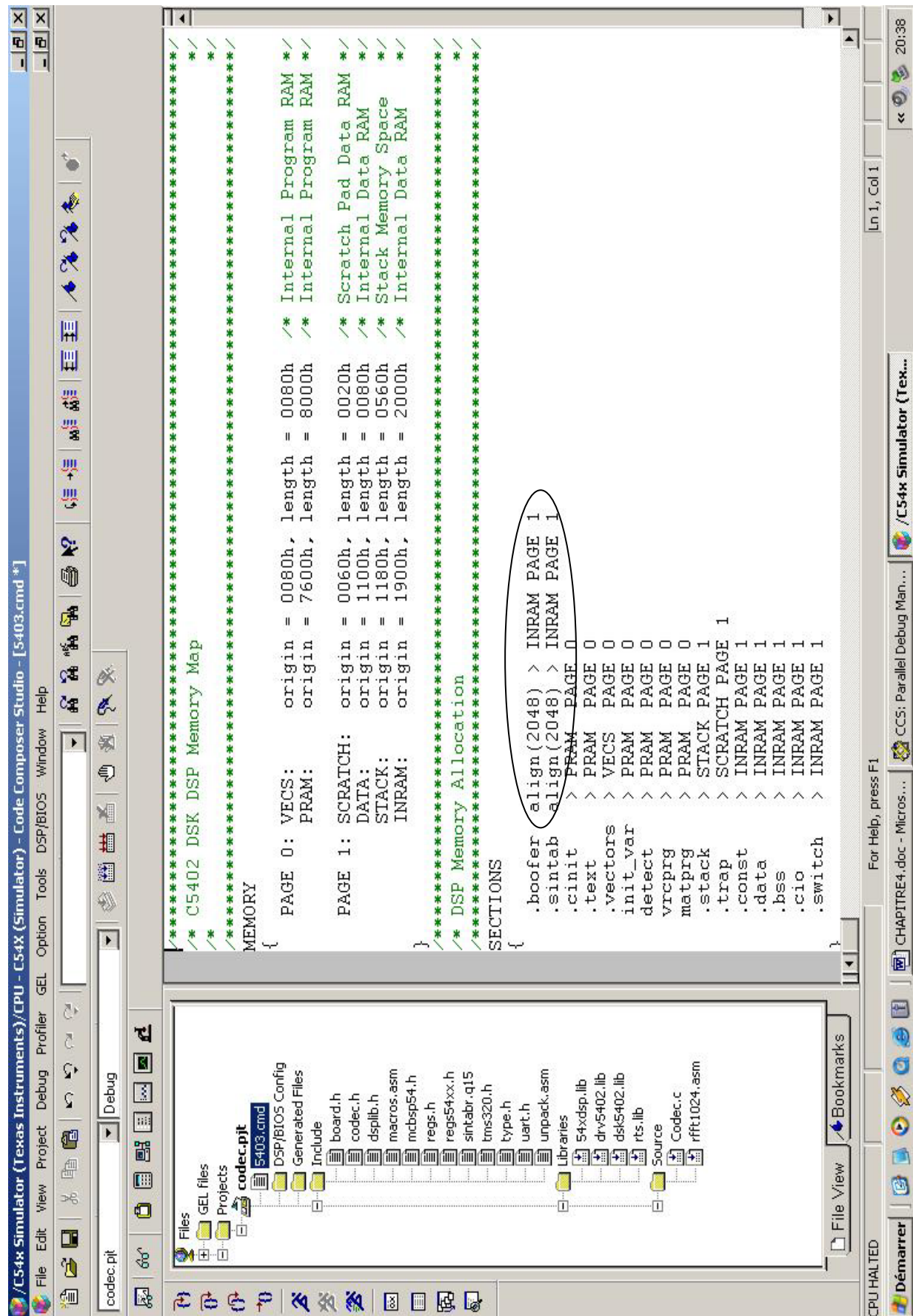


Fig.15 : Code Composer Studio

2.3.1.3.5.8. La détermination de l'indice de l'élément max

Encore une fois, dans le souci de satisfaire la contrainte du temps réel, nous avons utilisé une fonction écrite en assembleur et qui retourne l'indice de la valeur maximale d'un tableau. Pour déterminer la fréquence détectée, il suffit juste de multiplier l'indice obtenu par la résolution en fréquence qui est à peu près égale à 2Hz. Le n du schéma a été ajouté pour exprimer une tolérance dans la détection dans le cas où la résolution serait grande. La tolérance serait de n fois la résolution. Il suffit de changer le paramètre n

2.3.1.3.5.9. Le test et l'envoi de la commande sur le port série

Le test consiste à déterminer si l'indice détecté correspond bien aux indices attendus avec les incertitudes fixées par n . Si c'est la cas, un code est envoyé sur l'interface série.

2.3.1.3.6. Commentaires

- La résolution de 2Hz s'est avérée satisfaisante durant les tests que nous avons effectués avec la carte, en prenant comme signal d'entrée une sinusoïde fournie par un GBF et un oscilloscope pour visualiser le signal d'entrée. Le système donnait toujours des résultats satisfaisant même en descendant à l'amplitude limite, que fournit le GBF, de 5mV. Cela s'explique par le fait que les raies sont décelables du fait que la fenêtre temporelle n'est pas infinie. Donc au lieu d'avoir une simple raie, nous allons avoir des sinus cardinaux. Ce qui permet de détecter la présence d'une raie, surtout si le sinc a une base large, ce qui est le cas des fenêtres temporelles courtes.
- L'idéal serait d'avoir une fréquence d'échantillonnage de 200 Hz, ce qui permettrait d'avoir une bonne résolution - 0.2 Hz-.
- Le câble série permettant de connecter la carte à un ordinateur n'a pas été fourni avec la carte et il n'est pas disponible sur le marché. Nous avons été contraint d'en fabriquer un. La documentation de la carte [10] indique que la liaison utilise le protocole RS232 mais n'indique pas que la carte doit être considérée comme un ETCD (Equipement terminal de communication de données), et non pas un ETTD

(Equipement terminal de traitement de données). Chose que nous avons découverte après de multiples tests, et exercices de soudure et de dessoudures.

2.3.2. L'interface utilisateur

Le stimulus visuel et l'exécution des commandes envoyées par le bloc de traitement via la liaison série sont gérés par un programme développé en C++ pour l'environnement Windows.

L'interface utilisateur est un curseur de souris qui contient un stimulus visuel et qui se déplace sur l'écran en fonction des commandes arrivant dans le port série du PC.

2.3.2.1. Le bloc d'excitation

Ce bloc génère l'excitation visuelle. Selon le principe de fonctionnement de notre système, l'excitation doit venir d'un clignotement visible à fréquence constante.

L'utilisation d'une seule fréquence est suffisante pour un système à commande unique, par exemple un système tout ou rien (on / off).

Dans le cas de notre application, il s'agit de déplacer un curseur de souris sur l'écran d'un ordinateur et de cliquer à l'endroit désiré. Cette application doit recevoir au moins 5 commandes correspondant au :

- déplacement à droite ;
- déplacement à gauche ;
- déplacement vers le haut ;
- déplacement vers le bas ;
- click (bouton gauche).

L'excitation doit donc comporter 5 fréquences différentes.

2.3.2.2. Le développement du curseur

Pour le développement du curseur, nous avons procédé comme suit :

- Création d'une fenêtre qui contient 5 carrés de couleurs différentes (Fig.16).

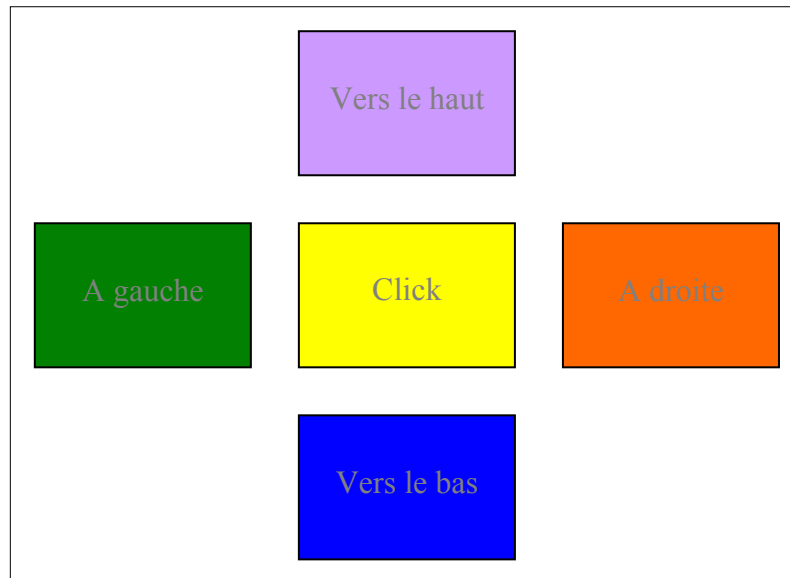


Fig.16 : Curseur contenant le stimulus visuel

- Définition de cinq minuteries (timers) avec des périodes définies en millisecondes. Nous définissons ces périodes de façon à obtenir les fréquences d'excitation voulues.
- Association de chaque timer à l'affichage d'un carré, ce qui fait que chaque carré clignote à l'une des fréquences définies.

Le programme développé est décrit par l'algorithme illustré à la figure 17.

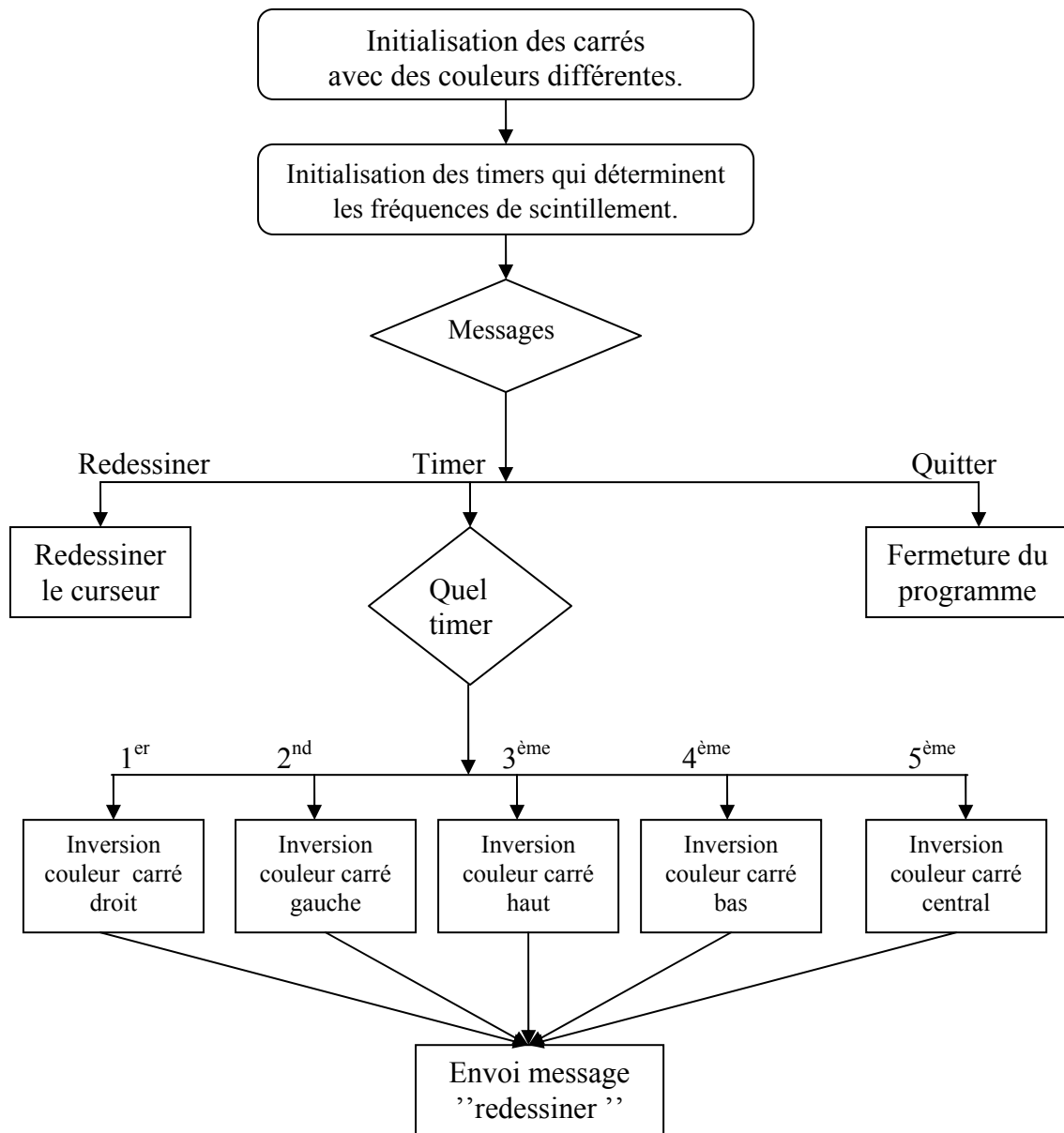


Fig.17 : Algorithme de génération du stimulus visuel

Remarques

1. Les couleurs des carrés n'ont a priori pas de rôle dans l'excitation. Néanmoins, chaque couleur provoque une réponse évoquée différente au niveau du cerveau [14].

2. Les fréquences citées ci-dessus sont celles qui seront sondées au niveau du bloc de traitement.
3. Il faut noter que la programmation sous Windows (et sous interface graphique en général), est événementielle et non pas séquentielle. Le programme se contente de répondre à des messages (événements) qu'il reçoit.

Commentaires

- L'expression ''Inversion de la couleur'' signifie que la couleur du carré doit basculer d'une couleur prédéfinie à la couleur de fond de la fenêtre et vice versa.
- Chaque fois que 'c'est l'heure' pour un timer, le système d'exploitation envoie un message pour l'indiquer. A cet instant la couleur du carré correspondant doit être inversée pour donner un scintillement cadencé par ce timer.
- Pour que ce changement de couleur soit effectif, il faut que la fenêtre qui contient le curseur soit redessinée, c'est pour cela qu'un message ''redessiner'' est envoyé.

Le bloc d'excitation placé au tout début du système se retrouve pour des raisons techniques étroitement lié au tout dernier élément de cette chaîne, qui est l'exécution de la commande. En effet, comme nous avons choisi de commander le mouvement d'un curseur, c'est celui-ci même, qui contient l'excitation visuelle, qui va subir (i.e. exécuter) la commande.

2.3.2.3. Le bloc d'exécution

C'est le dernier élément de notre système. Ce bloc reçoit la commande générée à la fin du traitement via la liaison série et exécute la fonction correspondante (Fig.18).

La tâche de ce bloc est de lire l'entrée du port série et d'en tester le contenu. Ce qui se traduit par le déplacement du curseur dans l'une des 4 directions ou un click (Fig.19).

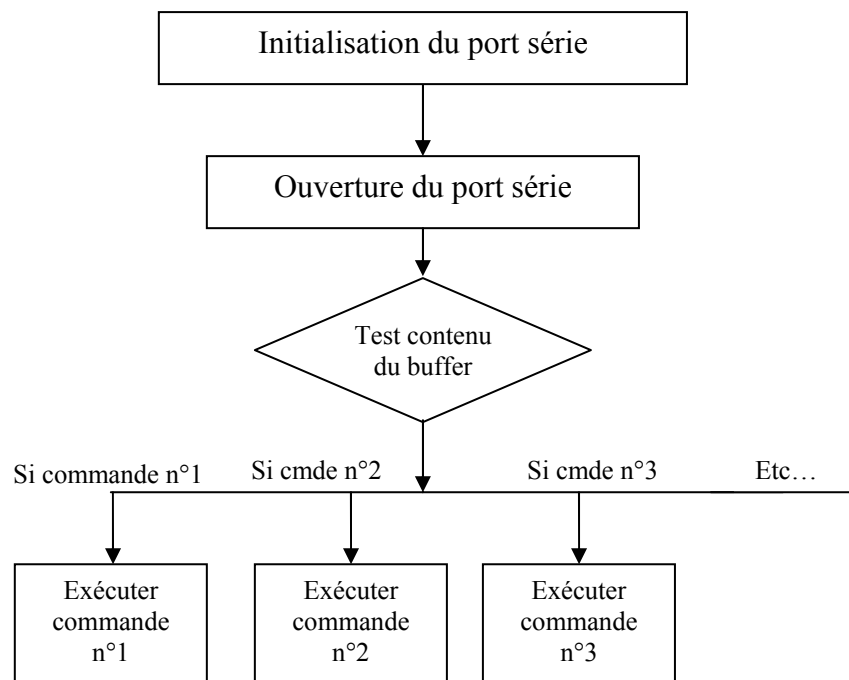


Fig.18 : Algorithme général de l'exécution de commandes reçues sur le port série

Commentaires

- Le protocole de communication utilisé est très simple. Chaque fois que le bloc de traitement produit une décision, il la convertit en commande. La commande que nous avons choisie est l'écriture de caractères sur le port série.
- Les commandes testées, qui sont représentées dans l'algorithme, sont les caractères reçus dans le buffer d'entrée du port série du PC cible.

Remarque

Le curseur est configuré de façon à ce que une fois arrivé sur l'un des bords de l'écran, le clignotement du côté opposé permettant de s'éloigner de ce bord, reste visible.

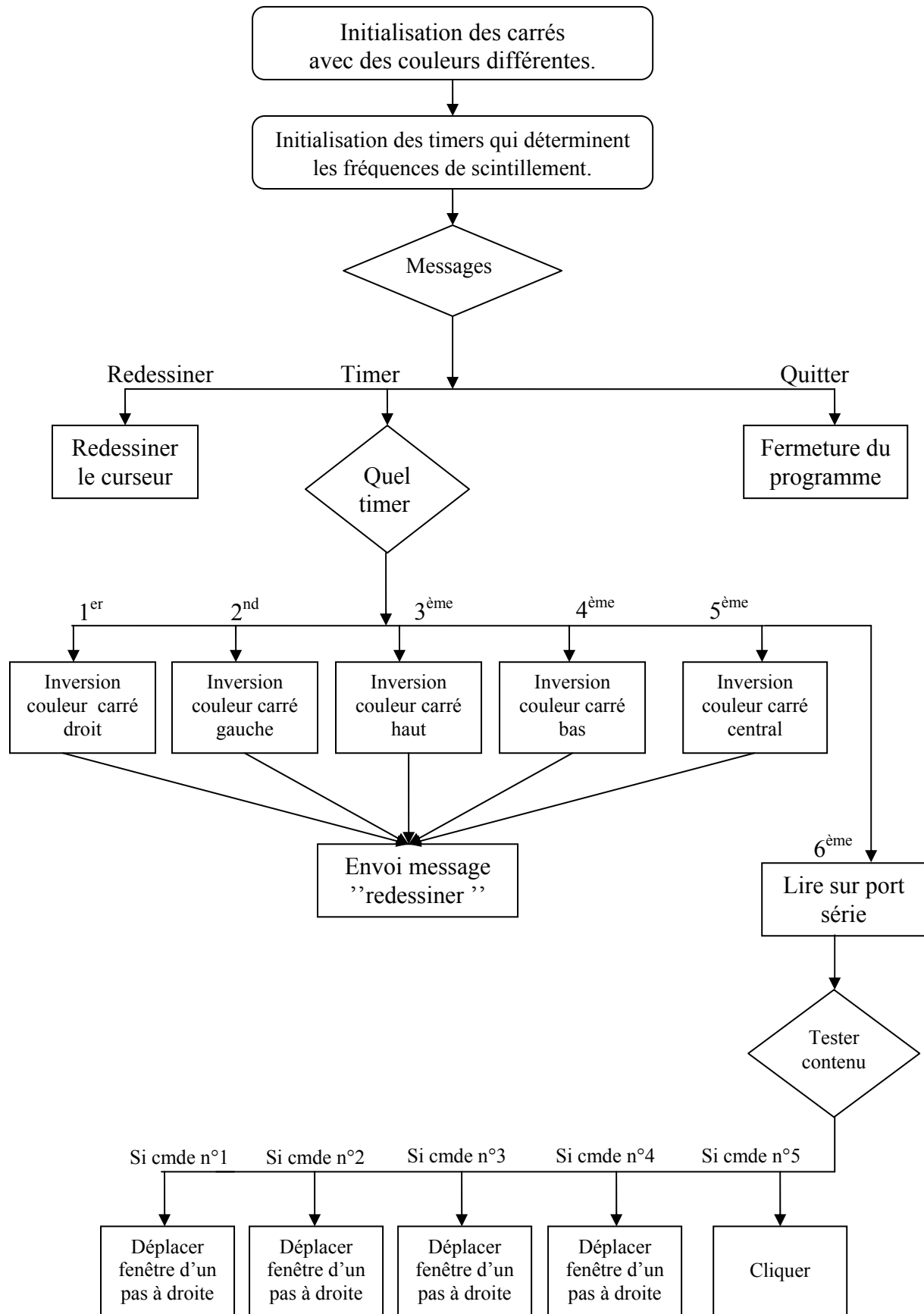


Fig.19 : Algorithme du système gérant le stimulus et l'exécution de la commande.

2.3.3. Conception de l'interface d'adaptation et de mise en forme

Cette interface a pour objectif d'adapter le signal du capteur à l'entrée analogique de la carte DSK. Comme la numérisation se fait au niveau de la carte, ce n'est pas une chaîne d'acquisition proprement dite, mais une interface d'adaptation et de mise en forme (Fig.20).

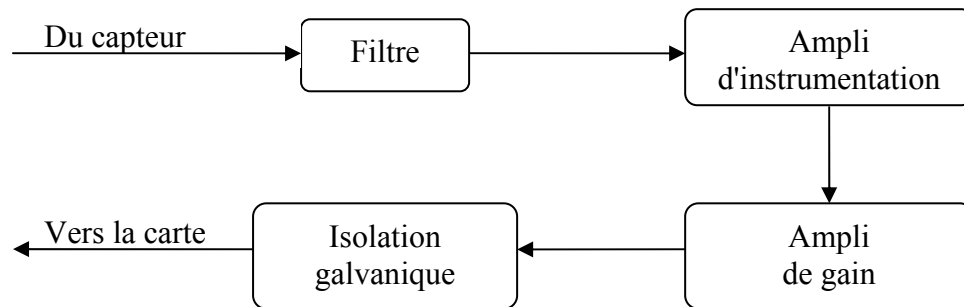


Fig.20 : Schéma de principe de l'interface d'adaptation

Nous avons simulé ce montage (annexe2) sur PSpice, et il a donné des résultats concluants. Nous allons décrire dans ce qui suit, chaque bloc de ce sous-système, et commenter par la suite le résultat des simulations.

2.3.3.1. Le filtre

Nous avons décidé de mettre un filtre passif, car un filtre actif introduit du bruit. Ce filtre sera en amont de l'amplificateur afin d'éviter d'amplifier les fréquences indésirables, ce sera le cas pour la raie de 50 Hz du réseau électrique et ses harmoniques dont l'énergie n'est pas négligeable. De plus, ce filtre nous évite d'avoir des fréquences d'intermodulation à la sortie des amplificateurs, notamment à la sortie du transistor de l'optocoupleur.

Le filtre choisi est un filtre RC (Fig.21). Sa fréquence de coupure (à -3dB) a pour expression :

$$F_c = 1 / 2\pi RC$$

$$\text{Pour } \begin{cases} R = 200\Omega \\ C = 20\mu F \end{cases} \Rightarrow F_c = 39.75\text{Hz}$$

Ce genre de filtre introduit un déphasage dans

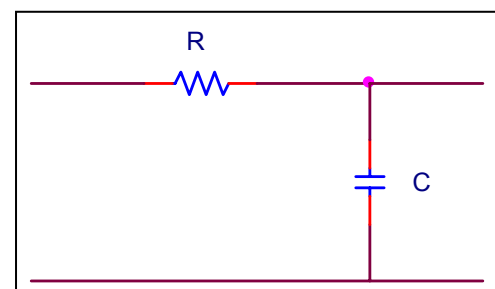


Fig.21 : Filtre passe-bas (cellule RC)

la plage $[0, \pi/4]$ pour les fréquences f appartenant à l'intervalle $[0, F_c]$. Ce phénomène n'est pas gênant pour notre application car la phase ne nous intéresse pas.

2.3.3.2. L'amplificateur d'instrumentation

Le montage illustré à la figure 22 constitue un amplificateur d'instrumentation. La sortie de cet amplificateur est proportionnelle à son entrée différentielle. Le gain est déterminé par le rapport des résistances du montage. L'étude du montage a été faite dans la référence [15], les résultats à retenir sont les suivants :

- Les performances du montage dépendent en majorité de la tolérance des résistances. Avec des résistances précises, un amplificateur opérationnel de performances moyennes satisfait au cahier de charge. Le LM324 a été retenu pour la facilité de sa mise en œuvre.
- Le gain différentiel du montage est réglé au niveau du premier étage, le second étage a un gain unité pour minimiser le gain du mode commun. Ce qui a pour conséquence d'augmenter le rapport de rejection du mode commun (RRMC) de 6dB [15].

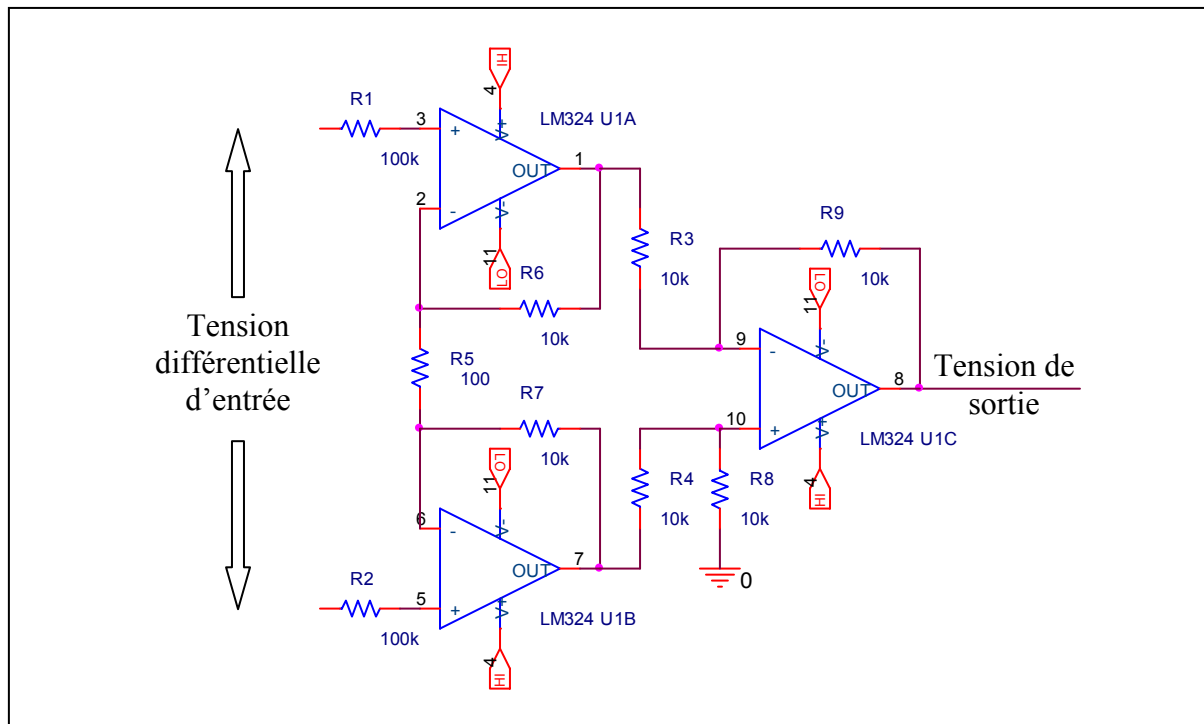


Fig.22 : Schéma de l'amplificateur d'instrumentation

Les résistances du montage doivent vérifier : $\frac{R_9}{R_3} = \frac{R_8}{R_4}$ et $R_6 = R_7$. Ce sont ces résistances qui doivent être appairées. C'est de leur précision que va dépendre la précision du résultat total [15].

Il existe des amplis d'instrumentation sous forme de circuit intégré dont le INA110, les résistances sont appairées par ablation Laser, ce qui procure de grandes performances, mais aussi un coût élevé. Pour notre cas des résistances à 0.1% sont acceptables.

Lors des simulations, nous avons constaté qu'il faut éviter les gains forts qui peuvent causer un des écrêtages. Par contre un faible gain conduit à un offset qui n'est autre que le mode commun. Ce dernier diminue en augmentant le gain du montage. Un compromis a été fait en choisissant les bonnes valeurs des résistances. Un signal à l'entrée différentielle à 100μVpp se trouve à la sortie du montage amplifié à 20mVpp. L'offset est négligeable, et quasiment imperceptible.

2.3.3.3. L'amplificateur de gain

Ce montage est un amplificateur non inverseur classique. Il a pour rôle de régler le niveau de signal pour avoir une bonne modulation de l'optocoupleur, notamment via un potentiomètre à la place de R_{12} (Fig.23).

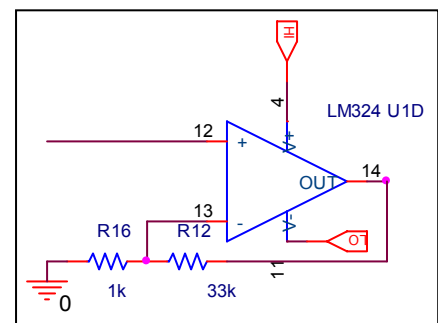


Fig.23 : Amplificateur en montage non-inverseur

2.3.3.4. L'isolation galvanique

L'isolation galvanique a pour but de protéger l'utilisateur contre tout éventuel risque d'électrocution, en l'isolant du réseau électrique. Cette tâche sera remplie par un optocoupleur, le 4N33 (Fig.24).

Le point de fonctionnement de la diode est fixé par la résistance R_{11} . Les variations devant moduler la diode sont transmises par la capacité $C1$.

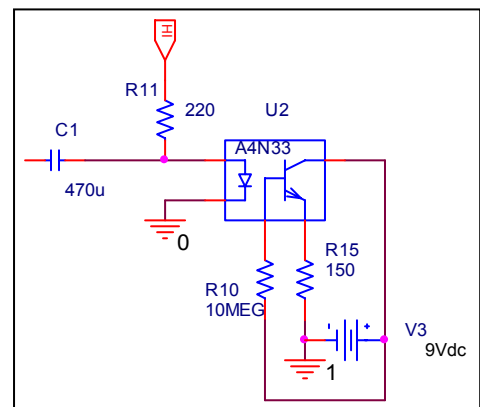


Fig.24 : Couplage optique

La base du phototransistor est mise en l'air à l'aide d'une résistance de 10 MégaOhm. La Tension aux bornes de la résistance R_{15} est prélevée pour attaquer l'entrée analogique de la carte DSK. Cette entrée possède une capacité de liaison supportant jusqu'à 10VDC, donc nous avons omis volontairement d'en mettre une sur l'interface d'adaptation.

Après avoir validé le fonctionnement de notre montage, nous avons réalisé le schéma du circuit imprimé (Fig.25), ainsi que le schéma d'implantation (Fig.26) sur Layout.

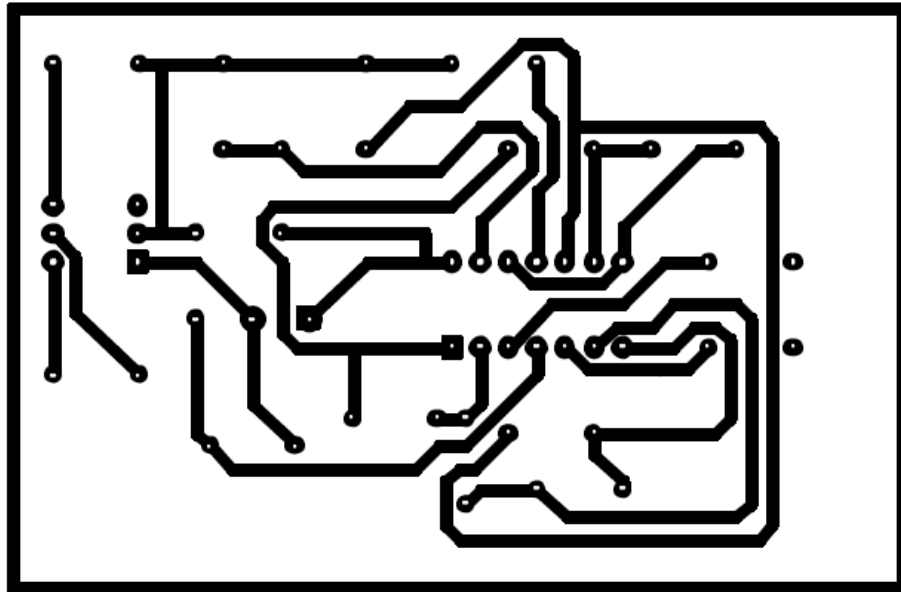


Fig.25 : Circuit imprimé de l'interface d'adaptation (filtre non inclus)

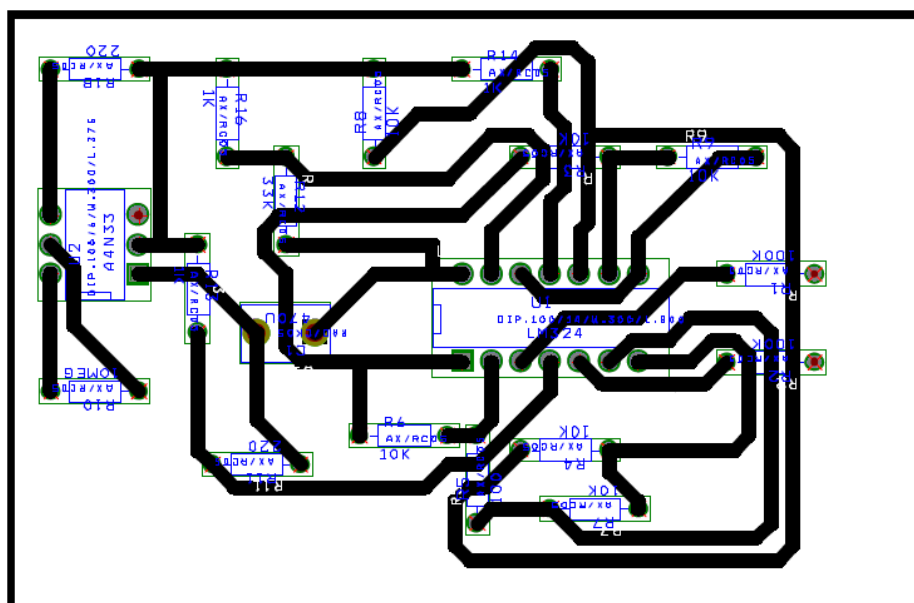


Fig.26 : Schéma d'implantation des composants

Remarques

- Le Filtre n'est pas inclus dans ces schémas.
- Les pistes ont été élargies afin d'éviter qu'elles ne se décollent lors de la soudure

2.4. Conclusion

Nous avons testé notre système en utilisant un GBF pour fournir le signal analogique d'entrée, notre programme implémenté sur la carte DSK et l'interface graphique développée. Les résultats obtenus permettent d'affirmer que mis à part l'acquisition du signal EEG, notre système est opérationnel.

La partie développée sous Simulink a fourni de bonnes performances en terme de temps réel et de détection. Néanmoins elle constitue une solution expérimentale, qui reste encombrante, vu la nécessité d'utiliser un micro-ordinateur.

L'implémentation sur la carte DSK fonctionne parfaitement mais il faut rester prudent car l'utilisation d'un signal EEG risque de poser quelques problèmes à cause de la résolution fréquentielle limitée. L'utilisation du même algorithme avec une nouvelle carte mieux adaptée peut pallier à ce problème.

Le dispositif de pointage est une interface graphique qui fournit les fonctionnalités de base d'une souris. Cette interface gère aussi l'excitation visuelle.

En simulation, l'interface d'adaptation et de mise en forme donne des résultats satisfaisants. Elle est prête à être réalisée mais ne l'a pas été faute de temps. Cela explique l'absence d'une étude sur le capteur à employer.

Conclusion générale

L'évolution dans la manière de considérer la personne handicapée, a eu pour conséquence de rechercher le moyen de lui permettre de se prendre en charge sans assistance extérieure.

Les signaux EEG sont largement employés en médecine pour détecter certaines pathologies. Les recherches actuelles tentent de décrypter les informations véhiculées par ces signaux.

L'idée de base d'une ICO est l'exploitation des signaux EEG pour apporter à la personne handicapée plus d'indépendance.

L'ICO que nous avons développée détecte l'activité mentale provoquée par une excitation visuelle et génère après traitement la commande du dispositif de pointage. Ce traitement a été réalisé selon deux approches, l'une logicielle et l'autre matérielle.

Dans les deux cas de figure, les résultats obtenus ont satisfait les objectifs visés. L'information est correctement détectée et traduite et la contrainte du temps réel est respectée. L'approche logicielle permet une grande souplesse de développement. L'approche matérielle fournit la solution embarquée.

Le dispositif de pointage n'est pas la seule application possible pour le système. Nous pouvons envisager de programmer d'autres applications en restant dans l'environnement du système d'exploitation ou en réalisant des dispositifs cibles compatibles avec le protocole série utilisé.

Pour que le système soit opérationnel en sa totalité, il faut le compléter par le bloc analogique. Nous avons réalisé l'étude. Il ne reste plus qu'à implanter les composants et faire une étude pour déterminer le capteur EEG adéquat.

Une fois achevé, le système devra subir des tests d'évaluation afin de déceler d'éventuelles imperfections à corriger.

Pour augmenter la fiabilité du système, nous proposons d'ajouter un algorithme pour éliminer les artefacts tels que le clignement des yeux et la contraction musculaire.

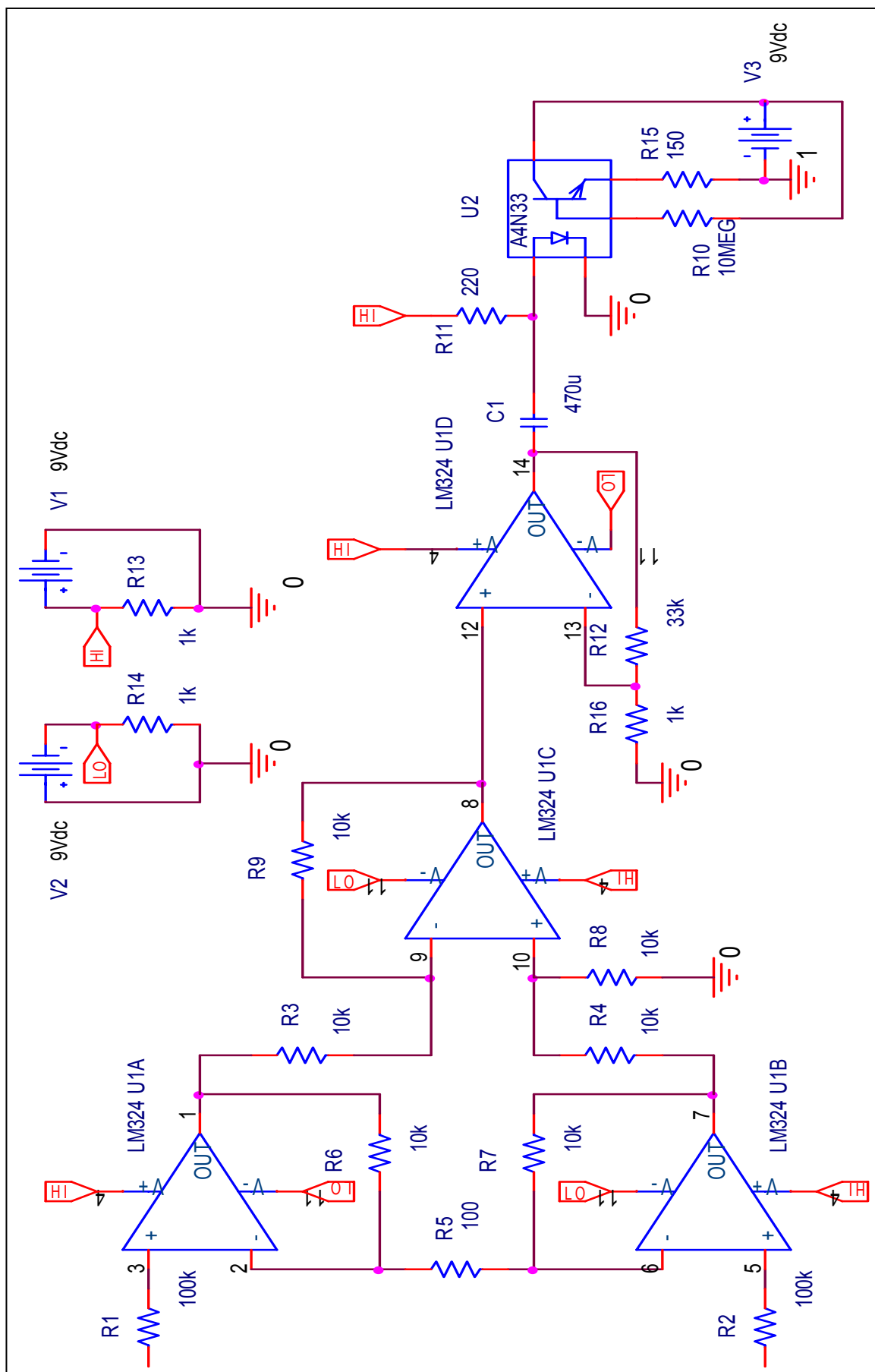
Les résultats de nos expérimentations présagent un possible aboutissement de ce projet en produit fini destiné aux handicapés moteurs et son utilisation en produit grand public pourra être envisagée.

Références bibliographiques

- [1] Jacques J. Vidal : ‘Toward Direct Brain Computer Communication’, In L.J. Mullins (ed.), Annual Review of Biophysics and Bioengineering, Annual Review Inc., 1973, pp.157-180.
- [2] J.R. Wolpaw, N. Birbaumer, W.J. Heetderks, D.J.McFarland, P.H. Peckham, G. Schalk, E. Donchin, L.A. Quatrano, C.J. Robinson, T.M. Vaughan, “Brain-computer interface technology: a review of the first international meeting,” IEEE Trans. Rehab.Eng., Vol.8, pp.164-173, Juin 2000.
- [3] Jonathan R. Wolpawa, Niels Birbaumer, Dennis J. McFarland, Gert Pfurtscheller, Theresa M. Vaughan : ‘Brain–Computer Interfaces for Communication and Control’, Clinical Neurophysiology (113), 2002, pp.767-791.
- [4] Site internet de l’université d’Aalborg, Danemark : <http://www.smi.auc.dk>
- [5] Gary Nelson Garcia Molina : ‘Direct Brain-Computer Communication Through Scalp Recorded EEG Signals’, Thèse de doctorat, Ecole Polytechnique Fédérale de Lausanne, Suisse, 2004.
- [6] Rémy Wahnoun, Rajiv Saigal, Ying Gu, Nicolas Paquet, Sofie DePauw, Andrew Chen, Saber Sami A.K., and Kim Dremstrup Nielsen, ”A Real-Time Brain-Computer Interface Based on Steady-State Visual Evoked Potentials”, Centre pour l’Interaction Sensori-motrice; Université de Aalborg, Danemark, 2002.
- [7] C Ming et Shang kai, “An EEG-based cursor control system. Proceeding of The First Joint BMES/EMBS Conference: Serving Humanity, Advancing Technology, Atlanta, GA, Etats Unis d’Amérique, 1999.
- [8] <http://www.ebwe.co.uk/arts/eegintro>
- [9] Alpo Varri : ‘Digital Processing of the EEG in Epilepsy’, Licenciata thesis, Université de Tampere de Technologie, Finlande, 1988.

- [10] TMS320VC5402 DSK Help, Code Composer Studio, Texas Instrument
- [11] Optimized DSP Library for C Programmers on the TMS320C54x, Application Report SPRA480B, Texas Instruments, October 2000
- [12] <http://www.dsprelated.com>
- [13] <http://dualist.stanford.edu/~ee265/labs/lab4.html>
- [14] Patrick Lee, Barry Lee, Akshay Sateesh, Joyce Yang Visual Evoked Potentials and the Physiological Analysis of Visual Processes in Man
- [15] Alexandre Dziadowiec, Marc Lescure : 'Fonctions à Amplificateurs Opérationnels', Eyrolles, 2002.

Annexes



Résumé

Dans notre travail, nous proposons un système d'aide aux personnes handicapées moteur. Il consiste en une interface cerveau ordinateur (ICO) qui exploite les réponses évoquées à un stimulus visuel permanent (SSVER) véhiculées par le signal électroencéphalogramme (EEG). Pour traiter ce signal, nous avons procédé selon deux approches, l'une logicielle et l'autre matérielle à base de DSP.

Mots clés : EEG, SSVER, ICO, DSP.

Abstract

In our work, we propose a help system for motor disabled people. It consist of a brain computer interface (BCI) that uses the steady state visuel evoqued response (SSVER) contained in the electroencephalogram signal (EEG). To process this signal, we use two approaches, the first is software and the second is hardware based on digital signals processor (DSP).

Keywords : EEG, SSVER, BCI, DSP.

ملخص

في عملنا هذا نقترح نظام مساعدة للأشخاص المعوقين حركيا، يتمثل في واجهة دماغ كمبيوتر. تستغل الأجابة الناتجة عن إثارة مرئية دائمة محمولة في الإشارات الكهربائية للدماغ. لمعالجة هذه الإشارات اتبعنا طريقتين احدهما منطقية و الأخرى مادية مبنية على أساس DSP.

الكلمات المفتاحية: الإشارات الكهربائية للدماغ، الأجابة الناتجة عن إثارة مرئية دائمة، واجهة دماغ كمبيوتر، DSP.