

UNIVERSITÉ MONTPELLIER II

Sciences et techniques du Languedoc



IUP Génie Électrique et Informatique Industrielle



Rapport du stage :

**Conception d'un système anti-bruit pour des
systèmes de ventilation (partie Software)**

Exposé par :

Zaki Nabil ABIDAT (M2 Pro OHST)

Tuteur :

Marc LAFAGE

Septembre 2006

*Je dédie ce modeste travail à mes parents
et à ma Zahra; et je tiens à remercier M. Lafage
et les responsables de CLIM'MAT, ainsi que
tous ceux qui m'ont aidés de près ou de loin à le
faire.*

Sommaire :

I)- Introduction	1
II)- L'entreprise.....	2
III)- Étude theorique.....	3
1- Introduction.....	3
2- Notions d'Acoustique	3
2.1- L'Acoustique.....	3
2.2- Les ondes sonores.....	3
2.3- La vitesse du son	4
2.4- La vitesse vibratoire	4
2.5- La pression sonore.....	5
2.6- L'intensité sonore	5
2.7- L'impédance acoustique.....	5
2.8- Niveau de pression sonore	5
2.9- Niveau de puissance sonore.....	6
2.10- Bruit acoustique	6
3- Les filtres adaptatifs.....	6
3.1- Définitions	6
3.2- L'algorithme LMS	7
3.3- L'algorithme RLS.....	8
4- Contrôle du bruit.....	9
4.1- Le contrôle passif du bruit	9
4.2- Le contrôle actif du bruit.....	9
4.2.1- Définitions	9
4.2.2- Feedback ANC.....	11
4.2.3- Feedforward ANC	11
- Effet du chemin secondaire.....	12
- Effet de la contre-réaction	15
5- Les DSP	16
5.1- Généralités.....	16
5.2- Les différents types de DSP.....	18
5.2.1- DSP à virgule flottante	18
5.2.2- DSP à virgule fixe.....	18
5.3- Le choix d'un DSP.....	19

IV)- Étude pratique	21
1- Préambule	21
2- Bruit utilisé.....	21
3- Simulations.....	21
3.1- Remarque	21
3.2- Résultats.....	22
3.3- L'application	25
V)- Conclusion	26
Annexes	28
- Présentation du DSK C6713	28
- La programmation du DSP.....	31

Bibliographie.

I)- Introduction

Durant mon année de formation au M2 Pro OHST, de l'Université Montpellier II, j'ai fait un stage de 5 mois, de mars à juillet 2006, chez *CLIM'MAT* (voir présentation de l'entreprise au chapitre suivant).

J'ai travaillé en collaboration avec un camarade de promo, M. Pedro LIMA, sur l'étude et la réalisation d'un système anti-bruit actif pour leurs systèmes de ventilation. Je me suis (plus) occupé de la partie logicielle ("soft") du travail qui nous a été confié, et lui de la partie matérielle ("hard").

Nous avons scindé notre travail en 3 tâches :

- L'étude théorique complète du contrôle actif du bruit, englobant le choix d'un algorithme, le choix de la méthode, etc.
- La simulation des méthodes du contrôle actif du bruit sous *Matlab*.
- L'application pratique sur un DSP à choisir.

Après une très brève présentation de l'entreprise *CLIM'MAT*, nous allons aborder en première partie : la théorie nécessaire à la compréhension du contrôle du bruit (notions d'Acoustique, algorithmes adaptatifs, ...) puis le choix de l'algorithme, de la méthode et du DSP ("*Digital Signal Processor*") pour notre application; en seconde partie on discutera de nos simulations et expérimentations

II)- L'entreprise

CLIMMAT est une petite entreprise de 12 salariés, créée en 2004, et spécialisée dans : l'installation, la maintenance et le dépannage des chauffages et climatisations centrales réversibles.

Elle fournit ses prestations sur tout le territoire français, et travaille avec les grandes marques : *Panasonic, Toshiba, LG, Hitachi, Mitsubishi, ... etc.*

Ses Coordonnées :

-Site web :

www.climmat.com

-Siège social (Saint Jean de Védas) :

12, Impasse des Poiriers

34430 Saint Jean de Védas

Tél. : + 33 4 67 65 80 57

Fax : + 33 4 67 65 47 10

Courriel : climmat@climmat.com

-Région Ouest (Agde) :

15, Rue Jean Roger

34300 AGDE

Tél. : + 33 4 67 21 26 38

Fax : + 33 4 67 21 27 09

Courriel : agde@climmat.com

III)- Étude théorique

1- Préambule :

Le bruit acoustique est une nuisance sonore, qui affecte physiquement et psychologiquement les êtres vivants, une longue exposition à ce bruit conduit à des dommages irréversibles.

Avec l'expansion, ces dernières années, des activités industrielles et l'augmentation des trafics (routiers, aériens et maritimes), on constate une très forte croissance du niveau de bruit dans notre vie quotidienne. Ainsi, la réduction de bruit est devenu l'un des thèmes majeurs de la recherche scientifique de cette dernière décennie. Et la prise en compte du critère "confort acoustique" dans la conception de produits, était auparavant une tâche de second plan développée en aval du dispositif existant; mais actuellement, on assiste de plus en plus à une apparition du critère "confort acoustique" au niveau de l'avant projet.

Dans la plupart des cas, on ne peut pas agir directement sur la source acoustique qui génère la perturbation; alors on utilise, soit des méthodes passives, soit des méthodes actives, pour atténuer le rayonnement sonore.

2- Notions d'Acoustique : [L1, R1]

2.1- L'Acoustique :

L'Acoustique est la science qui étudie la production, la propagation, la réception et les effets des ondes sonores (comprises entre 20 Hz et 15 kHz), infrasonores (< 20 Hz) et ultrasonores (>15 kHz).

2.2- Les ondes sonores :

Du point de vue physique une onde sonore est une propagation d'ondes de pressions dans un milieu élastique (liquide ou gaz), ou plus généralement un ébranlement élastique dans un milieu quelconque.

Ce milieu de propagation est capable de reprendre son état initial naturel après avoir subi les déformations relatives à une contrainte temporaire; ces déformations doivent être petites par rapport aux dimensions du corps, et les contraintes doivent être assez faibles pour ne pas dépasser le domaine élastique, car des contraintes trop fortes et des déformations importantes conduiraient après suppression des contraintes à des déformations permanentes et non réversibles.

Le milieu élastique peut être considéré comme l'ensemble de particules ayant une masse, une compressibilité, et de ce fait par leur contact avec les voisines, des liaisons élastiques qui leur permettent de se déplacer de leur position moyenne. Si une particule est déplacée, elle déplace à son tour les voisines par suite des liaisons élastiques; la perturbation initiale se propage et le mouvement

de chaque particule entraînée possède les principales caractéristiques du mouvement initiale, avec un retard proportionnel à son éloignement de la source.

Cependant, au cours du mouvement les amplitudes de déplacement, les vitesses et les accélérations, varieront en fonction des phénomènes de propagation (amortissement, dispersion, ...).

2.3- La Vitesse du son :

Une propagation transporte d'un point à un autre, avec un certain délai, l'état vibratoire, elle a donc une vitesse de phase (ou une célérité) et aussi une vitesse de groupe; mais en acoustique "classique" ces deux vitesses sont égales.

La vitesse du son ne dépend que des caractéristiques élastiques et de la température absolue du milieu, donc elle ne dépend ni des fréquences, ni des pressions, ni des intensités sonores. Elle est constante dans la mesure où le matériau est homogène.

Ci-dessous quelques valeurs de célérité sonore pour quelques milieux :

- Dans l'air :

$$c_0 \approx 20\sqrt{T_a} \quad (\text{Eq. 3.1})$$

T_a : représente la température absolue en [° K] (température en ° C + 273,15).

c_0 : représente la célérité en [m/s].

A 0° C : $c_0 = 330,45$ m/s; à 20° C : $c_0 = 342,35$ m/s; à 100° C : $c_0 = 386,25$ m/s.

- Dans les métaux : Aluminium : $c_0 = 5200$ m/s; Acier : $c_0 = 5100$ m/s; Bronze : $c_0 = 3000$ m/s.

- Dans l'eau : $c_0 = 1410$ m/s.

- Dans le bois : Bois blanc : $c_0 = 4200$ m/s; Bois dur : $c_0 = 3500$ m/s.

2.4- La vitesse vibratoire :

Les molécules du milieu sont agitées d'une vitesse vibratoire fluctuante $\vec{u}(t)$, de valeur très petite par rapport à la célérité du milieu.

Cette vitesse $\vec{u}(t)$ ayant une valeur moyenne nulle, on considère sa valeur efficace (ou valeur quadratique moyenne) :

$$u_e = \sqrt{u^2(t)} \quad (\text{Eq. 3.2})$$

2.5- La pression sonore :

Les sources sonores produisent en chaque point de l'espace une pression instantanée fluctuante $p(t)$, représentant les fluctuations de la source, cette pression s'ajoute à la pression atmosphérique (ambiante), mais est de valeur très petite par rapport à elle.

Ayant une valeur moyenne nulle, on considère sa valeur efficace (ou valeur quadratique moyenne), qu'on appelle *pression acoustique* :

$$p_e = \sqrt{\overline{p^2(t)}} \quad (\text{Eq. 3.3})$$

2.6- L'intensité sonore :

L'intensité sonore ou le flux d'énergie transporté par unité de surface, est la moyenne du produit de la pression sonore et de la vitesse ondulatoire, prise sur un temps plus long que les périodes de leurs fluctuations; elle est exprimée en $[\text{W}/\text{m}^2]$.

$$I = \overline{p \cdot u} \quad (\text{Eq. 3.4})$$

Si φ est la différence de phase entre p et u , on peut aussi écrire :

$$I = p_e \cdot u_e \cdot \cos \varphi \quad (\text{Eq. 3.5})$$

2.7- L'impédance acoustique :

L'impédance acoustique (spécifique) est définie comme étant le rapport entre les valeurs instantanées de la pression et de la vitesse en un point, ce rapport dépend du milieu de propagation et de ses limites; sa dimension est $[\text{Kg}/\text{m}^2 \cdot \text{s}]$ ou $[\text{N} \cdot \text{s}/\text{m}^3]$.

$$Z = \frac{p}{u} = R + j \cdot X = |Z| \cdot e^{j\varphi} \quad (\text{Eq. 3.6})$$

2.8- Niveau de pression sonore :

Le niveau de pression sonore, en anglais SPL ("*Sound Pressure Level*"), est défini comme étant le logarithme décimal du rapport entre une pression sonore et la pression de référence (seuil d'audibilité), qui vaut $p_{eff0} = 2 \cdot 10^{-5}$ Pa.

$$L_p = 20 \cdot \log \left(\frac{p_{eff}}{p_{eff0}} \right) \quad (\text{Eq. 3.7})$$

2.9- Niveau de puissance sonore :

Le niveau de pression sonore, en anglais SWL ("*Sound poWer Level*"), est défini comme étant le logarithme décimal du rapport entre une puissance sonore et la puissance de référence (seuil d'audibilité), qui vaut $P_0 = 10^{-12}$ W.

$$L_p = 10 \cdot \log\left(\frac{P}{P_0}\right) \quad (\text{Eq. 3.8})$$

2.10- Bruit acoustique :

Il existe deux sortes de bruits :

- Le bruit large bande ("*broadband noise*"), où l'énergie est répartie de manière uniforme sur toutes les fréquences. Il est généralement aléatoire.
- Le bruit à bande étroite ("*narrowband noise*"), où l'énergie est concentrée sur certaines bandes de fréquences. Il est généralement périodique, ou plus ou moins périodique. On le trouve surtout dans les moteurs, turbines, etc.

3- Les filtres adaptatifs : [L2, R1, R2, R3]

3.1- Définitions :

Un filtre adaptatif est, par définition, un filtre numérique dont les coefficients estimés au sens d'un critère donné (généralement de type des moindres carrés), s'adaptent aux variations des signaux reçus. Habituellement, un vecteur d'entrée et une réponse désirée sont utilisés pour définir un vecteur d'erreur qui contrôle alors l'évolution des paramètres du filtre adaptatif.

L'objectif des filtres adaptatifs est d'approximer les fonctions de transferts inconnues, en "apprenant" les caractéristiques des signaux au fur et à mesure que ces derniers se déroulent. Ils se composent de deux parties : un filtre numérique pour filtrer et un algorithme pour ajuster les coefficients de ce filtre.

Dans la figure suivante on peut voir un schéma simplifié d'un filtre adaptatif, où $d(n)$ représente le signal désiré, $y(n)$: la sortie du filtre numérique ($y(n) = x(n) * h(n)$), et $e(n)$: le signal d'erreur.

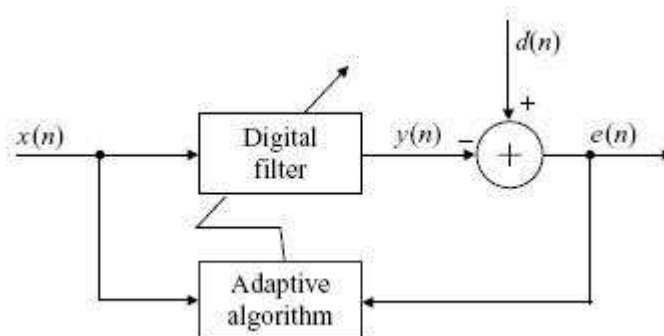


Fig. 3.1

Les filtres adaptatifs numériques peuvent être classés selon leurs réponses impulsionnelles :

- Finie, et dans ce cas on parle de filtre à réponse impulsionnelle finie (RIF, ou "FIR" en anglais). Ce type de filtre est de loin le plus utilisé dans les filtres adaptatifs, car ils sont inconditionnellement stables (absence de pôles).
- Infinie, et dans ce cas on parle de filtre à réponse impulsionnelle infinie (RII, ou "IIR" en anglais). Nonobstant le fait que ce type de filtre est plus proche de la réalité physique (présence de pôles et de zéros), ils sont très peu utilisés à cause des risques d'instabilités.

Le choix de l'algorithme du filtre adaptatif repose sur plusieurs critères, les plus importants :

- Le critère d'optimisation : rapidité de convergence, complexité, robustesse au bruit ...
- La structure du filtre programmable.
- Le type de signal traité : mono ou multidimensionnel, réel ou complexe ...
- La stabilité du filtre.

Nous allons présenter dans les prochains paragraphes, les deux principaux algorithmes adaptatifs utilisés en pratiques.

3.2- L'algorithme LMS :

L'algorithme LMS ("*Least Mean Squares*"), conçu en 1959, est le plus utilisé des algorithmes adaptatifs. Il est basé sur la méthode du gradient qui calcule et remet à jour les pondérations de façon récursive. On montre que l'erreur est une forme quadratique des pondérations et, intuitivement, la solution optimale est obtenue en corrigeant pas à pas le vecteur de pondération dans la direction du minimum.

Dans la figure suivante on peut voir l'algorithme LMS standard, où μ compris entre 0 et 1, représente le pas d'adaptation; plus μ est petit plus la convergence est lente mais plus l'EQM excédentaire (l'Écart Quadratique Moyen) est petit; et vice-versa.

Initialise by setting:

$$\mathbf{h}(0) = \mathbf{0}$$

For each time sample $n = 1, 2, \dots$ do the following:

$$\hat{x}(n) = \mathbf{h}^T(n-1) \mathbf{y}(n)$$

$$e(n) = x(n) - \hat{x}(n)$$

$$\mathbf{h}(n) = \mathbf{h}(n-1) + 2 \mu \mathbf{y}(n) e(n)$$

Fig. 3.2

Parmi les très nombreuses variantes de l'algorithme LMS standard, on peut citer les plus utilisées en pratique :

- Le LLMS ("*Leaky LMS*"), où un paramètre (γ) est multiplié par $h(n-1)$ dans la dernière équation de la Fig. 3.2.
- Le NLMS ("*Normalised LMS*"), où μ est non plus une constante, comme dans le LMS standard, mais une fonction de la puissance du signal d'entrée ($x(n)$ dans la Fig. 3.1). Cette variante est la plus utilisée.
- Le FxLMS, est surtout utilisé dans les systèmes ANC; là aussi μ est non pas une constante, mais une fonction de la puissance du signal d'entrée, et aussi du retard dû au chemin principal. C'est cet algorithme que nous utiliserons en pratique, nous allons le décrire plus longuement ultérieurement.

Les principaux avantages des algorithmes LMS, sont leur simplicité et leurs performances. Mais leurs convergences sont lentes par rapport aux autres algorithmes.

3.3- L'algorithme RLS :

La différence principale, au niveau de l'algorithme, entre le LMS et le RLS ("*Recursive Least Squares*"), réside dans le fait que ce dernier utilise, plus ou moins, toutes les données passées pour effectuer les calculs, alors que le LMS n'utilise que la donnée actuelle.

Le premier algorithme RLS a été conçu en 1950, mais la première version rapide n'a été développée qu'en 1974.

Dans la figure suivante on peut voir l'algorithme RLS standard, où α (γ , dans certains ouvrages) compris entre 0 et 1, représente le facteur d'oubli qui permet de donner plus d'importance aux échantillons les plus récents, tandis que les échantillons plus lointains dans le temps sont omis.

Initialise by setting:

$$\mathbf{R}_{yy}(0) = \frac{1}{\delta} I_N, \quad \delta \text{ is a small positive number}$$

$$\mathbf{h}(0) = \mathbf{0}$$

For each time sample $n = 1, 2, \dots$ do the following:

$$\hat{x}(n) = \mathbf{h}^T(n-1) \mathbf{y}(n)$$

$$e(n) = x(n) - \hat{x}(n)$$

$$\mathbf{R}_{yy}^{-1}(n) = \frac{1}{\alpha} \left(\mathbf{R}_{yy}^{-1}(n-1) - \frac{\mathbf{R}_{yy}^{-1}(n-1) \mathbf{y}(n) \mathbf{y}^T(n) \mathbf{R}_{yy}^{-1}(n-1)}{\left(\alpha + \mathbf{y}^T(n) \mathbf{R}_{yy}^{-1}(n-1) \mathbf{y}(n) \right)} \right)$$

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{R}_{yy}^{-1}(n) \mathbf{y}(n) e(n)$$

Fig. 3.3

Le principal avantage des l'algorithmes RLS, est leur convergence rapide; mais en revanche ils sont plus coûteux en espace mémoire et en temps de calculs.

4- Contrôle du bruit :

4.1- Le contrôle passif du bruit : [R3, R4]

L'approche traditionnelle pour l'annulation du bruit acoustique, c'est-à-dire passif, est de mettre des barrières acoustiques isolant le bruit et l'empêchant de se propager (ex. : fibre de verre, laine de roche, laine de chanvre, placoplâtre ...).

Et cela s'appuie sur deux principes physiques : le changement d'impédances (acoustiques), en utilisant une combinaison d'atténuateurs et/ou de canalisations; et l'absorption d'énergie (acoustiques), en utilisant des matériaux absorbants.

L'utilisation des méthodes passives pour les systèmes générant des bruits (acoustiques) basses fréquences est inadaptée, car dans ces bandes de fréquences; elles sont coûteuses, beaucoup trop volumineuses (à causes des fréquences basses), et quasiment inefficaces.

4.2- Le contrôle actif du bruit : [R3, R4, R5, P1-6]

4.2.1- Définitions :

L'ANC est l'acronyme anglais pour "Active Noise Cancellation" ou "Active Noise Control", il désigne le procédé permettant d'annuler un bruit acoustique, en générant par des filtres adaptatifs (prédictifs) un bruit ayant exactement les mêmes caractéristiques que le bruit, mais avec un

déphasage de 180°. Le résultat de la superposition acoustique, de ces deux bruits est quasi nul, comme on le voit sur la figure suivante :

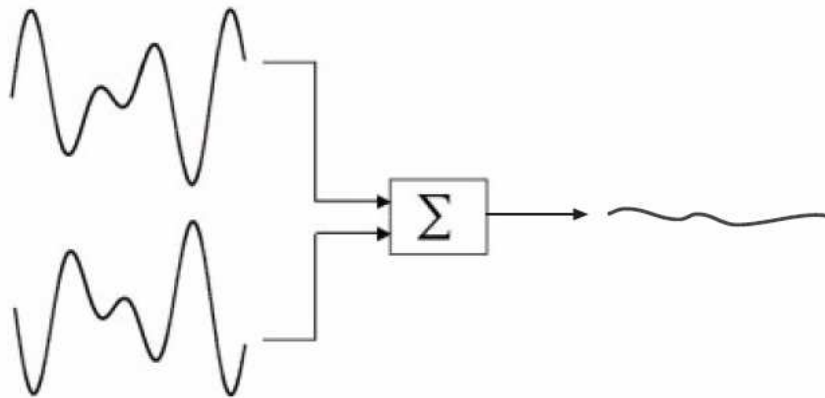


Fig. 3.4

En 1936, Paul Lueg, un ingénieur américain, a breveté le tout premier principe d'un système ANC. Mais le développement et l'application des premiers systèmes ANC en pratique n'ont commencé qu'à partir des années 80, avec l'apparition des premiers DSP performants.

On peut diviser les systèmes ANC en trois catégories, selon le comportement des ondes sonores dans l'environnement où on désire annuler le bruit :

- L'annulation du bruit dans tout l'espace libre. Cela nécessite que l'anti-bruit soit placé juste à coté de la source du bruit, en général à une longueur égale à 0.1 la longueur d'onde du bruit. C'est ce qu'on va essayer de faire avec notre application.
- L'annulation du bruit dans les espaces confinés (cavités, canalisations, etc). Dans ces espaces confinés l'existence des parois donne naissance à des modes (fréquences) propres, qui augmentent rapidement avec la fréquence max. La formule approchée de Maa, donne le nombre de modes en fonction de la fréquence max, du volume, et de la vitesse du son dans le milieu :

$$n_{(f)} \approx \frac{4 \cdot \pi}{3} \cdot V \cdot \left(\frac{f_{\max}}{c_0} \right)^3 \quad (\text{Eq. 3.9})$$

- L'annulation du bruit dans des zones "de silences". Ces petites zones ont, habituellement, des longueurs égales à 0.1 la longueur d'onde du bruit.

Il y a deux types de systèmes ANC, "*feedforward*" et "*feedback*", que nous allons présenter ci-dessous, mais dans tout ce qui suit nous nous limiterons au cas simple des systèmes mono-voies; vu que les systèmes ANC multi-voies ne sont que des systèmes mono-voies travaillant en parallèle. Le lecteur intéressé par les systèmes ANC multi-voies, est invité à consulter la bibliographie : **[P1]**.

4.2.2- Feedback ANC :

Ce système ANC est le plus simple et aussi le moins efficace. Il nécessite l'utilisation d'un haut-parleur pour générer l'anti-bruit, et d'un microphone, dit "d'erreur", pour capter le bruit résultant de la superposition du bruit et de l'anti-bruit; idéalement ce bruit résiduel est nul.

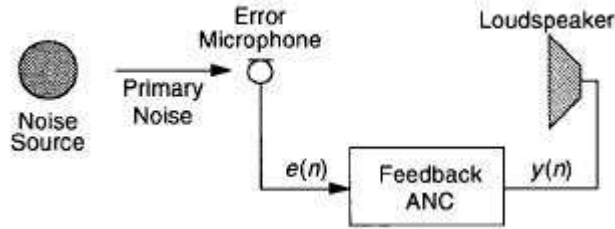


Fig. 3.5

4.2.3- Feedforward ANC :

A la différence du système précédent, celui-ci nécessite de connaître la nature du bruit, dit "primaire", avant sa propagation dans l'environnement où on veut l'annuler (ex. : cavité, canalisation, ...).

Dans le cas d'un bruit large bande, on utilise un microphone situé directement à la source (Fig. 3.6) pour capter ce bruit primaire; et dans le cas d'un bruit à bande étroite ou périodique, on peut se contenter d'un capteur seulement (ex. : un tachymètre pour les moteurs) pour connaître la nature du bruit primaire (les fréquences composants le bruit, la densité spectrale de puissance, etc) (Fig. 3.7).

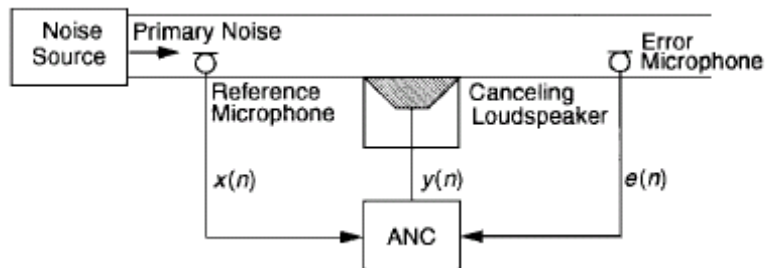


Fig. 3.6

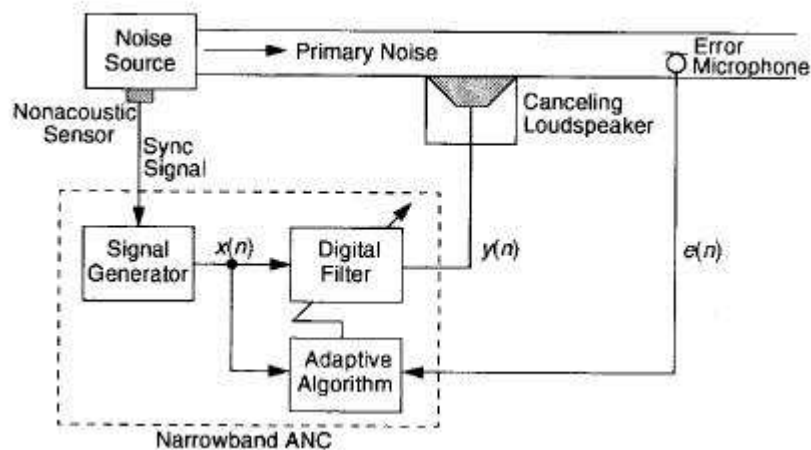


Fig. 3.7

Le système "*broad band feedforward ANC*", c'est-à-dire utilisant deux microphones et un haut-parleur (Fig. 3.6), est le système ANC le plus utilisé à cause de ses excellents résultats pratiques. Ça sera ce système que nous utiliserons dans notre application.

La figure suivante représente le schéma simplifié de ce système, dans lequel un filtre adaptatif (linéaire) $W(z)$ est utilisé pour estimer un processus (linéaire) inconnu $P(z)$. Ce dernier représente la fonction de transfert du chemin principal, c'est-à-dire la réponse acoustique du chemin allant du premier microphone (captant le bruit primaire, $x(n)$) au second microphone (captant l'erreur, $e(n)$).

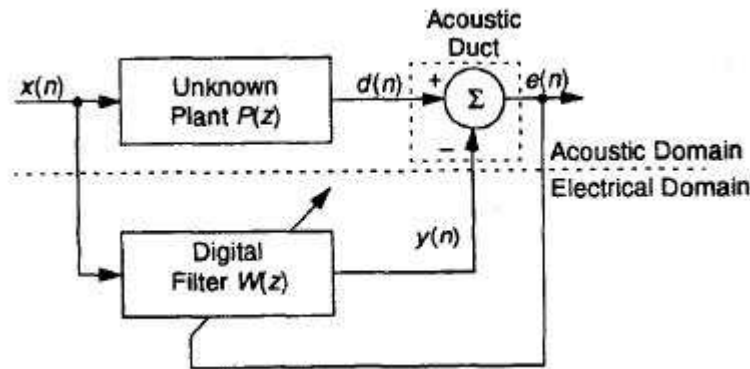


Fig. 3.8

Le but du filtre adaptatif $W(z)$ est la minimisation de l'erreur résiduelle $e(n)$, en générant un signal $y(n)$ devant neutraliser $d(n)$.

Pour $e(n) = 0$ (et $x(n) \neq 0$), c'est-à-dire dans le cas idéal, on a $P(z) = W(z)$, donc $y(n) = d(n)$.

Comme le calcul des coefficients du filtre $W(z)$ nécessite un certain temps, l'adaptation du filtre n'est pas instantanée. Si ce délai électrique devient plus long que le délai acoustique (du microphone de référence au haut-parleur), les performances du système seront inévitablement dégradées; car dans ce cas ci, la réponse du contrôleur est non causale.

Lorsque la condition de la causalité est satisfaite, le système ANC est capable d'annuler le bruit. Néanmoins, si cette condition n'est pas satisfaite, le système ne peut contrôler efficacement que des bruits à bande étroite, ou des bruits périodiques.

- Effet du chemin secondaire :

Mais l'utilisation du filtre adaptatif pour un système ANC, est compliquée par le fait que la jonction additive de la Fig. 3.8 représente la superposition acoustique dans l'espace entre le haut-parleur et le microphone d'erreur, où le bruit primaire est combiné avec l'anti-bruit produit par le filtre adaptatif.

Par conséquent il est nécessaire de compenser la fonction de transfert du chemin secondaire $S(z)$ de $y(n)$ à $e(n)$, laquelle inclut (Fig. 3.9) : le convertisseur numérique-analogique, l'amplificateur de puissance, le haut-parleur, le chemin acoustique du haut-parleur au microphone d'erreur, le microphone d'erreur, le préamplificateur, les filtres anti-repliements, le convertisseur analogique-numérique, ...

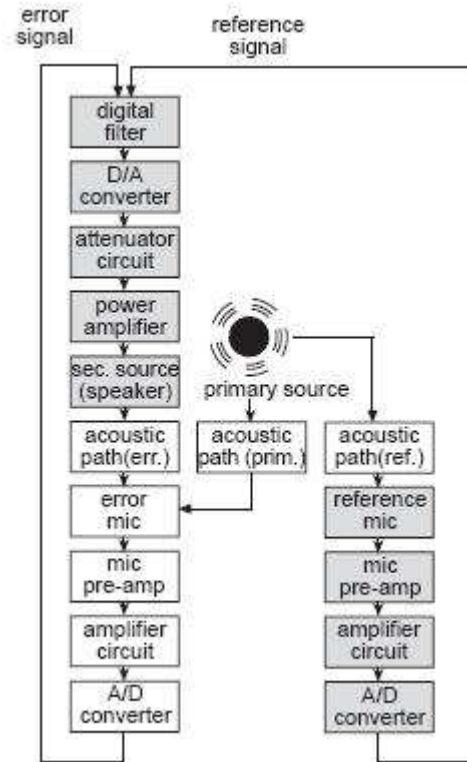


Fig. 3.9

Dans la figure suivante on peut voir un schéma équivalent à la Fig. 3.8, mais la jonction est maintenant électrique et non plus acoustique.

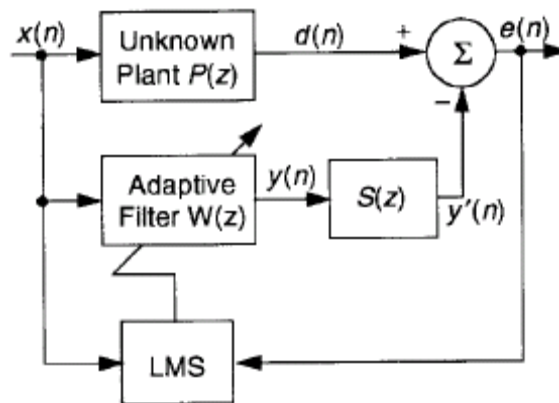


Fig. 3.10

- Le signal d'erreur devient :

$$E(z) = [P(z) - S(z) \cdot W(z)] \cdot X(z) \quad (\text{Eq. 3.10})$$

- Après la convergence du filtre adaptatif, et dans le cas idéal on a :

$$W(z) = \frac{P(z)}{S(z)} \quad (\text{Eq. 3.11})$$

En d'autres termes, le filtre $W(z)$ doit simultanément modéliser $P(z)$ et l'inverse de $S(z)$, pour pouvoir annuler le bruit.

L'avantage de cette approche est qu'avec un modèle adéquat du système, il peut répondre instantanément aux changements du signal d'entrée causé par les changements dans la source de bruit. Toutefois, les performances dépendent beaucoup de la fonction de transfert du chemin secondaire, $S(z)$.

Pour approximer cette dernière il faut utiliser un filtre RIF d'ordre élevé, vu que c'est une fonction rationnelle. Et il faut souligner, qu'il est impossible de compenser le retard inhérent dû à cette fonction de transfert, si le chemin principal $P(z)$ ne contient pas un délai d'au moins une longueur égale à ce retard.

L'introduction de la fonction de transfert du chemin secondaire dans un contrôleur qui utilise l'algorithme LMS standard (le plus utilisé en pratique) cause souvent une instabilité. Puisque le signal d'erreur n'est pas "aligné" correctement dans le temps avec le signal de référence, et cela est dû à la présence de $S(z)$. Pour compenser cet effet, il y a plusieurs manières :

- La première solution est de placer un filtre inverse, $1/S(z)$, en série avec $S(z)$ pour compenser son effet.
- La seconde est de placer un filtre identique sur le chemin du signal de référence pour mettre à jour les coefficients de l'algorithme LMS, cette méthode a pour appellation FxLMS, pour "Filtered-x LMS".

Comme un inverse n'existe pas nécessairement pour tout $S(z)$, l'algorithme FxLMS est généralement l'approche la plus efficace. Dans les applications ANC pratiques $S(z)$ est inconnu, et doit être estimé par un filtre supplémentaire $\hat{S}(z)$ (Fig. 3.11), généralement l'estimation est faite en "offline". Toutefois l'algorithme est très tolérant aux erreurs faites dans l'estimation de $S(z)$ par le filtre $\hat{S}(z)$.

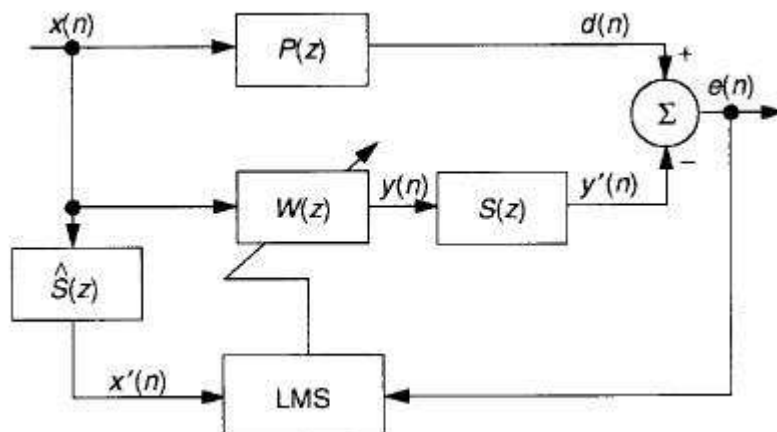


Fig. 3.11

- Effet de la contre-réaction :

La contre-réaction (ou "feedback") est la corruption du signal de référence capté par le premier microphone, par l'anti-bruit. Ci-dessous un schéma représentant un système ANC et incluant l'effet de la contre-réaction :

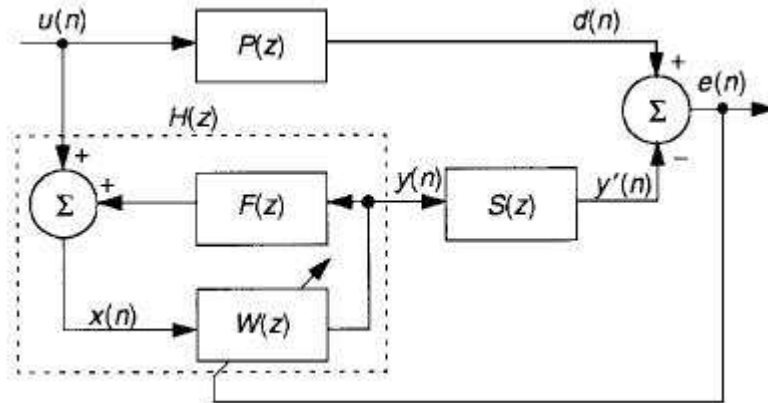


Fig. 3.12

Où : $u(n)$ est le bruit primaire, $x(n)$ est le signal capté par le microphone de référence, et $F(z)$ la fonction de transfert du chemin de contre-réaction de la sortie du filtre adaptatif $W(z)$ au premier microphone. La fonction de transfert du filtre adaptatif est :

$$W^o(z) = \frac{P(z)}{S(z) + P(z) \cdot F(z)} \quad (\text{Eq. 3.12})$$

De la Fig. 3.12, on peut déduire que la fonction de transfert en boucle ouverte de la boucle de contre-réaction est : $H_{BO}(z) = W(z) \cdot F(z)$; donc si le filtre converge vers la solution optimale :

$$H_{BO}(z) = \frac{P(z) \cdot F(z)}{S(z) + P(z) \cdot F(z)} \quad (\text{Eq. 3.13})$$

Cette dernière équation peut être utilisée pour tester la stabilité du système. Le système est instable quand la phase de la boucle ouverte atteint 180° et le gain de la boucle ouverte atteint l'unité (conditions de Barkhausen).

L'approche la plus simple pour résoudre le problème de la contre-réaction est l'utilisation d'un filtre de neutralisation, séparé dans le contrôleur, et c'est la même technique utilisée dans les systèmes d'annulation de l'écho acoustique. Ce modèle électrique de contre-réaction est commandé par le signal du chemin secondaire, et sa sortie est soustraite du signal du premier microphone. Un système ANC dans une canalisation, utilisant l'algorithme FxLMS avec neutralisation de la contre-réaction est illustré dans la figure suivante.

La contre-réaction du signal du microphone de référence est annulée électroniquement en utilisant une filtre de neutralisation de la réaction $\hat{F}(z)$, qui modélise la fonction de transfert du chemin de la réaction $F(z)$.

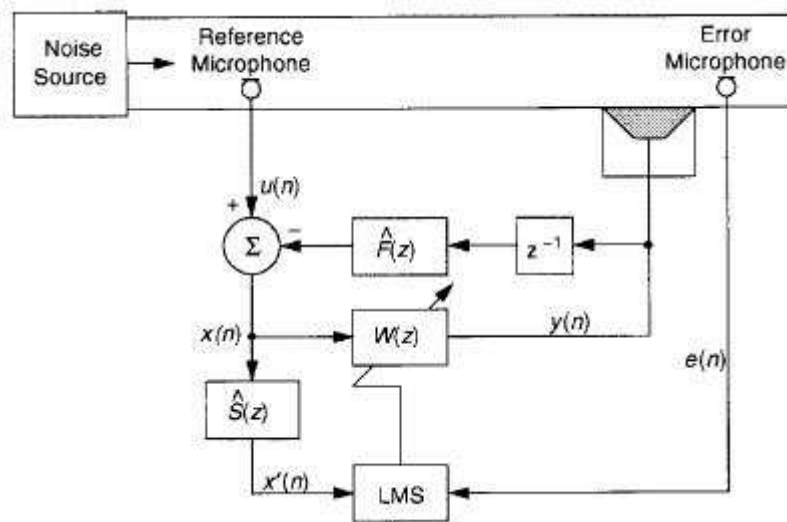


Fig. 3.13

Comme le bruit primaire est très corrélé avec l'antibruit, l'adaptation du filtre de neutralisation du "feedback" doit être inhibé, lorsque le système ANC est en fonctionnement, et ceci est aussi identique à l'annulation de l'écho acoustique.

En effet, la fonction de transfert du chemin du feedback est déterminée en utilisant une méthode adaptative "offline". Ainsi les modèles $\hat{S}(z)$ et $\hat{F}(z)$ peuvent être estimés simultanément en utilisant les techniques de modélisation "offline".

5- Les DSP : [L2]

5.1- Généralités :

Les processeurs dits DSP ("Digital Signal Processor") sont des microprocesseurs spécifiquement conçus pour le traitement des signaux digitaux.

Les premiers DSP ont été introduits en 1982 par *Texas Instruments*. Cette dernière contrôle environ 70% du marché mondial des DSP, le reste est partagé par : *Motorola*, *Analog Device*, *Lucent Technologies*, ...

On peut citer comme principaux avantages des DSP par rapport aux circuits analogiques "classiques" réalisant la même fonction :

- L'intégration, vu que ce sont des circuits numériques.
- Des possibilités propres au système de traitement numérique du signal. Car certaines fonctions de traitement du signal sont difficiles à implanter en analogique, voire irréalisables (ex. : un filtre à réponse en phase linéaire).
- La souplesse de programmation en utilisant divers langages de programmation (en particulier le C et l'Assembleur) ou même des programmes de simulation complets et de très haut niveau (*Matlab*).

- Stabilité, car en analogique les composants sont toujours plus ou moins soumis à des variations de leurs caractéristiques en fonction de la température, de la tension d'alimentation, du vieillissement, etc.
- Répétitivité et reproductibilité; alors que les valeurs des composants analogiques sont définies avec une marge de précision plus ou moins grande. Dans ces conditions, aucun montage analogique n'est strictement reproductible à l'identique, il existe toujours des différences qu'il convient de maintenir dans des limites acceptables.

Les particularités des DSP par rapport aux processeurs "généraux" (*Intel, AMD, Cyrix ...*) sont :

- Une architecture de type Harvard, séparant le bus du programme et le bus des données; et à jeu d'instruction réduit, RISC ("*Reduced Instruction Set Computer*").
- Des instructions adaptées aux opérations classiques de traitement de signaux : multiplication et accumulation en un seul cycle d'instruction (MAC), transferts de données par DMA ("*Direct Access Memory*"), FFT ("*Fast Fourier Transform*"), convolution, etc.
- Un adressage circulaire, permettant de ne pas faire des tests de modulo sur les pointeurs de tampons.
- Un mode d'adressage par inversion de bits, servant à réorganiser les échantillons de sortie des FFT.
- De la mémoire interne minimisant les temps d'accès RAM ("*Random Access Memory*").
- Des banques de mémoire, permettant de scinder par exemple partie réelle et partie imaginaire.
- Une architecture à pipe-line permettant de paralléliser le maximum d'opérations élémentaires (pré-chargement d'instruction, chargement des données, exécution d'opérations arithmétique/flottante, stockage des résultats, ...).
- Des ADC ("*Analog-to-Digital Converter*") et des DAC ("*Digital-to-Analog Converter*") intégrés.

Les capacités des DSP sont de plus en plus importantes, et les prix de ces processeurs ont diminué ces dernières années. On en utilise donc de plus en plus. La liste suivante, non exhaustive, donne un aperçu des fonctions et applications les plus courantes des DSP dans quelques domaines :

- Le militaire : guidage des missiles, navigation, cryptage, traitements des signaux radars et sonars, ...
- Le grand public : compression des signaux audio (CD), cartes multimédias pour PC, synthèse musicale, jeux, ...
- Télécommunications : modems, multiplexeurs, codeurs de paroles, télécopieurs, téléphonie filaire et cellulaire, routeurs, VoIP ("*Voice over IP*"), ...

- Le médical : compression et traitement des images (ex. : IRM, échographie), traitement des signaux physiques (ex. : ECG, EEG), ...
- L'automatisme : asservissement, diagnostic automatique, ...
- L'instrumentation : analyse spectrale, oscilloscope, générateur de signaux, ...
- L'automobile : équipements de contrôle moteur, aide à la navigation, commande vocale, ...

5.2- Les différents types de DSP :

On peut distinguer deux familles de DSP, selon le format de représentation des données : les DSP à virgule flottante et ceux à virgule fixe.

5.2.1- DSP à virgule flottante :

Les données sont représentées en utilisant une mantisse et un exposant. La représentation de ces nombres s'effectue selon la formule suivante : $n = \text{mantisse} \times 2^{\text{exposant}}$. Habituellement, la mantisse est un nombre fractionnaire (-1.0 à +1.0), et l'exposant est un entier indiquant la place de la virgule en base 2.

La très grande dynamique proposée par les DSP à virgule flottante permet virtuellement de ne pas se soucier des limites des résultats calculés lors de la conception d'un programme. Cet avantage a cependant un prix, à savoir qu'un système basé sur un DSP à virgule flottante a un coût de fabrication supérieur par rapport à un système basé sur DSP à virgule fixe; et la puce d'un DSP à virgule flottante nécessite à la fois une surface de silicium plus importante (cœur plus complexe), et un nombre de broches supérieur.

Les DSP à virgule flottante sont plus souples et plus faciles à programmer que les DSP à virgule fixe, mais ils sont plus onéreux et consomment plus.

5.2.2- DSP à virgule fixe :

Les données sont représentées comme étant des nombres fractionnaires à virgule fixe, ou comme des entiers classiques. La représentation de ces nombres fractionnaires s'appuie sur la méthode du complément à deux. Le principal avantage de cette représentation est de permettre facilement l'addition binaire de nombres aussi bien positifs que négatifs.

Un DSP à virgule fixe est un peu plus compliqué à programmer qu'un DSP à virgule flottante car la précision des calculs est un point critique, le concepteur de programmes doit rester vigilant à chaque étape d'un calcul. Il doit rechercher la plus grande dynamique possible (c'est-à-dire exploiter au mieux la gamme des nombres disponibles), pour conserver une bonne précision des calculs, tout en évitant autant que faire se peut les débordements du ou des accumulateurs. Les bits supplémentaires des accumulateurs (les bits de garde) prévus à cet effet permettent de réduire cette contrainte.

Les programmeurs contournent les limites des DSP à virgule fixe en déterminant à l'avance, et avec soins, la précision et la dynamique nécessaire (par des méthodes analytiques ou avec des outils de simulation) pour réaliser leurs projets.

Il est également possible d'effectuer des opérations en virgule flottante dans un DSP à virgule fixe par le biais de routines logicielles adéquates. Cette approche est néanmoins pénalisante en temps d'exécution, même sur un DSP à virgule fixe très rapide.

Les DSP à virgule fixe sont les plus utilisés, car ils sont moins chers et plus rapides que les DSP à virgule flottantes. On les trouve dans tous les produits de grande diffusion où le coût est un facteur important. Il peut cependant exister des exceptions, certains DSP à virgule fixe se présentant comme des micro contrôleurs perfectionnés plus chers qu'un DSP à virgule flottante de base.

5.3- Le choix d'un DSP :

La sélection d'un DSP se base avant tout sur la puissance de traitement nécessaire, toutefois, la performance du DSP n'est pas le seul critère à prendre en compte, il faut également tenir compte des impératifs suivants :

- Le type de DSP à utiliser (virgule fixe ou flottante) en fonction du domaine d'application.
- Les ressources mémoires utilisées.
- Les besoins spécifiques à l'application (d'un ou de plusieurs timers internes, de ports série synchrones ou asynchrone, etc).
- La nécessité éventuelle d'exécuter un système en temps réel.
- Le coût du DSP, et son rapport performance/prix en fonction du volume de production envisagé.
- La qualité de la documentation, et la disponibilité de notes d'applications, d'un support technique.
- La qualité du système de développement utilisé.
- La possibilité d'utiliser un langage de haut niveau.
- La présence de bibliothèques (du constructeur ou de tierces parties).
- La possibilité éventuelle de réaliser facilement des prototypes à faible coût.

Le choix n'est pas toujours simple et certains critères peuvent être contradictoires, certaines règles de choix se dégagent quand même. Ainsi pour des applications destinées à faire un fort volume de production, le critère déterminant est sans conteste le prix du DSP. Pour des applications à faible volume de production, le prix du DSP importe peu, le critère est alors la facilité de développement.

Dans tous les cas, la présence d'un bon support technique est un facteur à ne pas négliger, car un DSP est quand même plus complexe à mettre en œuvre qu'un microprocesseur classique.

En faisant une recherche sur le marché des DSP, on a trouvé que les produits de *Texas Instruments* (les séries C2000 et C6000), sont ceux qui présentent une très grande facilité de programmation par rapports aux autres produits des autres constructeurs; parce qu'au lieu d'utiliser le langage C, on peut utiliser le logiciel *Matlab* pour les programmer, la programmation des DSP avec ce dernier, se fait de manière très intuitive (interface graphique) et de façon totalement transparente pour l'utilisateur (blocks schématisant les fonctions).

Notre choix s'est porté sur la famille C6000, vu que la série C2000 (bas de gamme) est à virgule fixe et a des performances limitées. La série C6000 a, jusqu'à présent, trois DSK ("*DSP Starter Kit*") : C6713, C6416 et C6455. Nous avons choisi d'utiliser le C6713, vu que les deux autres DSK coûtent plus chers (495 \$ contre 395 \$) et sont surdimensionnés par rapport à notre application. Nous présenterons plus en détails le DSP en Annexes.

IV)- Étude Pratique

1- Préambule :

Nous allons présenter dans ce chapitre les simulations et les applications qu'on a faites, mais nous n'avons pas pu terminer entièrement notre travail, car nous avons rencontré des difficultés qu'on n'est malheureusement pas arrivé à surmonter, dans la modélisation de la fonction de transfert du chemin secondaire, et comme on la déjà vu dans le chapitre précédent, le rôle du chemin secondaire est primordial pour le bon fonctionnement du système ANC.

2- Bruit utilisé :

Pour faire nos simulations et expérimentations, nous avons enregistré le bruit généré par des appareils installés par l'entreprise, afin d'étudier sa nature et aussi de le générer à l'aide d'un haut parleur.

Dans la figure suivante, on peut voir le spectre du bruit (avec une échelle logarithmique), tracé à l'aide du logiciel gratuit du traitement du son : *Audacity 1.2.4*.

Le bruit peut être, à priori, annulé par l'ANC, vu que qu'il est totalement aléatoire (à large bande), et à basse fréquence; car comme on le voit, il peut être considéré comme étant nul à partir de 4 kHz.

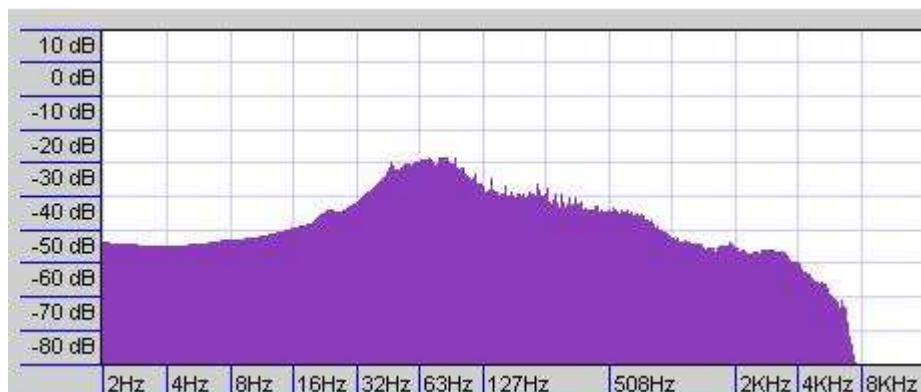


Fig. 4.1

3- Simulations :

3.1- Remarque :

La nouvelle version de *Matlab* : 2006a, qu'on utilise dans nos applications, fournit de nouveaux blocs de filtres adaptatifs ne permettant pas d'implémenter directement des applications ANC; vu qu'au lieu de pouvoir, comme dans les versions précédentes, les entrées : "input" et "error" des filtres adaptatifs, elle fournit : "input" et "desired".

Dans l'ancienne bibliothèque de filtres adaptatifs de *Matlab* (Fig. 4.2), l'entrée "input" représente le bruit primaire ($x(n)$ dans la Fig. 3.6); et l'entrée "error" le bruit résiduel ($e(n)$ dans la Fig. 3.6).

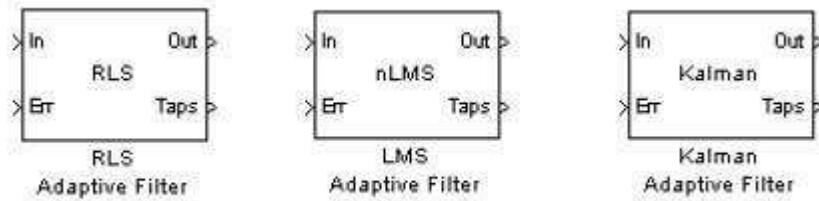


Fig. 4.2

Cependant dans la nouvelle bibliothèque de filtres adaptatifs (Fig. 4.3), l'entrée "input" représente toujours le bruit primaire; mais l'entrée "error" a été remplacée par "desired" ($d(n)$ dans la Fig. 3.8).

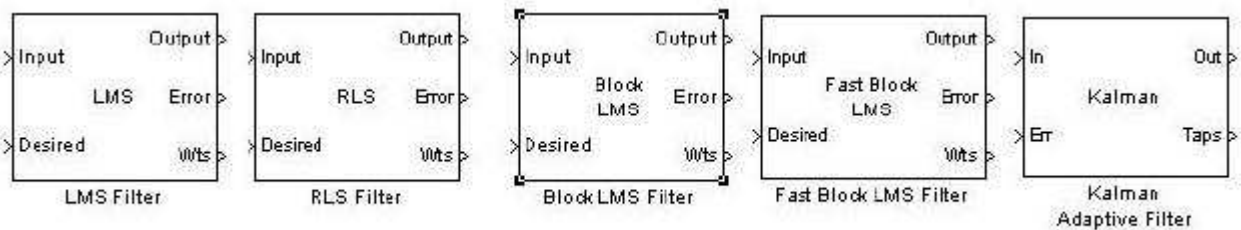


Fig. 4.3

Or ce qu'on veut faire, c'est utiliser l'entrée "error" et non pas "desired", car l'opération de soustraction va se faire physiquement (dans le domaine acoustique). Comme *Matlab 2006a* reconnaît toujours les filtres adaptatifs de l'ancienne bibliothèque de la version précédente (*Matlab 6.5*), on a utilisé ces filtres (LMS et RLS), avec la nouvelle version du *Matlab 2006a*.

3.2- Résultats :

On expérimenté un schéma ANC très simple, avec des retards pour représenter le retard acoustique, en utilisant l'algorithme NLMS (Fig. 4.4) puis RLS (Fig. 4.5) :

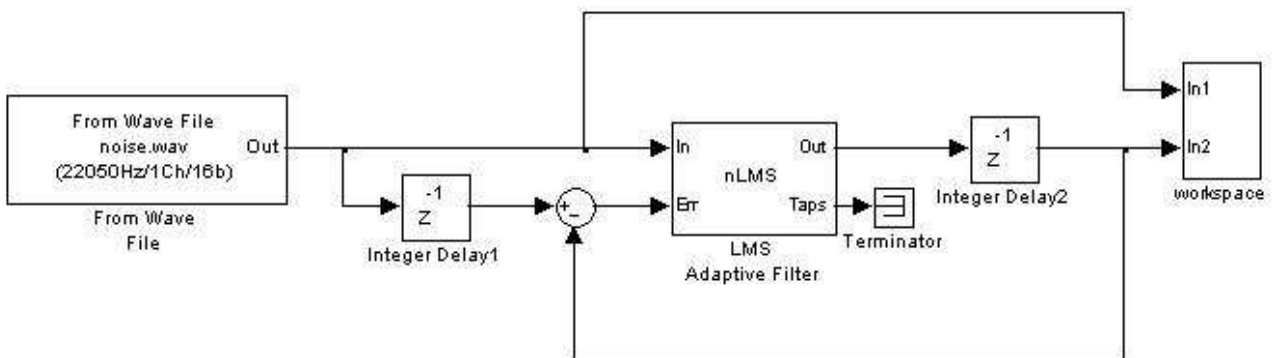


Fig. 4.4

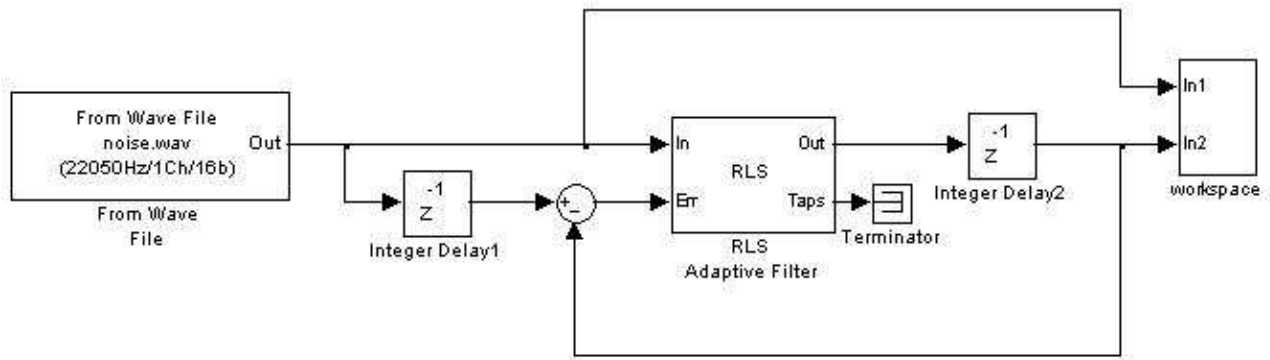


Fig. 4.5

Avec des filtres FIR à 32 coefficients, on obtient les deux figures de la page suivante : Fig. 4.6 pour le NLMS et la Fig.4.7 pour le RLS.

Après avoir fait plusieurs simulations, que nous n'allons pas reproduire ici faute de place, et en "jouant" sur plusieurs paramètres : gains, durées des retards, nombre de coefficients des filtres, le pas d'adaptation (filtre LMS), le facteur d'oubli (filtre RLS), etc. Nous avons trouvé que nos résultats concordent parfaitement avec la théorie des filtres adaptatifs, présentée au chapitre précédent, vu que la convergence du filtre RLS est plus rapide que celle du LMS, mais le calcul de ce dernier prends beaucoup moins de temps et nécessite moins de ressources.

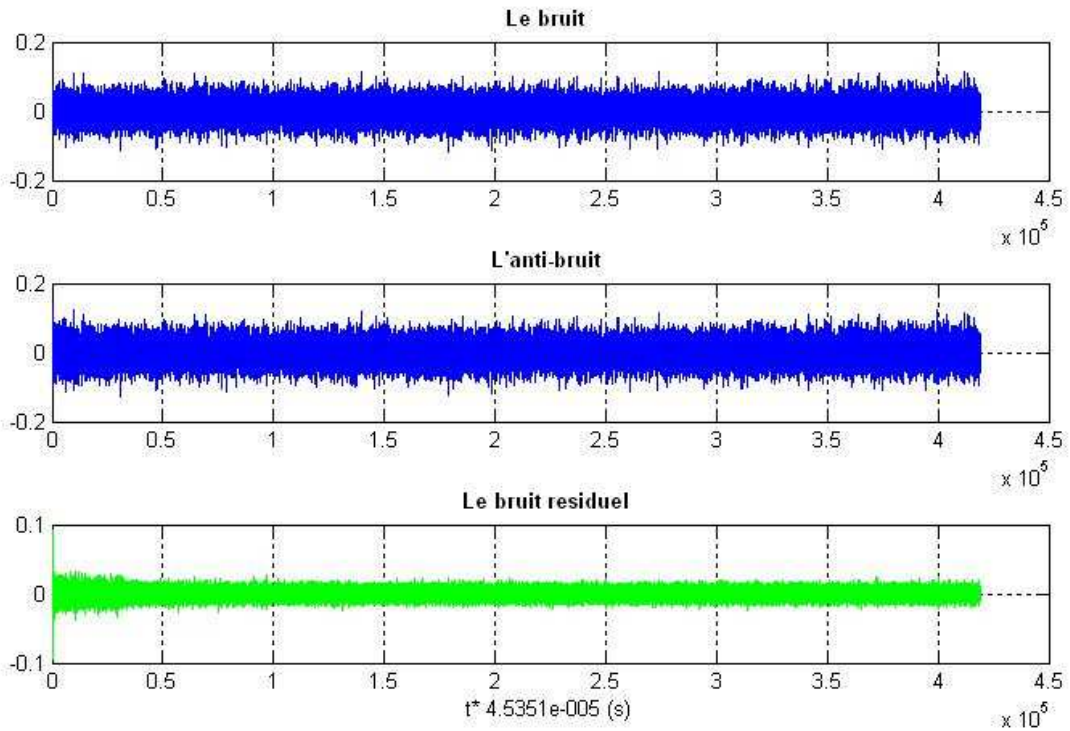


Fig. 4.6

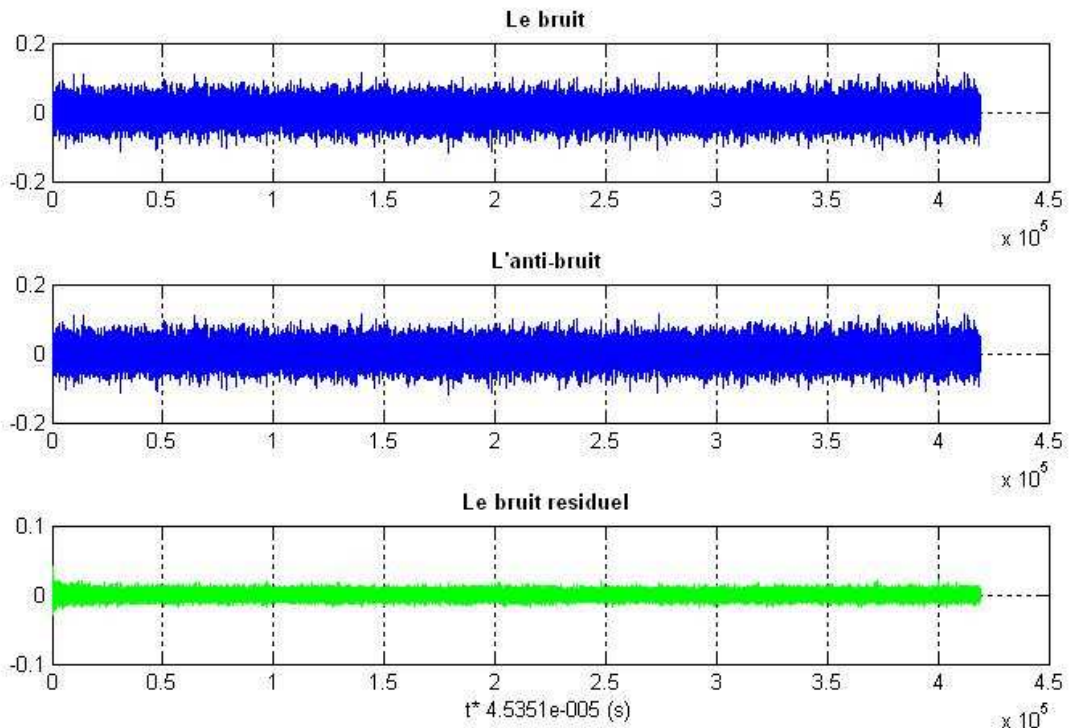


Fig. 4.7

3.3- L'application :

Comme on l'a déjà énoncé au début de ce chapitre, on n'a pas pu terminer complètement notre application, ainsi nous n'allons pas présenter le programme vu qu'il lui manque toute la partie : modélisation de la fonction de transfert du chemin secondaire; mais nous allons quand même décrire son fonctionnement.

Nous avons pensé à utiliser la méthode des SBPA (Séquence Binaire Pseudo Aléatoires), pour la détermination de la fonction de transfert du chemin secondaire, c'est-à-dire envoyer un bruit blanc sur le haut parleur ($x(t)$) puis capter le signal reçu à l'aide du microphone d'erreur ($y(t)$) et avec les deux signaux faire le calcul de la fonction de transfert ($h(t)$).

Mais on n'a pas pu faire ça en pratique, car on n'a pas su comment programmer le calcul de la fonction de transfert sur le DSP, quelle taille de mémoire utiliser (car il faut mémoriser $h(t)$ dans la mémoire de le DSP pour pouvoir l'utiliser), combien durera le calcul de la fonction de transfert, etc.

Généralement le calcul de la fonction de transfert se fait de manière automatique juste après la mise en marche du système et avant le démarrage de l'ANC, (les deux ne peuvent pas fonctionner en même temps). Mais dans les systèmes évolués, à intervalles réguliers et de manière automatique, le système ANC s'arrête et le système calcule de nouveau la fonction de transfert, pour tenir compte des variations physiques du chemin secondaire (humidité, chaleur, ...).

Dans notre programme, on a utilisé le DIP switch 0 pour faire basculer le système entre les deux fonctions (ANC et modélisation de la fonction de transfert du chemin secondaire). Quand le DIP switch 0 est en position haute, c'est la modélisation qui fonctionne, et quand on appuie dessus la modélisation s'arrête pour laisser place à l'ANC.

Nous avons utilisé la sortie (stéréo) "line out", pour émettre l'anti-bruit (système ANC) et aussi la SBPA (modélisation de la fonction de transfert du chemin secondaire); et on a utilisé l'entrée (stéréo) "line in" du DSK, on a mis sur la voie droite le microphone captant le bruit primaire (système ANC), et sur celle de gauche le microphone d'erreur (système ANC), et c'est ce même microphone qui captera la réponse du chemin secondaire au SBPA.

V)- Conclusion

Ce stage nous a permis de comprendre trois domaines importants du traitement du signal; l'utilisation pratique des filtres adaptatifs, l'application des systèmes ANC, et le fonctionnement et la mise en place d'un DSP.

Bien qu'on n'ait pas pu mener le projet à terme, par manque d'expériences et de connaissances dans le domaine, le programme est presque terminé, il ne lui manque que la partie : modélisation du chemin secondaire, oh combien nécessaire au fonctionnement des systèmes ANC performants.

Annexes :

- Présentation du DSK C6713 :

Le nom complet du DSK C6713, conçu par *Texas Instruments*, est : TMDSDSK6713, le kit et la carte contiennent :

- Un câble USB pour connecter la carte DSK au PC.
- Une alimentation fournissant du 5 V à partir du secteur.
- Le processeur TMS320C6713, à virgule flottante, avec 512 K de mémoire Flash et 8 MB de mémoire SDRAM (voir le schéma de la Fig. A.1 pour la disposition de la mémoire), le processeur fonctionne à 225 MHz, et peut atteindre 1800 MIPS et 1350 MFLOPS (voir le tableau de la Fig. A.2 pour les définitions des unités).
- Un codec audio stéréo à 24 bits, pouvant fonctionner aux fréquences d'échantillonnage suivantes : 8 kHz; 32 kHz; 44,1 kHz; 48 kHz et 96 kHz.
- Quatre prises audio jacks de 3,5 mm, deux pour les entrées : microphone (mono) et "line in" (stéréo), et deux pour les sorties : "speaker" (stéréo) et "line out" (stéréo). Réellement les deux entrées (respectivement les deux sorties) renvoient au même port physique, c'est-à-dire au même signal d'entrée (respectivement de sortie), la seule différence entre microphone et "line in" (respectivement "speaker" et "line out") réside dans les impédances des ports. Autrement dit, on a quatre prises audio, mais une seule entrée et une seule sortie, chacune disponible avec deux impédances différentes (Fig. A.3).
- Le logiciel complet *CCS (Code Composer Studio) 3.1*, pour éditer, compiler, debugger, linker, etc, les programmes.
- Le *Matlab 6.5*, incomplet et en version d'essai limitée à 30 jours.

Le DSK permet l'utilisation (Fig. A.4) : du protocole de transfert de données en temps réel entre le DSP et le PC, RTDX ("*Real Time Data Exchange*"); du protocole de contrôle des périphériques d'E/S, JTAG ("*Joint Test Action Group*"); etc.

La carte comporte (Fig. A.5) : un emplacement libre pour l'ajout d'un périphérique ou d'une carte additionnels ("*daughter card*"), un emplacement pour ajouter de la mémoire, quatre LED ("*Light-Emitting Diodes*") programmables, quatre DIP switches ("*Dual In-line Pin*") programmables, etc.

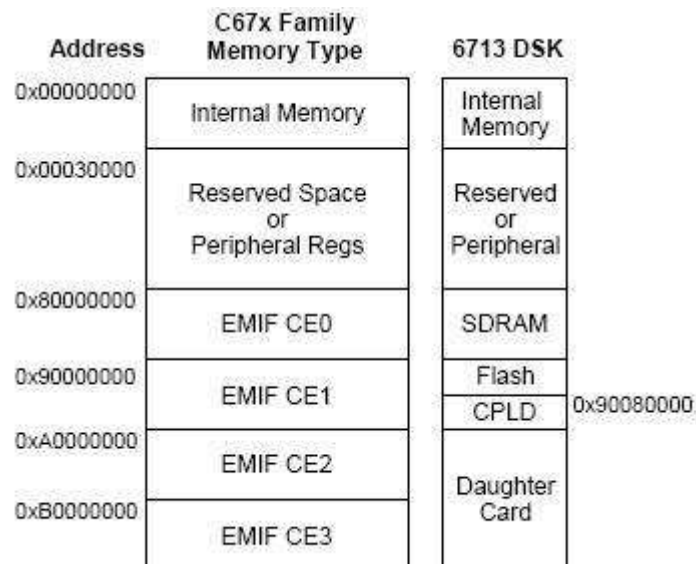


Fig. A.1

<u>Acronyme anglais :</u>	<u>Définition :</u>
MFLOPS	<p><i>Million Floating-Point Operations Per Second.</i></p> <p>Mesure le nombre d'opérations à virgule flottante (multiplications, additions, soustractions, etc.) que le DSP à virgule flottante peut réaliser en une seconde.</p>
MOPS	<p><i>Million Operations Per Second.</i></p> <p>Mesure le nombre total d'opérations que le DSP peut effectuer en une seconde. Par opérations, il faut comprendre non seulement le traitement des données, mais également les accès DMA, les transferts de données, les opérations d'E/S, etc. Cette définition mesure donc les performances globales d'un DSP plutôt que ses seules capacités de calcul.</p>
MIPS	<p><i>Million Instructions Per Second.</i></p> <p>Mesure le nombre de codes machines (instructions) que le DSP peut effectuer en une seconde. Bien que cette mesure s'applique à tous les types de DSP, le MFLOPS est préféré dans le cas d'un DSP à virgule flottante.</p>
MBPS	<p><i>Mega-Bytes Per Second.</i></p> <p>Mesure la largeur de bande d'un bus particulier ou d'un dispositif d'E/S, c'est à dire son taux de transfert.</p>

Fig. A.2

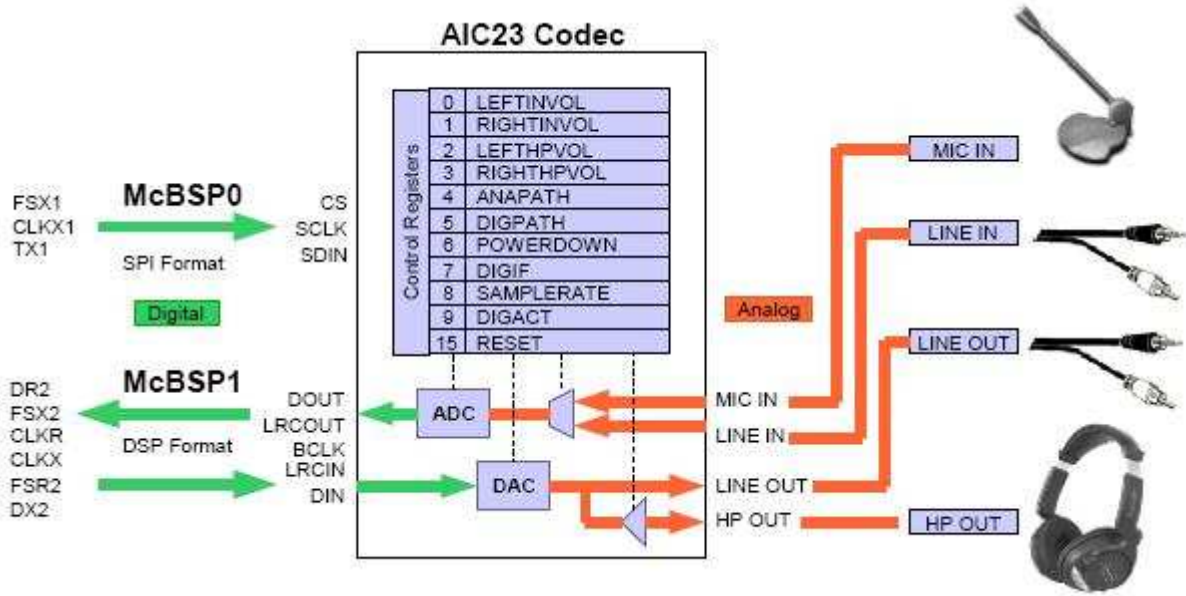


Fig. A.3

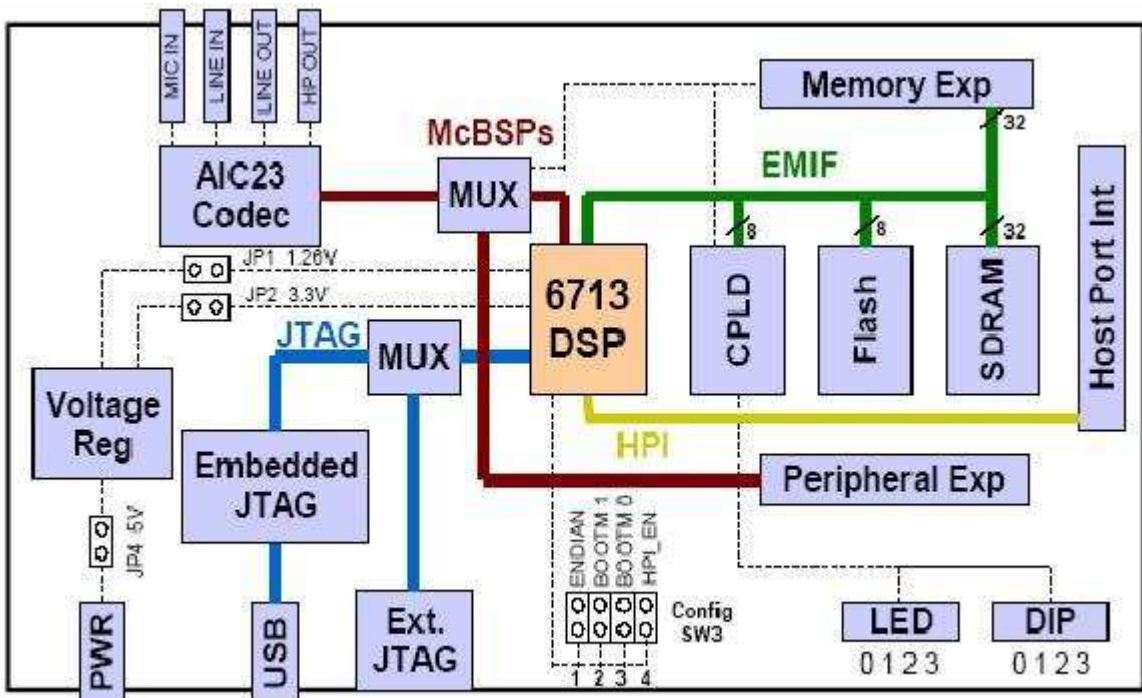


Fig. A.4

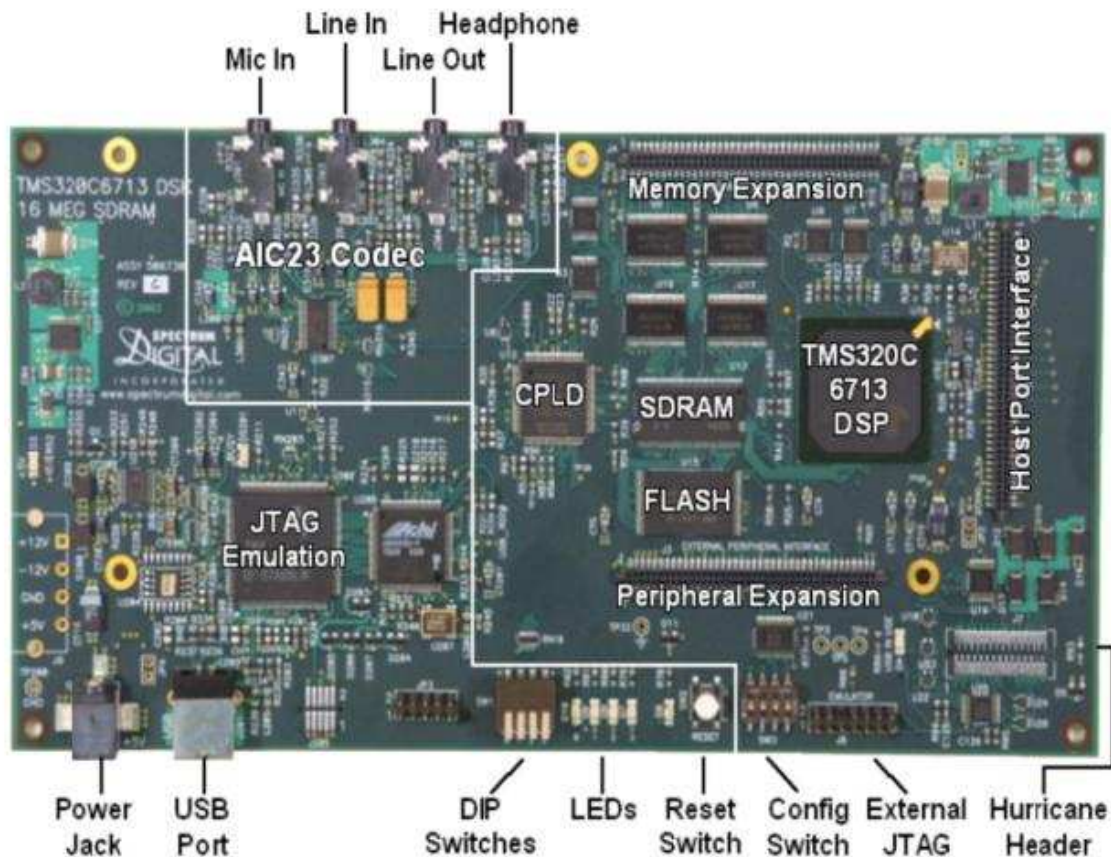


Fig. A.5

- La programmation du DSP :

La programmation du DSP passe obligatoirement par le *CCS*; soit de manière directe quand on programme directement avec l'interface graphique du *CCS* le DSP en C ou en Assembleur (Fig. A.6); soit de manière indirecte quand on programme le DSP avec le *Matlab*, comme c'est le cas pour notre travail (Fig. A.7).

Dans ce dernier cas, la programmation se fait d'une manière très intuitive et totalement transparente pour l'utilisateur, et c'est le *Matlab* qui génère automatiquement le code C pour le *CCS* (Fig. A.8).

Dans tous les cas, le compilateur C compile un programme de source écrit en C avec extension ".c" afin d'engendrer un fichier en Assembleur, avec une extension ".asm". L'Assembleur assemble la source ".asm" pour produire un fichier d'objets en langage machine avec une extension ".obj". Le "linker" associe les fichiers d'objets et les bibliothèques d'objets comme entrées, pour générer un fichier exécutable avec l'extension ".out". Ce dernier exécutable représente un fichier au format COFF ("*Common Object File Format*"), populaire dans les systèmes à base d'UNIX et adopté par plusieurs fabricants de DSP. Il peut être chargé et exécuté directement sur le processeur du DSP (Fig. A.9).

Il y a aussi la possibilité d'utiliser le fichier source "*linear assembly*" avec extension ".sa" qui est un mélange entre le C et l'Assembleur. Un "optimiseur" linéaire optimise ce fichier source pour créer un fichier assembleur avec une extension ".asm" (exactement comme pour le compilateur C).

Plusieurs options sont disponibles lors de la compilation d'un "projet", par exemple en utilisant le RTDX : l'analyse en temps réel des performances du DSP, de sa mémoire, des variables utilisées, etc.

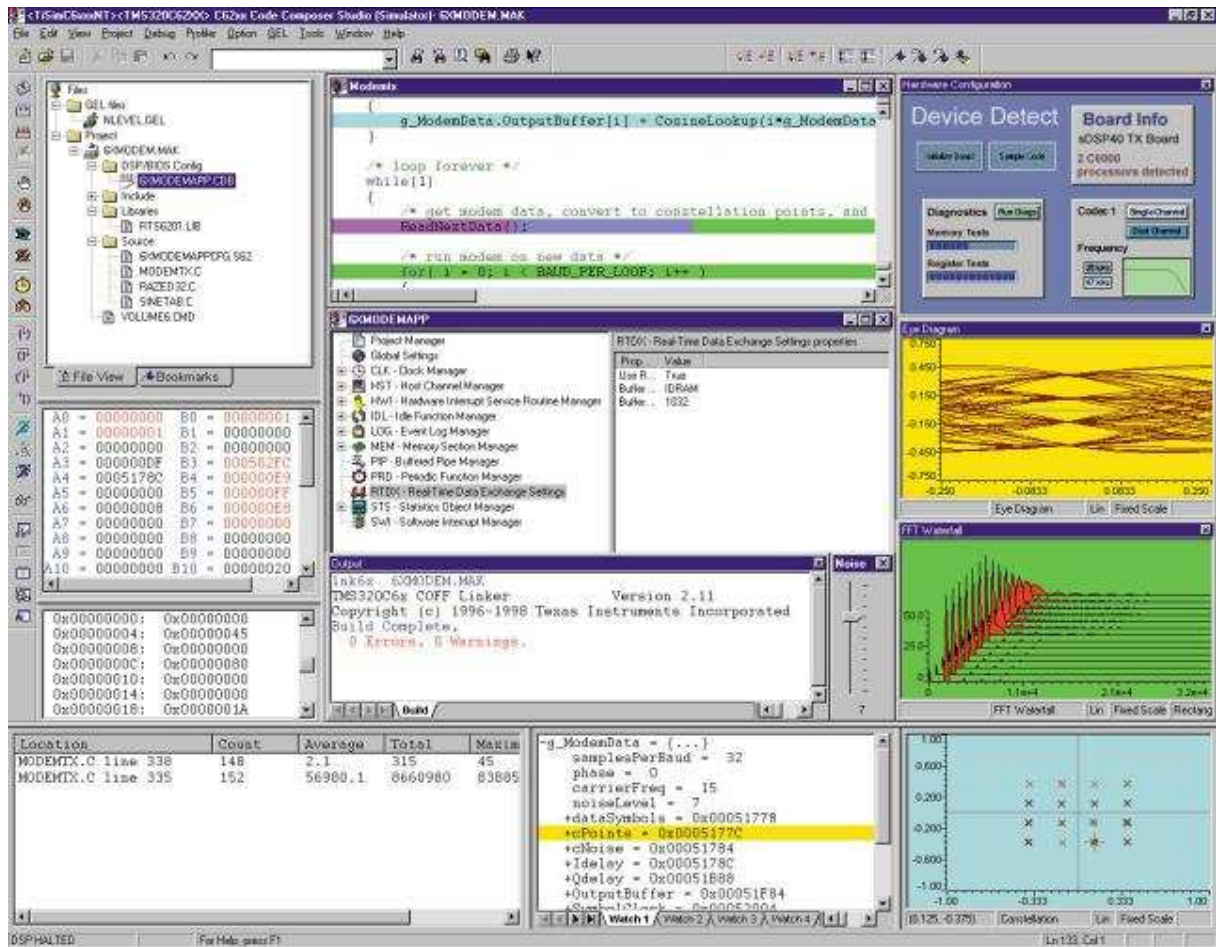


Fig. A.6

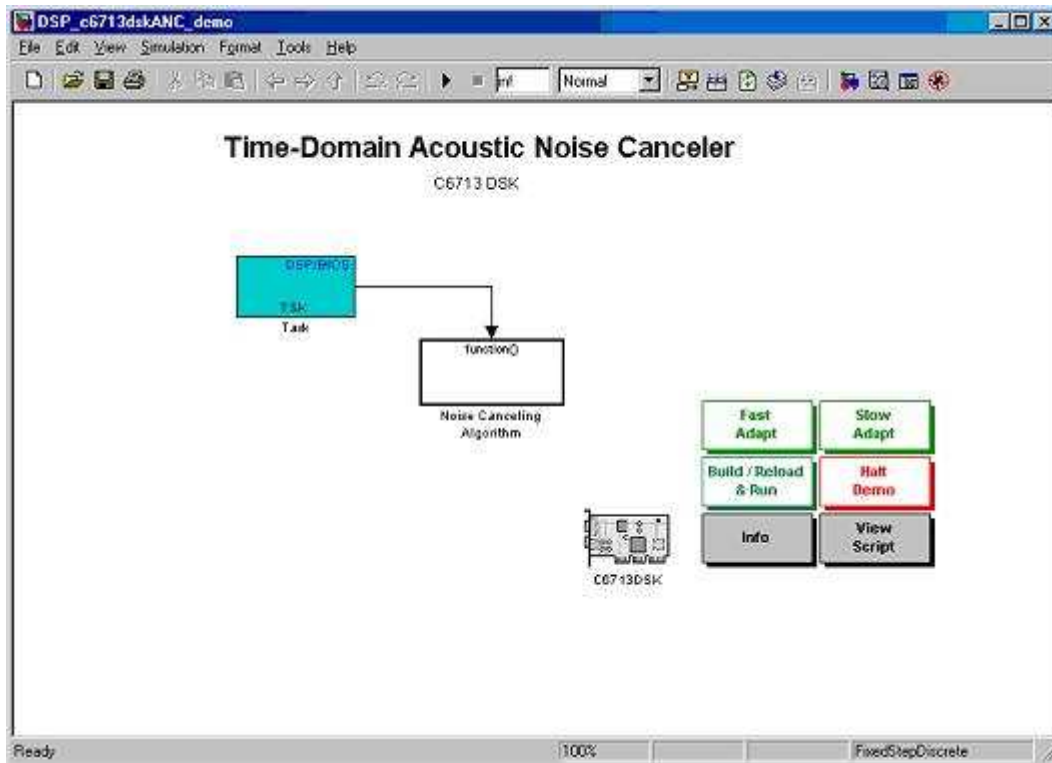


Fig. A.7

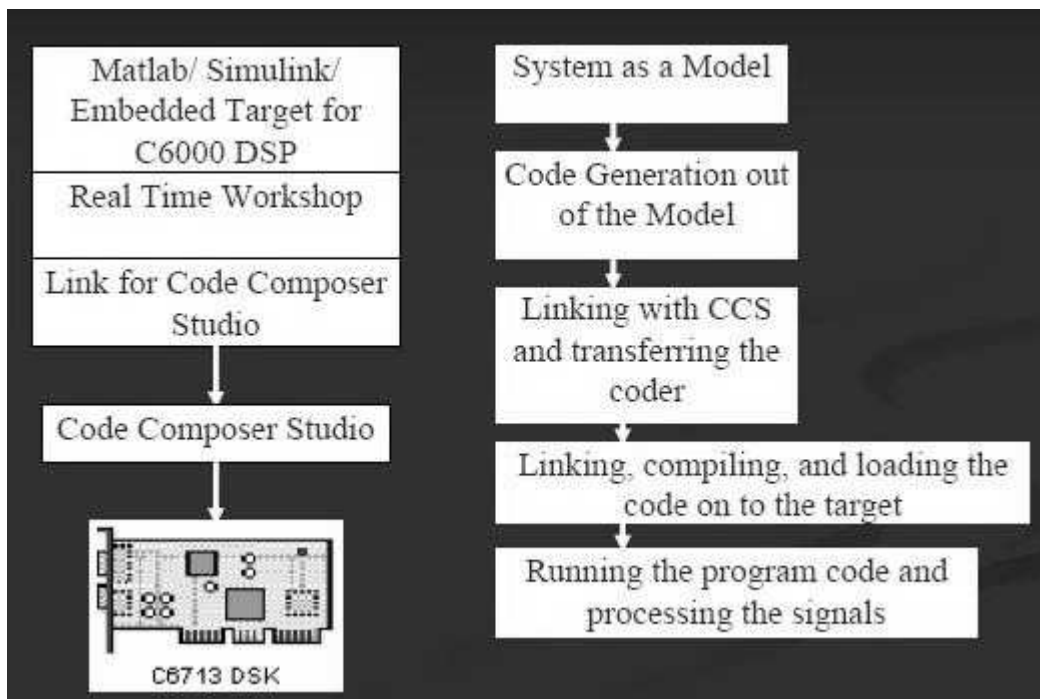


Fig. A.8

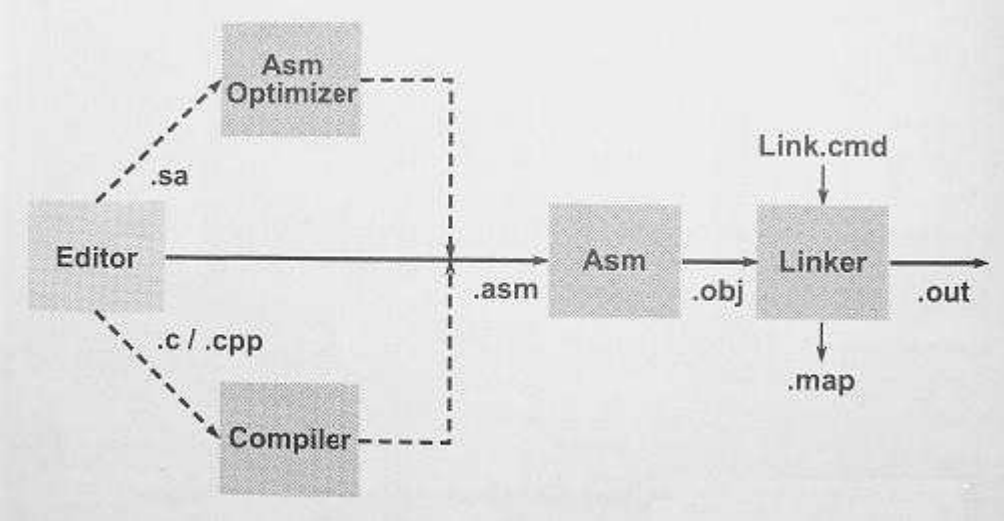


Fig. A.9

Bibliographie :

- Livres :

[L1] : Le Livre des techniques du son. (Tome 1, 3ème édition)
Collectif d'auteurs sous la direction de *D. Mercier*.
Ed. Dunod, 2002.
ISBN : 2100063049.

[L2] : Real-Time Digital Signal Processing - Implementations and Applications.
(2ème édition)
S. Kuo, B. Lee et W. Tian.
Ed. Wiley, 2006.
ISBN : 0470014954.

- Rapports :

[R1] : Active noise control.
Aleksandar Milosevic et Urs Schaufelberger.
Rapport du PFE en Électrotechnique (2005).
University of Applied Sciences Rapperswil HSR (Suisse).

[R2] : Active noise control in an airflow.
Carl Enander.
Rapport du PFE en Master (2005).
Uppsala University (Suede).

[R3] : Investigation in active noise control techniques for headsets.
Cédric Thiery.
Rapport du DEA STIR (1999).
INSA de Rennes (France).

[R4] : Développement et optimisation d'un système de contrôle actif pour les absorbants acoustiques hybrides.
Olivier Hilbrunner.
Thèse de Doctorat en Acoustique (2003).
École Centrale de Lyon (France).

[R5] : Characterization of active sound cancellation zone of silence in reverberant enclosure with periodic disturbance.
Eileen Anderson.
Rapport du PFE en Master (2004).
University of Arkansas (EU).

- Publications :

[P1] : Active noise control: A tutorial review.

S. M. Kuo et D. R. Morgan.

Proc. of the IEEE, vol. 87, no. 6, pp. 943 - 973, 06/1999.

[P2] : Active noise control system for headphone applications.

S. M. Kuo, S. Mitra et W. S. Gan.

IEEE Trans. on Control Systems Technology, vol. 14, no. 2, pp. 331 - 335, 03/2006.

[P3] : On use of averaging in FxLMS algorithm for single-channel feedforward ANC systems.

M. T. Akhtar, M. Abe, et M. Kawamata.

Proc. IEEE ISCAS 2003, vol. 4, pp. 389 - 392, 05/2003.

[P4] : Modified-filtered-x LMS algorithm based active noise control system with improved online secondary-path modeling.

M. T. Akhtar, M. Abe, et M. Kawamata.

Proc. IEEE 2004 Int. Mid. Symp. Circuits Systems (MWSCAS2004), Hiroshima (Japan), pp. I-13 - I-16, 07/2004.

[P5] : A new method for active noise control systems with online acoustic feedback path modeling.

M. T. Akhtar, M. Abe, et M. Kawamata.

IEEE Transactions on Speech and Audio Processing, vol. 13, no. 5, part 2, pp. 1082 - 1088, 09/2005.

[P6] : Design of robust controller for the active noise control system.

H. Hamidi, F. Taringoo et A. Nasiri.

Control Conference, 2004. 5th Asian, vol. 2, pp. 1162 - 1167, 07/2004.

- Divers sites et documents Internet :

www.serdb.com/gloss.html#

www.faqs.org/faqs/active-noise-control-faq

freespace.virgin.net/sanjeev.sarpal/adaptive%20filter.htm

www.igi.tugraz.at/lehre/CI/tutorials/AdaptiveFilter/node3.html

<http://perso.orange.fr/lapiste/DSP/dsp.htm>

<http://b.l.free.fr/index.html>

www.ti.com

Abstract:

The purpose of this humble work is the conception and the realization of an anti-noise system for fan devices, that includes: firstly the study of the whole anti-noise theoretical background, and also the choice of the algorithm, the method and the DSP to use in our application; then the simulation with *Matlab*, and finally the practical realization of the system.

Résumé :

Le but de ce modeste travail est la conception et la réalisation d'un système anti-bruit pour des systèmes de ventilation, ceci englobe : en premier lieu l'étude théorique complète du contrôle actif du bruit, et aussi le choix de l'algorithme, de la méthode et du DSP à utiliser dans notre application; puis la simulation sous *Matlab*, et finalement la réalisation pratique du système.