DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DESIGN AND IMPLEMENTATION OF A DSP BASED ACTIVE NOISE CONTROLLER

by
Hakan KAHRAMAN

July, 2014
İZMİR

DESIGN AND IMPLEMENTATION OF A DSP BASED ACTIVE NOISE CONTROLLER

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Master of Science of
Electrical and Electronics Engineering

by
Hakan KAHRAMAN

July, 2014
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"DESIGN AND IMPLEMANTATION OF A DSP BASED ACTIVE NOISE CONTROLLER"** completed by **HAKAN KAHRAMAN** under supervision of **ASS. PROF. HATİCE DOĞAN** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Hatice DOĞAN

Supervisor

Assist. Prof. Dr. Gülese K. Demir

(Jury Member)

Assoc. Prof. Dr. Zeki KIRAL

(Jury Member)

Prof.Dr. Ayşe OKUR
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

I would like to thank my supervisor, Hatice DOĞAN, for providing me the opportunity to pursue and finish this degree, her valuable knowledge and suggestions helped me a lot in the project.

I would also like to thank TÜBİTAK for their financial support throughout the master.

This thesis was supported by 2012.KB.FEN.051 numbered DEU-BAP project.

Hakan KAHRAMAN

DESIGN AND IMPLEMENTATION OF A DSP BASED ACTIVE NOISE CONTROLLER

ABSTRACT

Noise is a general name for all disturbing sounds. Noise not only affects livings, but also affects the machines in a bad way. The continuous increase in human population and developing technological needs resulted in increasing noise and noise variety. Therefore, noise cancellation methods have been spread. Different kinds of headphones are used for sharp (high frequency) noises, while active noise cancellation methods are used for bass (low frequency) noises. The application areas of these active noise controls vary from infant incubators to automobile cabins, and from plane cabins to hospital environments. Here, the noises have low frequency components. Application of active noise controls on high frequency noises is not well developed. In this project, active noise cancellation is used for the dentist drill sound, which has very high frequency components. During the project, MATLAB software has been used. Firstly the active noise control algorithm is designed in the simulation environment. Optimum filter parameters are determined. Then after the system performance is justified, hardware parts are implemented. In the hardware setup, drill sound is recorded via microphones and the negative sound of this noise is transmitted to the room through a speaker. In this way, theorically, the total environment noise is reduced and diminished with a particular percentage. The hardware setup in this project achieved this theory in the practical life. The noise reduction is around fifty percent in a specific area of the room. These results are compared with other ANC methods used in this project, and the corresponding graphs are added to this report. The advantages and the disadvantages of these ANC methods are mentioned. This project differs from previous active noise control researches because a different adaptive algorithm is used. First of all, the noise has high frequencies. The filter is wideband and normalized. The secondary filter is online. The results of the comparison of these ANC researches and this thesis project showed that this thesis project performance is better.

Keywords: Active noise cancellation, ANC applications, online secondary path filter, FxLMS algorithm

DSP TABANLI AKTİF GÜRÜLTÜ KONTROLÜNÜN TASARIMI VE GERÇEKLEMESİ

ÖZ

Gürültü, bütün rahatsız edici sesler için kullanılan genel bir isimdir. Canlıların yanı sıra makineleri de kötü bir şekilde etkilemektedir. İnsan nüfusundaki sürekli artış ve teknolojik ihtiyaçların büyümesi gürültü ve gürültü çeşitliliğini arttırmaktadır. Bu yüzden gürültü kontrol metotları geliştirilmiştir. Çeşitli kulaklıklar yüksek frekanstaki gürültülerin kontrolü için kullanılırken, aktif gürültü kontrol metotları düşük frekanstaki gürültüler için kullanılır. Uygulama alanları bebek kuvözlerinden araba kabinlerine, uçak kabinlerinden hastane ortamlarına kadar çeşitlilik göstermektedir. Bu alanlarda gürültü düşük frekansa sahiptir. Aktif gürültü kontrolleri, yaygın olmamakla birlikte yüksek frekanslar için de kullanılabilmektedir. Bu projede, yüksek frekans bileşenlerine sahip olan dişçi delgisi sesleri için aktif gürültü kontrolü kullanılmıştır. Uygulama süresi boyunca MATLAB yazılımı kullanılmıştır. Öncelikli olarak algoritma simülasyon ortamında tasarlanmıştır. Uygun filtre parametreleri bulunmuştur. Sistem performansı doğrulandıktan sonra donanım kısımları dâhil edilmiştir. Donanım bölümünde dişçi delgi sesi mikrofonlar ile kaydedilir ve oluşturulan eksi ses ortama hoparlör yardımı ile aktarılır. Bu sayede teorik olarak ortamdaki gürültü belirli bir oranda yok edilir. Kurulan donanımsal yapı, bu teoriyi pratik hayata geçirmiştir. Gürültü elemesi açık alanda yüzde elli civarındadır. Elde edilen sonuçlar diğer ANC metotları ile karşılaştırılmış ve sonuçlar, grafikler ile birlikte raporlanmıştır. Bu metotların avantajları ve dezavantajları listelenmiştir. Projenin önceki araştırmalardan farklılığı geliştirilmiş bir algoritmanın kullanılmış olmasıdır. İlk olarak, gürültü yüksek frekanslara sahiptir. Kullanılan filtre geniş bir bantlıdır ve normalize edilmiştir. İkincil yol çevrimiçidir. Elde edilen sonuçlar proje performansının yapılan diğer araştırmalardan daha iyi olduğunu göstermiştir.

Anahtar kelimeler: Aktif gürültü kontrolü, ANC uygulamaları, çevrimiçi ikincil yol filtresi, FxLMS algoritması

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE
INTRODUCTION

Along with the enlarging human population, acoustic noise has become a big problem because of industries, electrical devices, gas and fluid transportations, vehicles, air planes and so on. Acoustic noise causes serious health problems such as hearing loss, hypertension, digestive disturbances, increased heart rate and sleeping problem (Goines & Hagler, 2007). Besides, increased mechanical fatigue and reduced component life are also the bad consequences of acoustic noises (Abdel, Ahmadi & Loo, 1990). In order to overcome these problems passive noise control has been implemented for a long time.

In passive noise control, there is no need for complex filter systems, signal processing cards, or any electrical components. Mufflers (Bi, Li, Chen & Liu, 2011), silencers (Passive flue gas silencer with mineral fibre absorber, n.d.), acoustic attenuators (Acoustic attenuators, n.d.), barriers, enclosures and headphones were developed as passive noise controller for instance. This control has been used for high frequency signals. Since these signals have shorter wavelengths it is easier to attenuate them. This is also valid for communication signals. However, when a signal has low frequency (consequently longer wavelength), it is hard to diminish that signal. The only way for passive control is using either heavy or expensive materials. Since it is not a desirable situation, active noise control was developed (Elliott & Nelson, 1993).

Active noise control was granted that name because it requires active electronic parts unlike passive noise control. In this control, the main idea is to produce an anti noise signal which will eliminate the existing acoustic noise. The most important thing is to obtain the exact amplitude with the opposite phase. Finally, this anti-noise cancels the noise signal and the summation becomes just a white noise, which always exists. Figure 1.1 shows the visual implementation of this process.

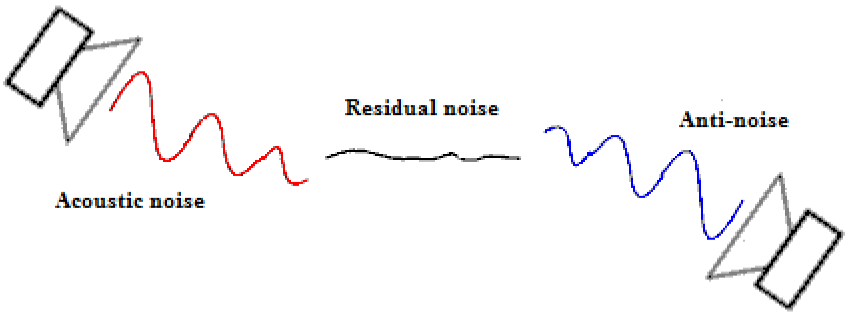

Figure 1.1 Active noise cancelling

ANC requires at least one microphone, one loudspeaker, and an anti-noise signal generator, which will eliminate the acoustic noise. These parts can be implemented either in a headphone or explicitly. Headphone design is the better one because it encircles the ears and also provides passive control. On the other hand, explicit ANC has also been studied for decades. These studies are generally for low frequency (less than 400 Hz) signals.

First feedback ANC system was developed by May & Olson (1953). In this system, there exists only one microphone, error microphone, and one speaker. This is the most basic form of feedback ANC systems. The desired noise was estimated in the filter and fed back to the filter input. More detailed information about feedback ANC was documented in the section 2.1.

1.1 Headset Applications

Gan & Kuo (2003), implemented active noise control into communication systems using headsets. Headsets do not only apply noise control but also provide place for communication hardware. They also the human communication is performed contrary to classical ANC headsets, which are used only for noise cancelling. Besides, active noise cancellation is implemented two times. One is at the receiver side, and the other is at transmitter side. Adaptive feedback ANC system at the transmitter side eliminates the noise, enhances the human sound and delivers this clear sound to the transmitter unit. Adaptive feedback ANC system at the receiver side, on the other hand, eliminates the environment noise that corrupts the receiving signal. These active noise control systems basically use least mean square (LMS) algorithms, which is the most popular algorithm to minimize (zero if possible) the total system error in ANC systems. These headsets are adaptive for variety of noises, robust, and practical.

Schumacher, Krüger, Jeub, Vary & Beaugeant (2011), proposed a new approach for broadband feedback ANC for headsets. At the earliest stage of the headset ANC system developments, non adaptive feedback ANC algorithms were used. This algorithm is suitable to attenuate low frequency ambient noises. On the other hand, adaptive ANC algorithms are effective for larger frequency range and especially periodic ambient noises. These two algorithms are combined under a new method called **novel hybrid feedback ANC**. This new combined method provides better results, higher overall noise attenuation, compared to pure non-adaptive systems and pure adaptive systems. Hybrid feedback ANC based headsets are low-cost, high-quality and practical compared to the other two methods. Furthermore, due to practical constraints, a mixed analog-digital realization was proposed (Schumacher, Krüger, Jeub, Vary & Beaugeant, 2011). This realization splits the overall system into two parts. In the digital part filtered-x LMS (improved LMS algorithm for real environment) is used while in the analog part a control filter is used to prevent the ambient noises from sudden jumps.

1.2 Open Environment Applications

Besides headset, ANC algorithms were used in a number of different applications. For example, ANC is used for the estimations of FMRI (Functional Magnetic Resonance Imaging) machine noises. These studies are very beneficial because the communication between the doctor and the patient in the machine is very important. Machine noise generally suppresses the person’s voice. In order to prevent this, Venkate Raghavendra Ramachandran implemented active noise control in fmri machines (Ramachandran, Panahi & Perez, 2008). LabVIEW program (as the software tool) and FPGA (as the hardware tool) were used with three different ANC system algorithms, namely normalized least mean square (NLMS), recursive least squares (RLS) and affine projection algorithm (APA). The overall system performances are according to the computation complexity and the signal (human voice) to noise ratios (SNR). High SNR values mean better performance. RLS algorithm is the best one for this system but it is the most complex one. It should be noted that there is a threshold between complexity and performance. Both cannot exist in the same algorithm.

Snoring sounds are also in the ANC application areas. These sounds are usually irritating for human ear. Therefore, Kuo and Chakravarthy developed a real-time ANC application to reduce the snoring sound (Kuo & Chakravarty, 2006). Lower system complexity and overall cost was achieved by using three different FxLMS algorithms. Using three different FxLMS algorithms, six snore ANC structures were realized. As the hardware, twin-size headboards were used. The snore signals showed that there was a wide area with a satisfactory filter performance with a low-cost.

When the disturbing noises are considered, periodic ones are prior. Because of this, Bodson, Jensen, & Douglas (2001), have focused on this topic in their studies. Feedback ANC algorithms were applied to various periodic noises with different unknown frequencies. Normally, pure sinusoidal noises with specific frequencies can easily be eliminated with adaptive filters. However, this process is a little bit difficult for unknown frequencies. For this process, first fundamentals of the noise are calculated and the control block parameters are adjusted according to these fundamentals and noise magnitudes. Then cascading these blocks with compensators, which are formed via the acoustic path and the noise frequencies, results in a good performance. In other words, adaptive algorithm is done without using LMS algorithm. It uses sinusoidal functions and fundamental frequencies of the noise.

ANC is also used in infant incubators. It is one of the most crucial practical topics in ANC area because infants are very vulnerable to environment sounds. The main purpose in this topic is to eliminate the noises in the care unit and infant incubators.

The infant incubator was realized with a baby doll, a speaker, an error microphone, and a computer system (Thanigai, Kuo & Yenduri, 2007). FxLMS algorithm was applied. However, the results included peaks at some points because of the nonlinearity of the environment noises. In order to reduce these peaks in output error, a different method with M-estimate function was used and the resulting error peaks were eliminated. Final ANC error is minimized but may not be enough for some cases. Therefore, audio integrated ANC is proposed for this situation. Error signal is combined with the audio. This audio was selected as heart sound to provide womb effect to the baby. To conclude, not only the baby growing was improved with the integrated heart sound but also the baby wasn't affected by the minimized (not zero) output error.

In case of the existence of reference microphone, the received data sets definitely affect each other. In order to overcome this problem, a new study was developed by Yifeng (1997). Different reference signals require different filter parameters and their acoustic noises will be different as it can be seen in Figure 1.2. Reference signals exist in the same environment and they highly affect each other. Decorrelated filter was developed so as to eliminate these effects. Finally, several different independent adaptive filter based subsystems were obtained. It is stated that this process can also be done with passive isolation, but this method would be more costly. This decorrelation filter, therefore, provides a low cost and high robust multi-reference ANC system. Figure 1.2 shows its clear block representation.

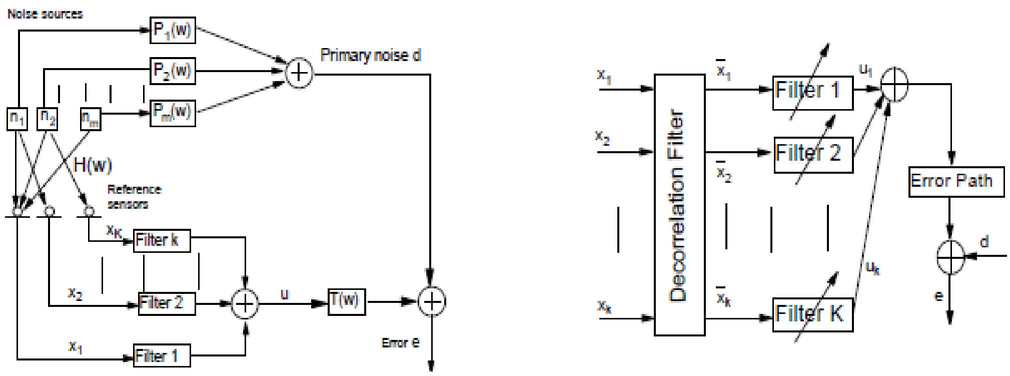

Figure 1.2 Decorrelation filter block representation (Yifeng, 1997)

There are also ANC-based researches regarding air and fluid noises. Air noise research is about fan noises (Homma, 2004). It begins with fan noise analyses. It is clear that it would have very different characteristics in space and in a duct. The second one is more important because the covering duct also provides passive noise control which enhances the performance of active noise control. Then a matching model for fun noise in a duct was implemented. This model underwent several active noise control methods, mainly LMS algorithms. While applying these algorithms, 8-segmented model was proposed. That is the duct is divided into 8 segments since the fan noise is non-linear at every point in the duct. This 8-segment model requires 8 different speakers and 8 different microphones as is can be seen in Figure 1.3. This process is a little bit costly, but its result is much better than the 1-segment noise cancellation. After modeling, the real system was designed and the results of passive and active noise controls (first separately then combined model) were observed and they are very satisfying. The hardware structure is illustrated in Figure 1.3.

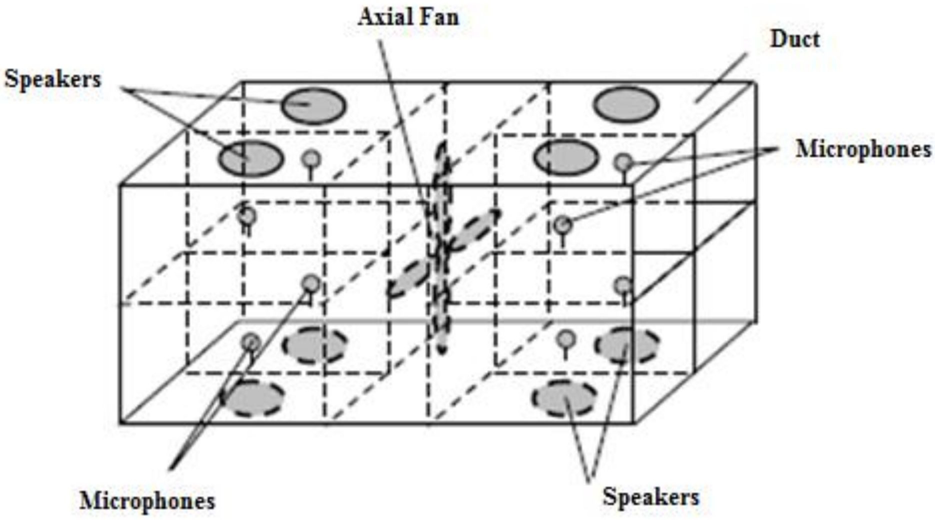

Figure 1.3 Air noise cancellation hardware model in a duct (Homma, 2004)

In the fluid borne study, active noise control is also very important (Wang, 2008). If the fluid noise is tried to eliminate just by passive noise control, then it would be very costly and not-portable. Firstly, the fluid borne system was modeled on the computer. Then the off-line and online noise cancellations were implemented as usual. In online cancellation part, there occurs a secondary path. This path represents the total effect of digital to analog converter, hardware components and speaker response. In the second path estimation part, they used four different estimation methods and compared their performances with respect to each other. As the adaptive filtering, FxLMS and 2-weights LMS notch filter were used. The second method also gave a reasonable result since the fluid noise is periodic, and periodic sounds are easy to be eliminated. However, the fluid borne frequency is important and it affects the system performance. Wang (2008), also investigated this frequency effect after completing the experiment setup with real pump.

ANC systems are generally applied to low-frequency noises. Another example to this is a study about automobile cabin noises (Wang, Gan & Kuo, 2005). Several decades ago, the implementation of ANC in an automobile would be unreasonable because it requires several hardware components (microphones, speakers, DSP boards and so on) and engines sounds are hard to be tracked due to its variety (non-linearity). Therefore, the FxLMS adaptive algorithm was implemented for a certain narrowband frequency range. First of all, the engine sound was estimated as if a periodic sound. Then its harmonics were extracted and 2-weight FxLMS notch filter was applied to the engine noise. This algorithm is called as active noise equalizer (ANE), different from ANC because the engine noise is not purely periodic and this notch filter doesn’t eliminate the noise entirely. It just reduces the annoying frequencies. It seems this study will be a big step for ANC systems in automobiles or other transportations.

Forest machines also generate irritating noises. Forsgren studied on these forest machines (Forsgren, 2011). FxLMS algorithm is applied after identification of primary and secondary paths. The overall result is reasonable when the system SNR values are considered. However, local performances may not be good, especially after 300Hz frequency.

The core of this thesis is an active noise control system for dental drill sounds. Dental drills produce broadband noises, which have all frequency values. These noises badly affect both the patience and the doctor in a long term. In order to reduce these bad effects, this project was performed. The main purpose of this design was to overcome high frequency acoustic noises (up to 23 kHz), without using any headphone. Frequency values higher than 23 kHz are out of interest because human hearing range is upper limited with 23 kHz. Experiments were done by using the combinations of the MATLAB blocks and codes, then DSP board. 2 microphones and 1 speaker were used for adaptive algorithm. 1 extra microphone was used for error measurement. This microphone reading is not fed back to adaptive filter. Kaymak, Atherton, Rotter, and Miller also dealt with dental drill sounds (Kaymak, Atherton, Rotter & Millar, 2007). They firstly determined the sound characteristics, and consecutively applied offline and online FxLMS ANC algorithm. They focused on narrowband noises. Therefore, their algorithm only works on particular frequency range. Finally, they used offline secondary path. These path parameters were fixed. Even though the interested noises are same, this thesis differs from (Kaymak, Atherton, Rotter & Millar, 2007) in many ways. First of all, broadband noise cancellation was applied. The frequency range is not limited with a single peak value. Second, FxLMS algorithm is normalized, which gives a better performance result. Finally, applied secondary path is online. Compared to offline secondary path, it is more robust to the environment changes such as device transportation, temperature change, and environment vibrations.

This thesis is organized as follows. In chapter two, theoretical background about ANC systems is documented. In this part, Feedback, feedforward, and hybrid ANC systems are examined. In feedforward part, there are adaptive filters listed, namely wiener filter, steepest descent method, LMS (Least Mean Squares), NLMS (Normalized LMS), sign LMS, FxLMS (Filtered-x LMS), linear prediction, kalman and particle filters. Definitions, graphical representations, equations, convergence boundaries, advantages and disadvantages are listed in this chapter.

In chapter three, drill sound characteristics are examined. Time domain and frequency domain graphs are obtained with different drill sounds. Modeling and simulations are performed using different LMS types and filter parameters. According to the results, the best algorithm and filter parameters are determined. MATLAB blocks are added, step size boundaries are determined, and algorithm equations are written with the filter parameters. Also online secondary path is modeled.

Chapter four gives detailed information about hardware setup. Optimum positioning of speakers and microphones are calculated with graphical results. Hardware microphone input system is added and its characteristic response is calculated. DSP card, and speaker features are listed in this chapter. Speaker frequency response and its bad effects are also mentioned here.

Chapter five contains the final FxLMS algorithms, their block representations and the algorithm results. These algorithms are separated in two parts, broadband and narrowband. Their detailed characteristics and bandpass filtering are documented in this chapter. Also acoustic feedback is introduced. Filter parameters are calculated and block representation is presented. For the result part, 40 different drill sounds were used. The average performances of offline NLMS, simple online NLMS, normalized FxLMS algorithms are calculated for these drill sounds. Additionally, narrowband normalized FxLMS and normalized FxLMS with acoustic feedback algorithms were developed and tested for specific drill sounds. The graphical and numeric results of all these algorithms were added in this chapter. Comparisons of these algorithms were also mentioned here.

As the conclusion part, chapter six is written. Brief information about the project and project report, project specialties, the comments on results, comparison of results and possible future works are documented in this chapter.

CHAPTER TWO
THEORETICAL BACKGROUND

In ANC systems, there are mainly three algorithm types, namely feedback (non-adaptive), feedforward (adaptive) and hybrid ANC. The type of the algorithm is specified according to the existence of reference signal. Reference signal represents the main noise. If the primary noise can be recorded and fed into the adaptive filter, then it is labeled as reference signal.

2.1 Feedback ANC

Feedback ANC is chosen in the applications where the reference signal cannot be reachable or coherently generated. In this situation, error signal is recorded and algorithm is applied to the system in order to reduce this error signal regardless of the reference signal. This system was firstly introduced by Olson & May (1953). Then it was applied using hearing protectors by Carme (1988), Wheeler & Smeatham (1992), and Veit (1988). Figure 2.1 shows a simple representation of feedback ANC algorithms (Raphael, n.d.).

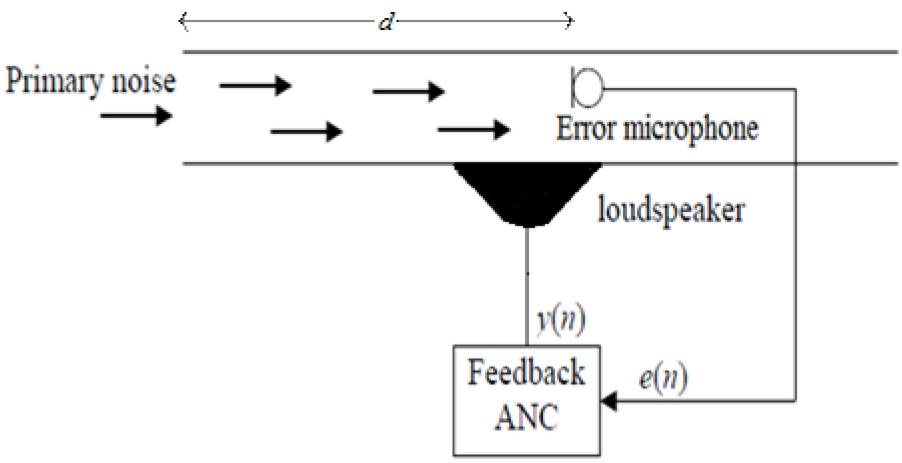

Figure 2.1 Simple hardware representation of the feedback ANC algorithm

In this simple representation, e(n) is the error signal and y(n) is the cancelling signal. The error microphone is placed in the region of the sound generator such that the loop delay is decreased. However, it has several problems as well. First of all, noise attenuation and the frequency range (not enough even though this range can be increased by decreasing “d” distance) are broadband limited. Besides, there is most likely instability in the system. Finally, because of the loop delay, only the periodic noises can be completely eliminated. Therefore, this representation can be improved as Figure 2.2.

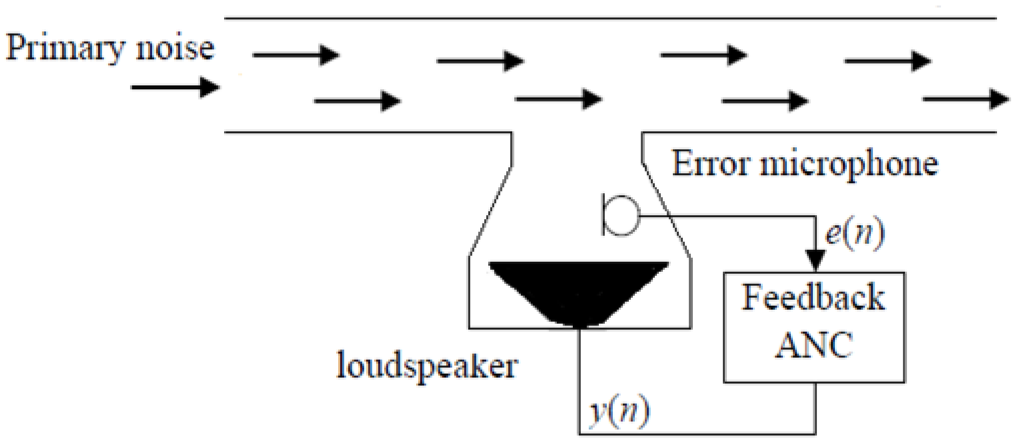

Figure 2.2 Improved feedback ANC algorithm hardware representation

In this representation the loudspeaker and the error microphone is stored in a small area connected to the duct. In this way, the closed area has good roll-off characteristics and the microphone is isolated from the nearby surface reflections.

2.1.1 General Analysis of Feedback ANC

In Figure 2.3, d(n) is the reference signal (primary noise), e(n) is the error signal, y(n) is the control signal, v(n) is the secondary noise, W(z) is the controller transfer function, S(z) is the transfer function of secondary path (represents the path between digital y(n) and the signal at desired noiseless area), and H(z) is the overall transfer function. In z-domain Equation (2.1) and Equation (2.2) can be written.

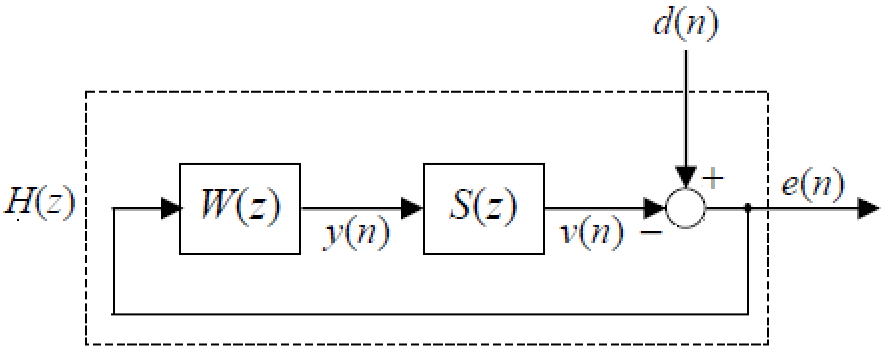

Figure 2.3 Block representation of a simple feedback ANC algorithm

$$H(z) = \frac{E(z)}{D(z)} \tag{2.1}$$

$$H(z) = \frac{1}{1+W(z)S(z)} \tag{2.2}$$

As $\mathrm{H(z)}$ goes to zero then error signal becomes negligible. This is achieved by increasing $|\mathrm{W(z)S(z)}|$. This can easily be done by making $\mathrm{S(z)} = 1$ and $\mathrm{W(z)} = \mathrm{M}$, where M is a large number so that $|\mathrm{H(z)}|$ goes to zero. However, in real life $\mathrm{S(z)}$ is not always constant and there is usually a phase shift resulting instability. If primary noise is a sinusoidal wavelength $(\cos(w_c nT))$, where w_c is cutoff angular frequency and T is the period. Then the z-domain representation of the sinusoidal wavelength, $\mathrm{D(z)}$ becomes:

$$D(z) = \frac{z^2-\cos(w_cT)z}{z^2-2\cos(w_cT)z+1} \tag{2.3}$$

Controller transfer function can be selected as:

$$W(z) = K\frac{z(z-a)}{z^2-2\cos(w_cT)z+1} \tag{2.4}$$

K is the controller gain, $(\mathrm{z}-\mathrm{a})$ is introduced to reduce the phase loss, and the decimator is to cancel out decimator of $\mathrm{D(z)}$. Then the overall system is stable.

To conclude, not requiring a reference microphone, stabilizing periodic noises and having a relatively simple algorithm are advantages of feedback ANC systems. On the contrary, having limited bandwidth, high possibility of instability, and not being robust are the disadvantages.

2.2 Feedforward ANC

2.2.1 Wiener Filter

In adaptive algorithms, the main purpose is to minimize the estimated error signal. Wiener filter is one of the first adaptive filters and it was proposed and published by Wiener. However, the discrete-time equivalent was derived Kolmogorov. Wiener filter then pioneered to many other algorithms and applications.

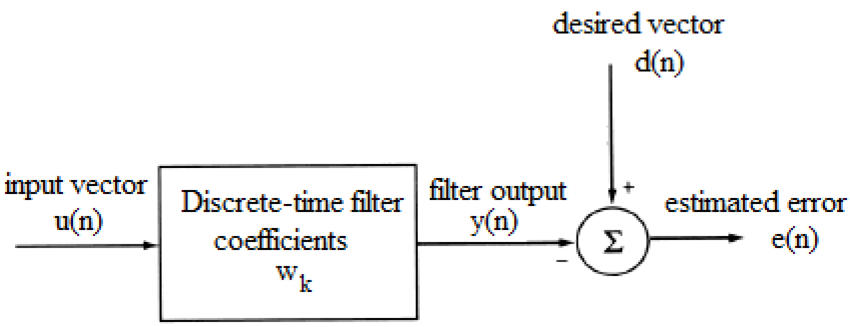

Figure 2.4 Wiener filter algorithm block representation

Figure 2.4 shows a clear indication of wiener filtering algorithm. In this situation, input and desired vectors are known and filter coefficients are found in order to minimize the estimated error. Hence, expectations of error based signals are considered. $E\{e(n)\}$, $E\{e^2(n)\}$, …, $E\{e^k(n)\}$ are possible expectation values for estimation error. For simplicity $\mathrm{e(n)}$ and $e^2(n)$ are considered. Because of the sign problem of $\mathrm{e(n)}$, generally $E\{e^2(n)\}$ is used for adaptive filter algorithms. This can be equalized to $\mathrm{E\{e(n).e*(n)\}}$ and then $\mathrm{E\{[d(n)-y(n)].[d(n)-y(n)]*\}}$, where '$*$' represents the conjugate value. Output can be calculated as (2.5).

$$y(n) = \boldsymbol{w}^T(n).\boldsymbol{u}(n) \tag{2.5}$$

$\boldsymbol{w}^T$ is the transpose of the weight matrix $\boldsymbol{w}$. If the mean square estimation is equalized to zero, then the Equation (2.6) can be found.

$$\frac{\partial E\{[d(n)-y(n)].[d(n)-y(n)]^*\}}{\partial w} = 0 \tag{2.6}$$

$\frac{\partial}{\partial w}$ is the partial differential according to the frequency filter coefficient w. Then (2.5) in (2.6) results in (2.7).

$$\frac{\partial}{\partial \mathrm{w}}[\mathrm{E}\{\mathrm{d(n).d(n)^*}\} - \mathrm{E}\{\mathrm{d(n)}[\mathbf{w}(\mathrm{n})^{\mathrm{T}}\mathbf{u}(\mathrm{n})]^*\} - \mathrm{E}\{\mathbf{w}(\mathrm{n})^{\mathrm{T}}\mathbf{u}(\mathrm{n})\mathrm{d(n)^*}\} + \mathrm{E}\{\mathbf{w}(\mathrm{n})^{\mathrm{T}}\mathbf{u}(\mathrm{n})[\mathbf{w}(\mathrm{n})^{\mathrm{T}}\mathbf{u}(\mathrm{n})]^*\}] = 0 \tag{2.7}$$

The first term becomes zero because $\mathrm{d(n)}$ is not related with angular frequency. Second and third terms are simplified as $-2\mathbf{p}$, where '$\mathbf{p}$' is the cross-correlation vector of the input and desired signals (Dmitriev, 2012), which is the estimated value of the multiplication of $\boldsymbol{u}(n)$ and $\boldsymbol{d}^*(n)$. Final term is simplified as $2\mathbf{Rw}$. $\mathbf{R}$ is the autocorrelation matrix of the input signal, which is the estimated value of the multiplication of $\boldsymbol{u}(n)$ and $\boldsymbol{u}^*(n)$. The final expression is minimized by these Wiener-Hopf equations and optimal weight coefficients, which minimizes the estimated error, are found to be as

$$\boldsymbol{w} = \boldsymbol{R}^{-1}.\boldsymbol{p} \tag{2.8}$$

2.2.2 Steepest Descent Algorithm

In wiener filter method, the inverse of autocorrelation matrix, R^{-1}, must be calculated. It is not an easy task when the matrix dimension is large. Instead of direct calculation of R^{-1}, the steepest descent algorithm can be used. The steepest descent algorithm which was first published by Debye (1909) is an iterative process that tries to minimize a cost function. This can be achieved by an iterative process.

$$J(w(n+1)) \leq J(w(n)) \tag{2.9}$$

Where $\mathrm{J}(.)$ is the cost function. There is variety of cost functions. It simply indicates the cost for producing output units. (2.9) implies that weights should be updated in the direction of decreasing cost function. In other words, in the opposite direction of gradient vector ∇, which gives the maximum increase direction in the cost function.

$$\mathrm{w}(\mathrm{n}+1) = \mathrm{w}(\mathrm{n}) + \frac{1}{2}\mu(-\nabla_{\mathrm{w}}\mathrm{J}(\mathrm{n})) \tag{2.10}$$

$J(n)$ is represented as mean square error, $\mathrm{E}\{[\mathrm{e}(\mathrm{n})].[\mathrm{e}(\mathrm{n})]^*\}$. Similar to (2.7), after few more equations (Adaptive signal processing, n.d.) the final equation is to be found as

$$w(n+1) = w(n) + \mu(p - Rw(n)), n = 0,1,2,\ldots \tag{2.11}$$

This weight iteration is done until the weight is no longer changes. μ is the step size and specifies the convergence, robustness, convergence speed and steady state error. Selecting of the step size μ is crucial because for convergence the maximum limit for step size is $\frac{2}{\lambda_{max}}$ and the time constant is calculated as (2.12) (Barner, 2009).

$$\tau_m = \frac{-1}{ln(1-\mu\lambda_m)} \tag{2.12}$$

λ_m are the eigenvalues of the input vector and λ_{max} is the biggest one of these.

2.2.3 LMS (Least Mean Squares)

LMS algorithm is similar to stochastic gradient method and it was first introduced by Dernard Widrow and his Ph.D. student Ted Hoff (Stanford University). In contrast to steepest gradient method indicated in Equation (2.11), LMS algorithm does not require the autocorrelation matrix R and cross correlation vector p. Moreover, it can be used in non-stationary processes (time-varying signals) because it uses instantaneous samples instead of ensemble averages. It has feedback connection (low pass filter) and its computational complexity is low. The process is easier, but the algorithm both has slow rate of convergence and is vulnerable to gradient noise.

As in the steepest descent method, update rule is Equation (2.10), where J(n) is the cost function. The new parameters are based on instantaneous values. The gradient of the cost function is

$$\nabla_{\mathrm{w}} \hat{\mathbf{J}}(\mathrm{n}) = -2\hat{\mathbf{p}} + 2\hat{\mathbf{R}}\mathbf{w}(\mathrm{n}) \tag{2.13}$$

Cross correlation vector is

$$\hat{\boldsymbol{p}} = \boldsymbol{u}(n)d^{*}(n) \tag{2.14}$$

Autocorrelation matrix is

$$\hat{\boldsymbol{R}} = \boldsymbol{u}(n)\boldsymbol{u}^{H}(n) \tag{2.15}$$

Where $u^{H}(n)$ is the hermitian (both transpose and conjugate) of $\mathrm{u}(\mathrm{n})$. Combining these equation gives the final weight update rule.

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \mu\boldsymbol{u}(n)e^{*}(n) \tag{2.16}$$

The Equation (2.17) represents the error value.

$$e(n) = d(n) - \hat{\boldsymbol{w}}^H(n)\boldsymbol{u}(n) \tag{2.17}$$

Selection of step size μ is crucial. Small step size results in slower process, but the filter becomes more resistant to gradient noise. Upper bounds for step size are determined by the Table 2.1 (Schmidt, n.d.).

Table 2.1 LMS algorithm step size upper bounds for system convergence

Perspective of LMS Filter	**Upper bound for step-size**	**Parameter descriptions**
Zero order solutions of LMS filters	$\frac{2}{\lambda_{max}}$	λ_{max} is the maximum eigenvalue of input correlation matrix
Long LMS filters	$\frac{2}{MS_{max}}$	M is filter length, and S_{max} is maximum of input power spectral density
H^{∞} optimality of LMS filters	$\min\limits_{1 \le n \le N} \frac{1}{\|\|u(n)\|\|^2}$	N is the iteration number for training the filter, and u(n) is the input signal

2.2.4 NLMS (Normalized Least Mean Squares)

In LMS algorithm step size μ is chosen constant. Since the process is stochastic, the amplitude of the input samples affects the performance (Klippel, n.d.). Structure and computation of NLMS algorithm is same as LMS algorithm except in NLMS, step size, $\mu(n)$, is time dependent. In order to reduce the effect of input samples, step size is divided by the square of that input sample. In other words, step size is chosen as

$$\mu(n) = \frac{K}{||\boldsymbol{u}(n)||^2} \tag{2.18}$$

K is optional gain. Because of the square expression, convergence rate is higher than LMS convergence rate. However, because of the possibility of input sample's being very small, decimator of step size may have an extra term. The final update rule becomes (2.19).

$$w_i(n+1) = w_i(n) + \frac{K}{\varepsilon(n) + ||u_i(n)||^2} u_i(n) e^*(n) \qquad (2.19)$$

$\varepsilon(n)$ is the regulation factor used to prevent the decimator from being zero, and it is chosen as small values.

2.2.5 Sign-error LMS, Sign-data LMS and Sign-sign LMS

Sometimes implementation of LMS algorithm into DSP boards, FPGA cards, and application-specific integrated circuits (ASICs) may be very hard. In order to reduce the complexity, sign operation is involved in the weight update equation. Existence of two parameters (u(n) and e(n)) results in the following three different types of LMS (Least Mean Squares algorithms, n.d.).

$$\text{Sign(u)} = \begin{cases} -1 & if \;\; u < 0 \\ 0 & if \;\; u = 0 \\ 1 & if \;\; u > 0 \end{cases} \qquad (2.20)$$

The following update equations are used in these LMS algorithms.

Sign-error LMS (Pilot LMS):

$$w_i(n+1) = w_i(n) + \mu u_i(n) sign(e^*(n)) \qquad (2.21)$$

Sign-data LMS (signed regressor):

$$w_i(n+1) = w_i(n) + \mu sign(u_i(n)) e^*(n) \qquad (2.22)$$

Sign-sign LMS (zero-forcing LMS):

$$w_i(n+1) = w_i(n) + \mu sign(u_i(n)) sign(e^*(n)) \tag{2.23}$$

Sign LMS algorithms may have less implementation complexity but they have lower convergence rate (even may not converge) and greater steady-state error. Besides, in analog realizations, these algorithms result in different dc offset implications.

2.2.6 FxLMS (Filtered-x LMS)

In ANC systems, in order to eliminate the noise, a secondary anti noise generator is used. Because of the existence of secondary path effect, standard LMS algorithm does not work on real ANC cases. Figure 2.5 shows a clear indication of LMS and FxLMS algorithms.

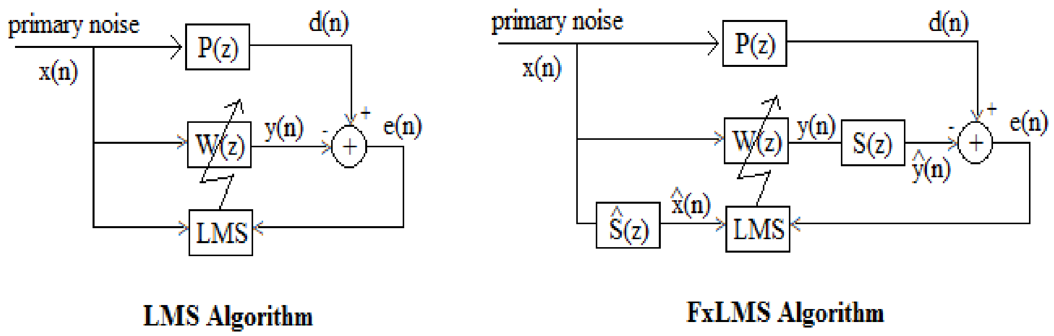

Figure 2.5 Block diagram representation of both LMS and FxLMS algorithms

Secondary path simply represents the hardware components, such as digital to analog, and analog to digital converters, power amplifiers, and preamplifiers, reconstruction and antialiasing filters. Figure 2.6 shows the block diagram of inside of the secondary path.

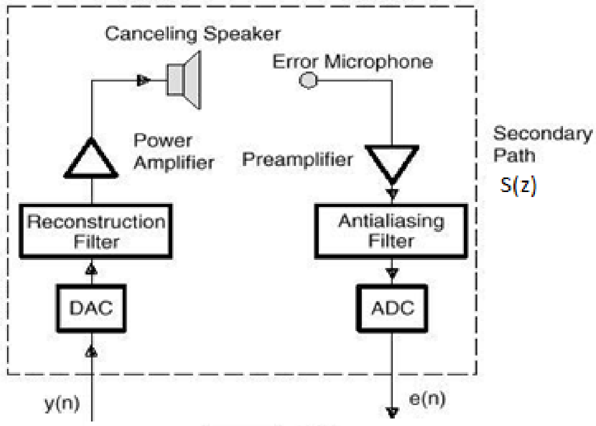

Figure 2.6 Block representation of the secondary path (Raphael, n.d.)

Existence of secondary path transfer function results in amplitude and phase change (delay) of output signal y(n). Therefore, error signal cannot be minimized. In order to compensate this delay and amplitude change, primary noise should also be modified. This modification can be done by inverse of the secondary transfer function S(z). However, S(z) is not a minimum phase system in general. Instead of inverse of S(z), its estimated equivalent is used in order to filter the primary noise. Because of this filtering process this algorithm is called as FxLMS algorithm. Calculation analysis is same as the LMS analysis. The only difference is using filtered $x(n)$, $\hat{x}(n)$. Final FxLMS equation becomes as follows:

$$e(n) = d(n) - \hat{y}(n) \tag{2.24}$$

$$\hat{\mathrm{y}}(\mathrm{n}) = \mathrm{s}(\mathrm{n}) * \mathrm{y}(\mathrm{n}) \tag{2.25}$$

$$\mathrm{y}(\mathrm{n}) = \boldsymbol{w}^T(n)\boldsymbol{x}(n) \tag{2.26}$$

$$w_k(n+1) = w_k(n) - \mu e(n)\hat{x}(n) \tag{2.27}$$

P(z) is primary transfer function, S(z) is secondary transfer function, $\hat{S}(z)$ is estimated S(z), y(n) is output signal, x(n) is primary input signal, e(n) is error signal, $\hat{y}(n)$ is estimated output signal, $\hat{x}(n)$ is filtered input signal, d(n) is desired signal, and w(n) is weight vector.

After many years, FxLMS algorithm has been developed by various new methods. For example, Sun's algorithm, which restricts the primary noise with some threshold (Tahir, Wataru & Akinori, 2011) is one of them. By this threshold method, impulsive noises are neglected. Other than sun's algorithm, normalized FxLMS, modified normalized FxLMS (Tahir, Wataru & Akinori, 2011), leaky FxLMS, sign-sign FxLMS, variable step size FxLMS, and decorrelated FxLMS (Vicente, Elliot & Masgrau, n.d.) are some of the methods enhanced in FxLMS algorithm.

2.2.7 LS, Linear Prediction, Kalman, and Particle Filtering

These algorithms are some of the remaining methods, which can be used for error minimization. First of all, in LS method, the inputs exist and the outputs are observed. Then a modeling is performed so that the mean square of the difference between the model output and the real output is minimized. Through all of these processes, our approach is deterministic (time averages are considered), and weight estimates are unbiased as long as error measurements have zero mean.

Linear prediction, on the other hand, tries to estimate the upcoming input value. For example, existing input is $\mathrm{u(n)}$ and the system is focused on $\mathrm{u(n+1)}$ so that the adaptive filter can eliminate $\mathrm{u(n+1)}$ value when it comes to the system. Linear prediction algorithm has two main types as feedforward and feedback linear prediction. Besides, linear prediction-based filters have a variable M, filter order, and the filter performance strongly depends on this filter order.

Kalman filter has a main difference which is being applicable to both stationary and non-stationary processes. Algorithm is done recursively and it uses previous estimates as well as the new input data. Kalman filters have lots of parameters (innovation parameters related with new input datas, kalman gain…ect.) and calculations (Maybeck, 1990).

Particle filtering is the most recent adaptive filter algorithm. It has lots of sub-methods (Arulampalam, Maskell, Gordon & Clapp, 2002) and the number of these methods has been continuously increasing. Main idea for particle filter is to divide the current input data into particles (M, filter order). Then same process is applied for the next input data, and the patterns of each particle are observed and estimated for the next input data. Note that, the input data are noisy; therefore, selection of filter order is crucial. Larger filter order results in better results, but it also increases the computation time. The main purpose is to find an optimum order considering these problems.

2.3 Hybrid ANC

Hybrid ANC is actually the combination of feedback and feedforward ANCs. In this ANC system, the input is not available; however, the designer wants to apply an adaptive algorithm just based on the error. Therefore, it is also called as adaptive feedback ANC (AFANC). The main purpose is still to minimize the mean square. The results may be both non-satisfying (because of the lack of the input itself) and satisfying (if the input variation is slow enough to track just through the error). There is a balance between these situations. A well-designed AFANC can easily perform up to 50dB noise reduction in a large room. This reduction is reasonable when compared with feedback ANC (not so effective) applications. Figure 2.7 shows a clearer indication of these processes (Kuo & Morgan, 2008).

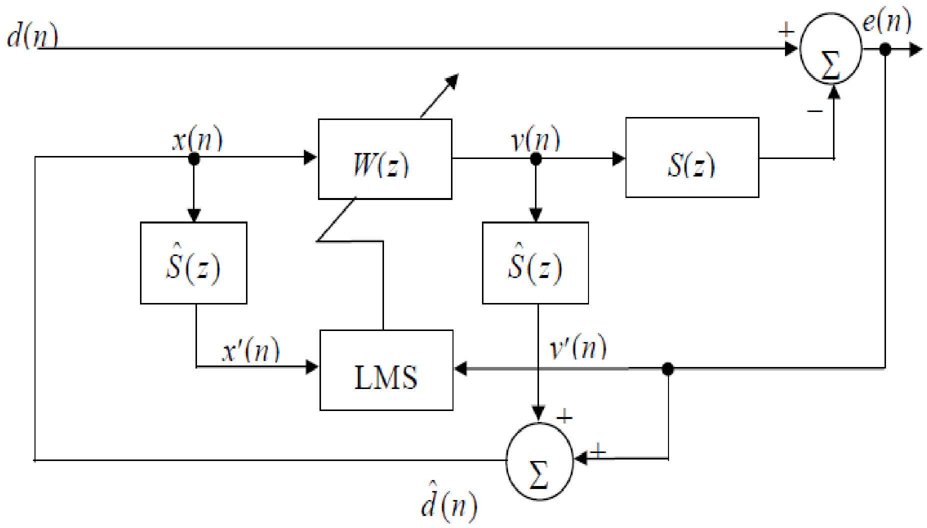

Figure 2.7 Block diagram of an adaptive feedback ANC algorithm

This configuration resembles the FxLMS configuration. The main difference is that there is no direct connection from the desired signal to the system. Therefore, input values for the adaptive filter is calculated via error signal e(n). Other than that, plant transfer functions, LMS algorithm (just an example), and weight parameters still exist. The overall performance highly depends on the input frequency. To conclude, hybrid ANCs are more vulnerable to high input frequencies than adaptive ANCs (Kuo & Morgan, 2008). Since the dentist drill sounds contain high frequency components, this method didn't preferred. The hybrid ANC also has the disadvantages of the normal feedback ANC algorithm.

In this chapter, feedback, feedforward and hybrid ANC algorithms were examined. The types of feedforward algorithms were defined. Definitions, equations, derivations, performances, advantages, and disadvantages were explained in a detailed way.

CHAPTER THREE
DRILL CHARACTERISTICS, MODELING AND SIMULATION

3.1 Drill Characteristics

As it is indicated earlier, dentist drills result in high frequency noises. Therefore, standard adaptive filter algorithms cannot compensate the noise changes. In order to overcome this problem, the drill sound characteristics must be deeply investigated. At first, 40 different drill sounds were recorded. 20 of them were from free-running drill, and the other 20 were from different drilling processes. To conclude, all samples vary from each other regarding both time and frequency domains. Then frequency domain magnitudes of the samples were examined because the signal power is easily calculated in the frequency domain (power spectral density). By this way, one can also know which frequency to focus for the best performance. Power spectral density (PSD) is representation of the total power according to a given frequency value. It is calculated by the Equation (3.1):

$$PSD = \frac{1}{N} \left| \sum_{n=0}^{N-1} x[n] * e^{-jwn} \right|^2 \tag{3.1}$$

Where N is the window size, x[n] is the signal magnitude at time instance n, and w is the frequency value at which the frequency power is calculated.

Two different sound examples and their power spectral densities can be seen in Figure 3.1. Drill sound 1 is from the free running drill sounds, whereas drill sound 2 is from the drill sounds recorded while drilling.

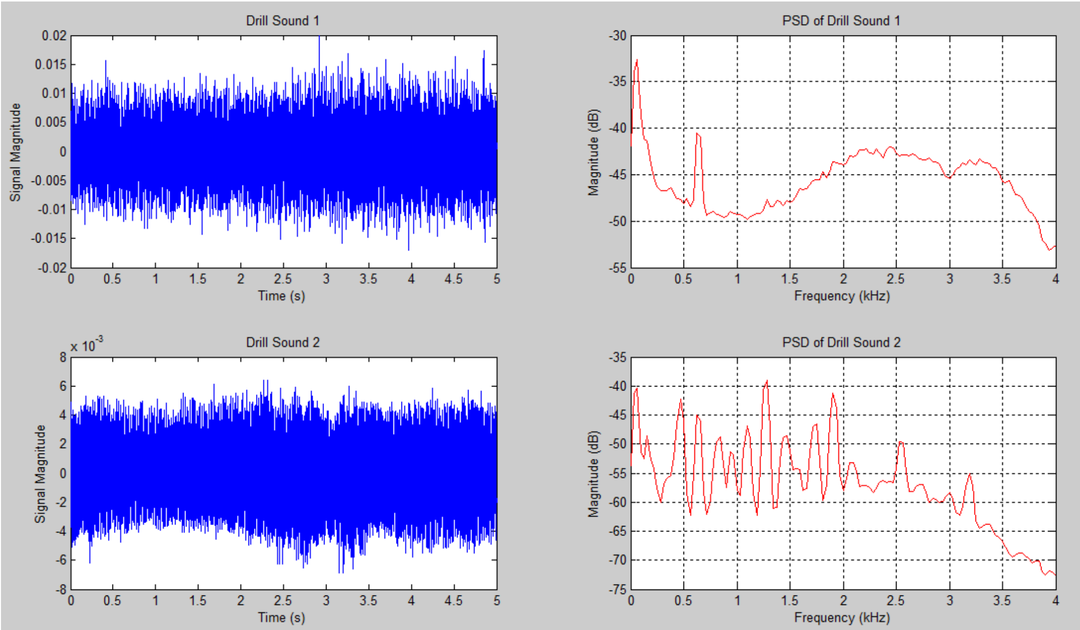

Figure 3.1 Two different drill sounds in time domain and their power spectral densities

It is seen that these two sounds have different time domain characteristics. First one is more like a periodic sound. Hence, the spectrum domain has a peak around frequency 600Hz. In this situation, main purpose is to eliminate this peak. In the second example, the sound sample looks like a non-periodic sound. This case is hard to implement the ANC algorithm on because it has both a non-linearity and high frequency. As it is seen in the spectrum analysis, power peaks are scattered around all frequencies and all of these ranges must be minimized. In other words, ANC algorithm should be applied to every frequency range, namely broadband ANC, which is one of the specialties of this project.

Note that humans cannot clearly comprehend the frequencies greater than 4 kHz. Therefore, the sampling frequency for drill sounds is selected as 8000 Hz since the maximum frequency that we are interested in for the drill is 4 kHz respectively. Otherwise, an overlapping in the spectrum domain might be the result. In the spectrum domains, both sounds have decreasing power magnitudes after around 3.5 kHz. This is the good part because higher frequencies won't be considered.

The frequency phases are also important in real time ANC applications. Real time situations are the focus. Therefore, LMS algorithm fails and it is developed as

FxLMS with primary and secondary paths, $P(z)$ and $S(z)$. These paths compensate the magnitude and phase changes occurred in the LMS algorithm resulting in online real-time adaptive ANC algorithm.

3.2 Modeling and Simulation

These parts were completely performed as simulation, not as real time application. Modeling is performed via MATLAB program. As the first step, DSP toolbox of the simulink was used. There is a prepared LMS block performing NLMS algorithm, which is the normalized version of the most common ANC adaptive algorithm, LMS. There are several options in the block changing the algorithm parameters, namely filter length, step size and LMS types. Different LMS algorithms, such as LMS, NLMs, sign-error LMS, sign-data LMS, and sign-sign LMS, are applied to one of the recorded drill sounds with filter length 256, and step size 0.001. Figure 3.2 was obtained as the system error result.

System error represents the difference between the desired noise $d(n)$ and the system output $y(n)$. As it is seen in four of the LMS types, system error is minimized after a particular sample number. NLMS algorithm has the best convergence time and better steady state error. Hence, NLMS algorithm was used. If the NLMS algorithm result is considered, a drill sound with around 0.2 time domain amplitude results in 0.01 time domain error amplitude, which means around 25-30 dB signal to noise ratio (SNR) was obtained for this sound sample. This result is very satisfying compared to previous ANC studies.

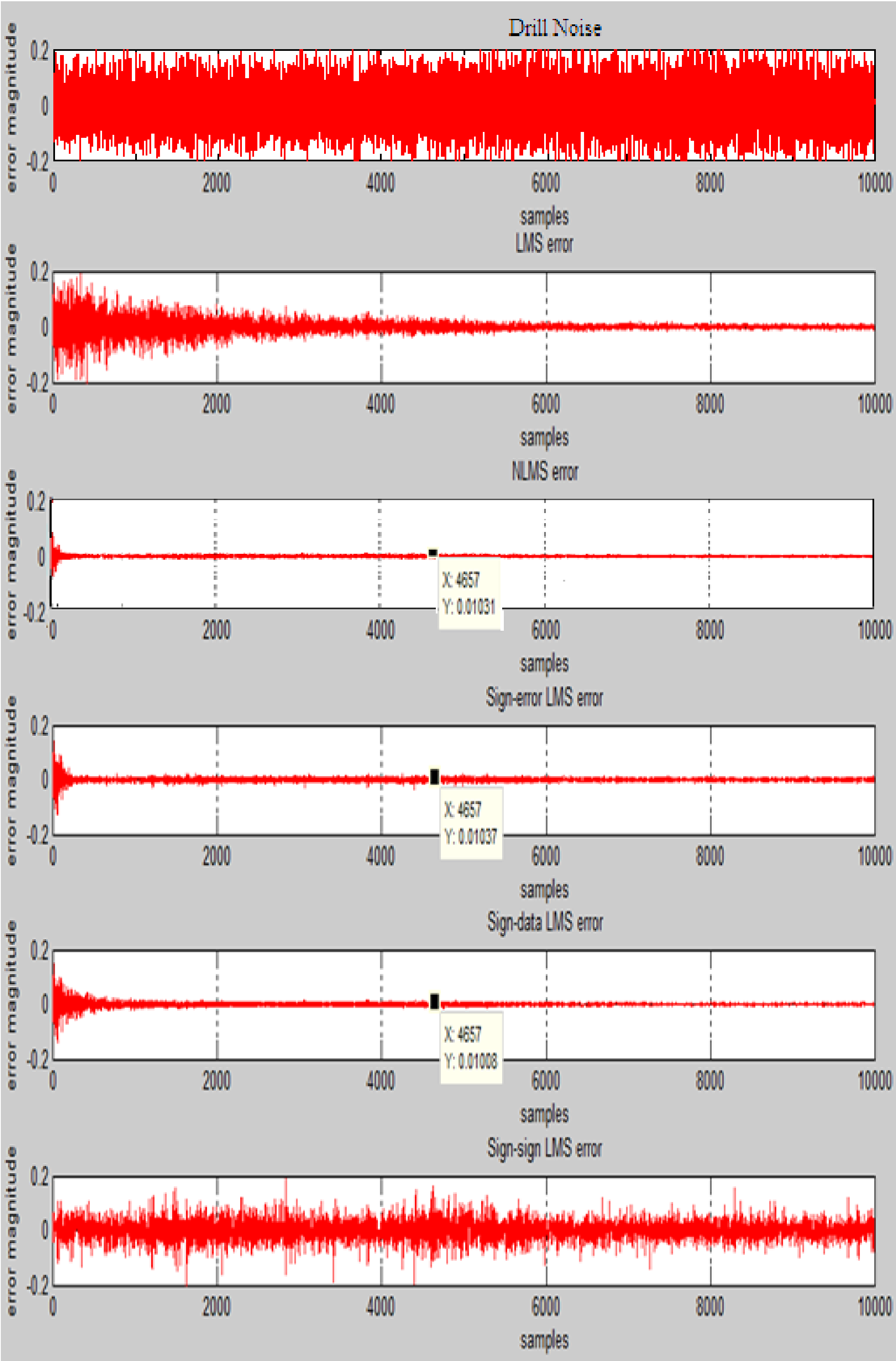

Figure 3.2 Error signal convergence in LMS, NLMS, sign-error LMS, sign-data LMS, and sign-sign LMS algotihms

As the second part, filter length and step size were changed in the LMS block. It can easily be guessed that these parameters highly affect the error result to the some point. Figure 3.3 clearly shows the convergence of three of the weights and the filter output.

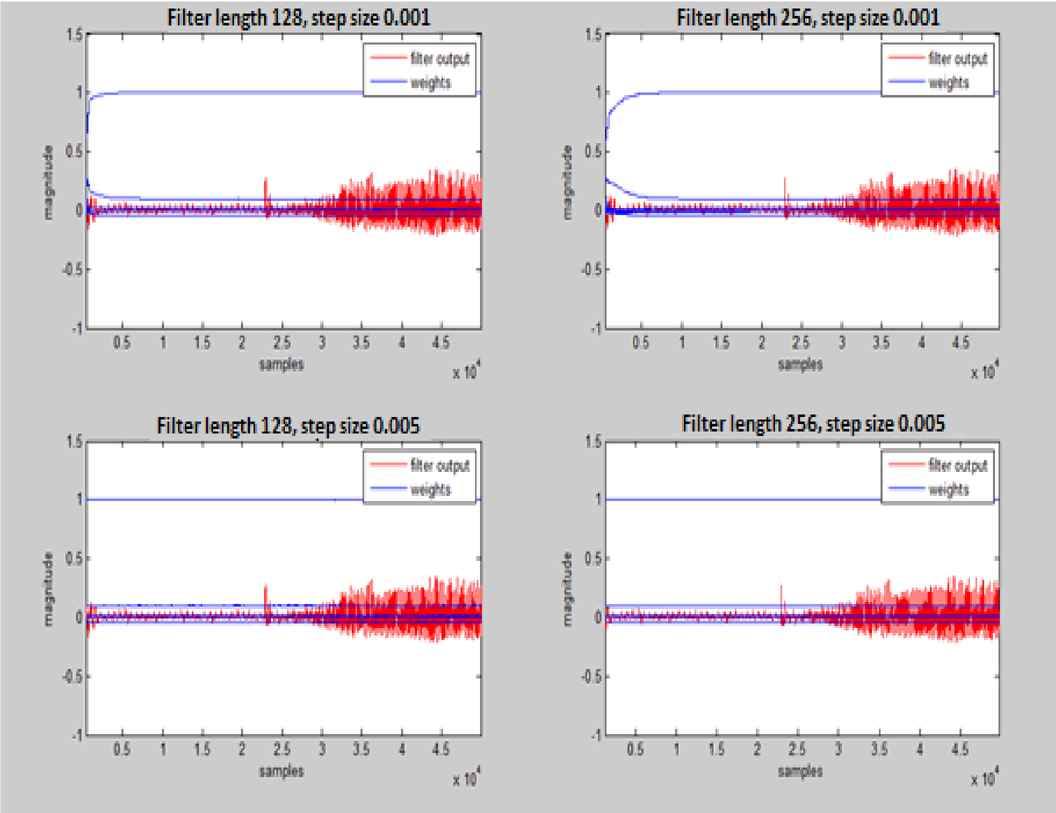

Figure 3.3 Convergence of 3 weights and the filter output according to filter length and step size

For better understanding, error convergence was also plotted according to these parameters, indicated in Figure 3.4. The indicated error magnitudes are respectively -1.365e-005, 4.789e-040, 2.028e-005, and 1.871e-038. Increasing filter length results in less steady state error. However, it requires more time to converge. Small step size, on the other hand relatively reduces the convergence time, but large step size may cause instability, which makes the algorithm useless in real time applications.

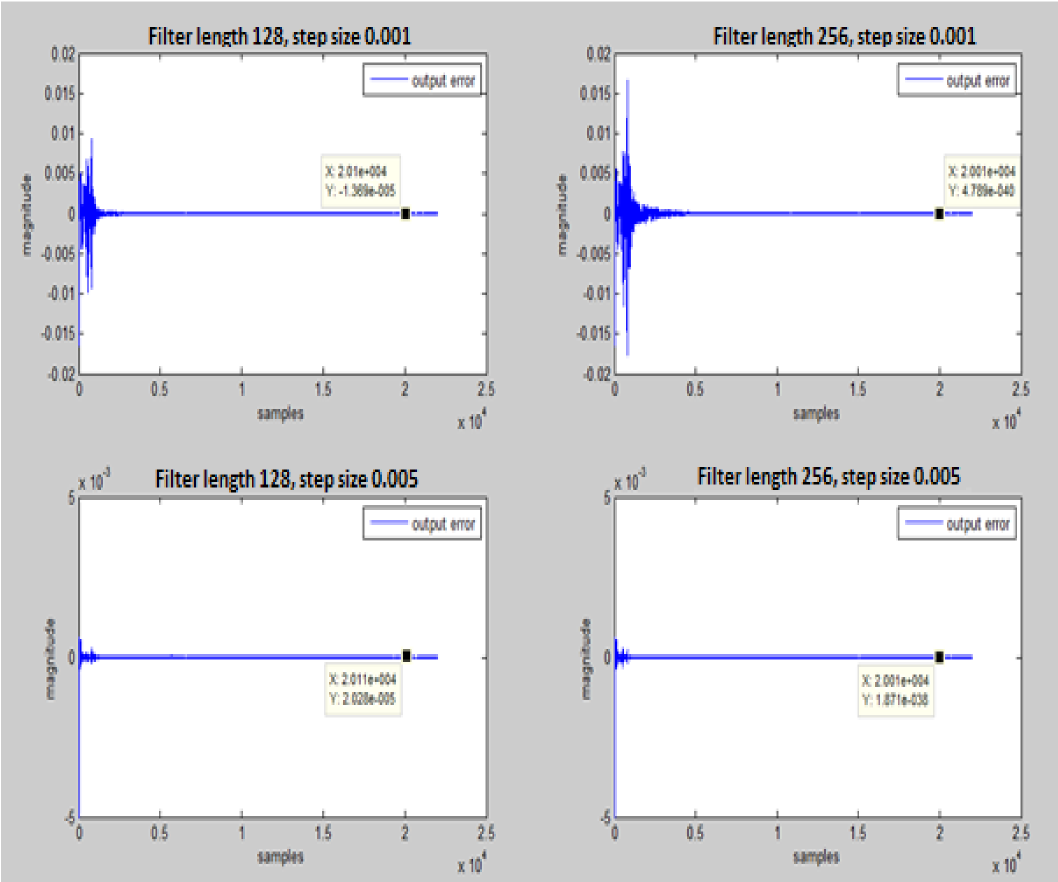

Figure 3.4 Filter error signal according to filter length and step size

For our online system, filter length was determined to be 256 as a result of extensive simulations. After 256 orders, filter lags for some input values because the DSP clock speed cannot catch with this filter order in a limited time. After determining this filter order, step size was selected according to the Equation (3.2) (Schmidt, n.d.).

$$\mu < \frac{2}{MS_{max}} \tag{3.2}$$

M is the filter length, 256 for this project, and S_{max} is the maximum value of the input PSD, 5 for this project. Resulting in Equation (3.3).

$$\mu < 1.56 * 10^{-3} \tag{3.3}$$

As a result of extensive simulations, the best combination of these parameters was found as 0.001 for step size and 256 for the filter length. After these configurations, MATLAB LMS block was removed and the whole procedure was done by MATLAB functions, and the required blocks. In the real time operations, normalized FxLMS algorithm is required. The error signal is supposed to be fed back into the filter and the MATLAB LMS block cannot be externally modified to obtain FxLMS block. Figure 3.5 shows the NLMS configuration block diagram designed in simulink.

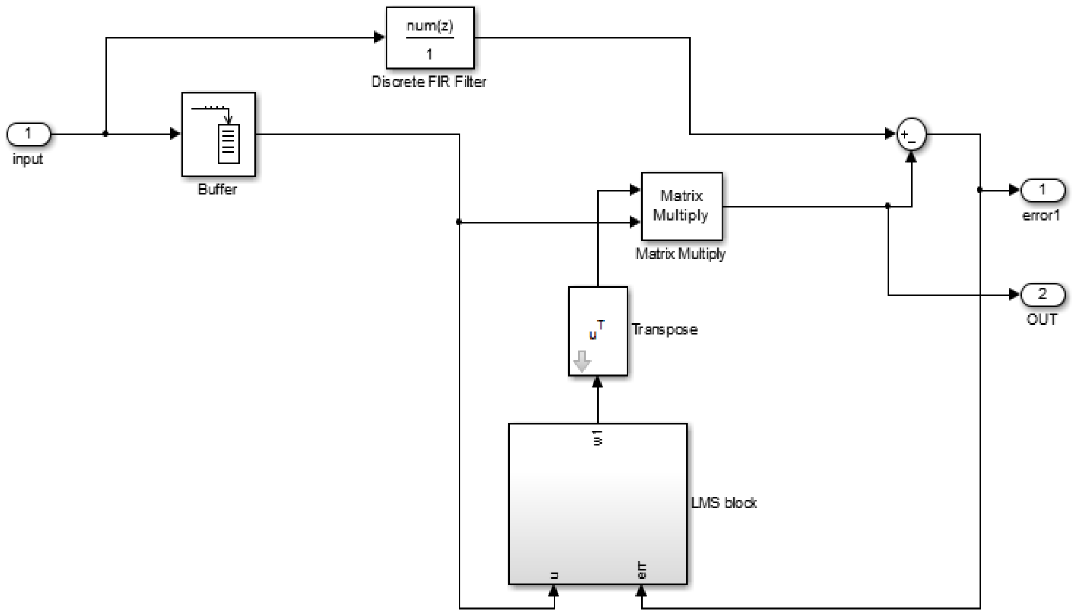

Figure 3.5 NLMS block diagram designed in MATLAB simulink

'Discrete FIR Filter' is a z-domain filter example for the primary path, which is a representation of the sound delay and loss between input and desired microphones. Buffer with 256 sample capacity was used in order to store the input samples and transport to the LMS subsystem block by block. LMS block adapts the filter weights according to NLMS algorithm so that the output error is minimized. Inside of the LMS block is as Figure 3.6.

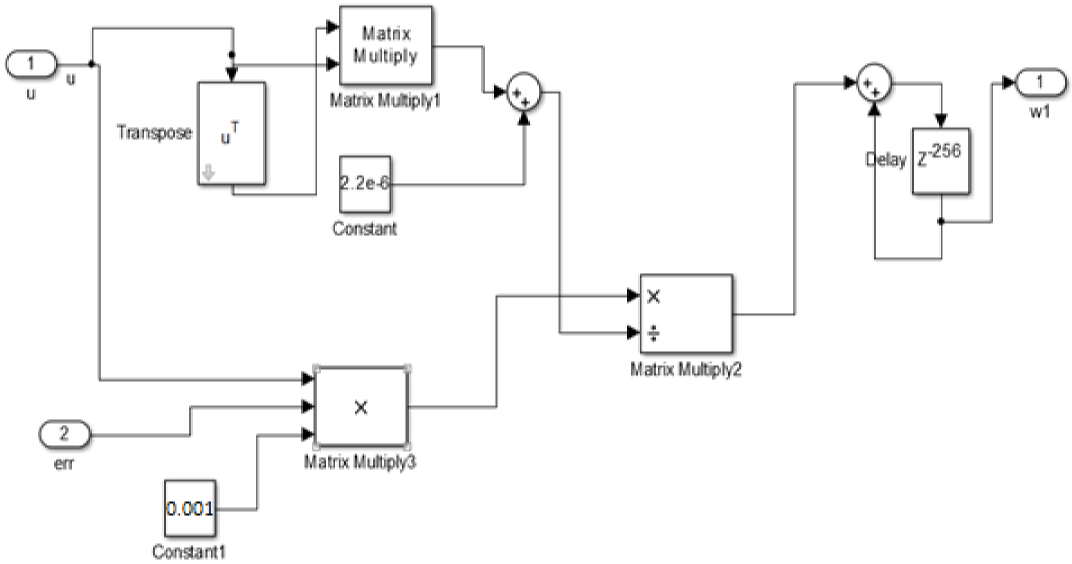

Figure 3.6 Internal structure of the LMS block

First of all constant 2.2e-6 is regulation factor, and it was used in order for the decimator not to be zero. Otherwise, the system output would go to infinity. This 2.2 value was recommended by the MATLAB. Constant1, on the other hand, is the step size. Delay was used to compensate the filter length. To conclude, the whole NLMS system is as Equation (3.4).

$$w_i(n+1) = w_i(n) + \frac{0.001}{2.2*10^{-6}+||u_i(n)||^2} u_i(n)e^*(n) \tag{3.4}$$

This project design in simulink has slightly better performance than the MATLAB LMS block. First of all, the error results are more satisfying though it is a small difference. Then, changing of the primary path in MATLAB LMS block may result in instability because it cannot adapt an external code block so easily. Data types, sampling frequencies, and frame sizes may differ in these code blocks, and some internal variables cannot be modified in MATLAB LMS block. This block is not flexible. In the project design, changing primary path can be balanced via the other code blocks. Primary path is usually selected as 1-order FIR filter. Let’s say the it is $P(z) = \frac{-7z}{10z-5}$. Then Figure 3.7 shows the error results. At 764th sample, error value is 0.007737 in MATLAB LMS block, while it is 0.006908 in project LMS block.

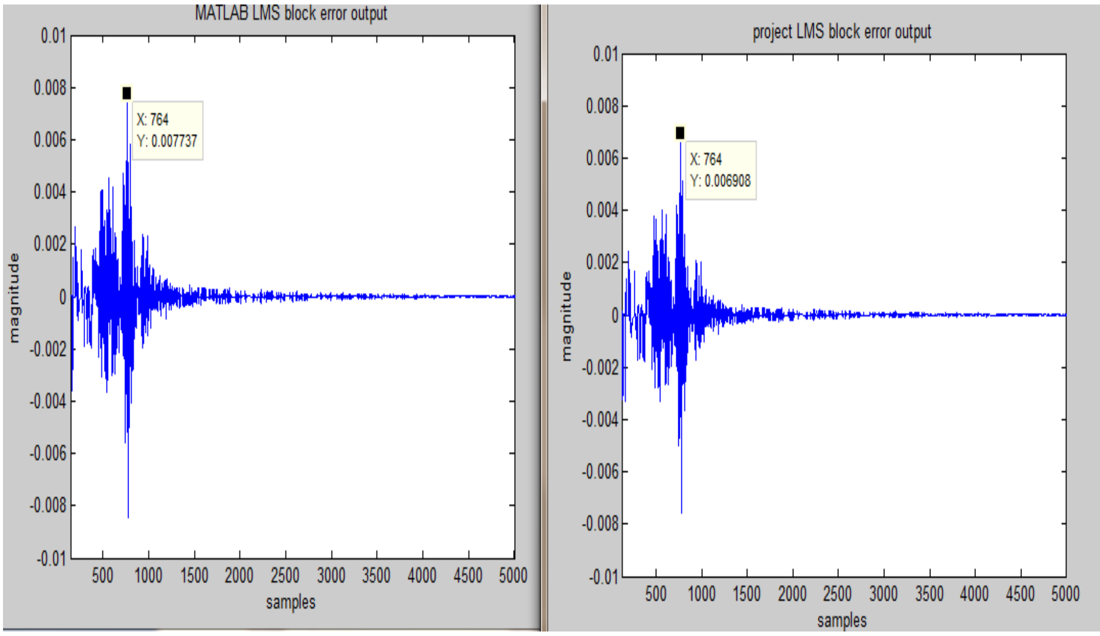

Figure 3.7 Error result comparison between MATLAB LMS block and project LMS block

These parts were related to the offline NLMS system. However, online NLMS systems are required to be implemented, in which an unknown drill sound is received and it is eliminated via NLMS algorithm. In order to do this, same simulation is preserved except ‘from workspace’ block. It was replaced by ‘from audio device’. Thanks to this block, the microphone connected to the computer receives the drill sound continuously and delivers it to the NLMS filter. ‘Basic online NLMS’ name was assigned for this algorithm. No speakers and secondary paths are involved. Figure 3.8 represents the error output for a random drill sound input. Convergence was achieved after 10000th sample, which corresponds to 0.1 second.

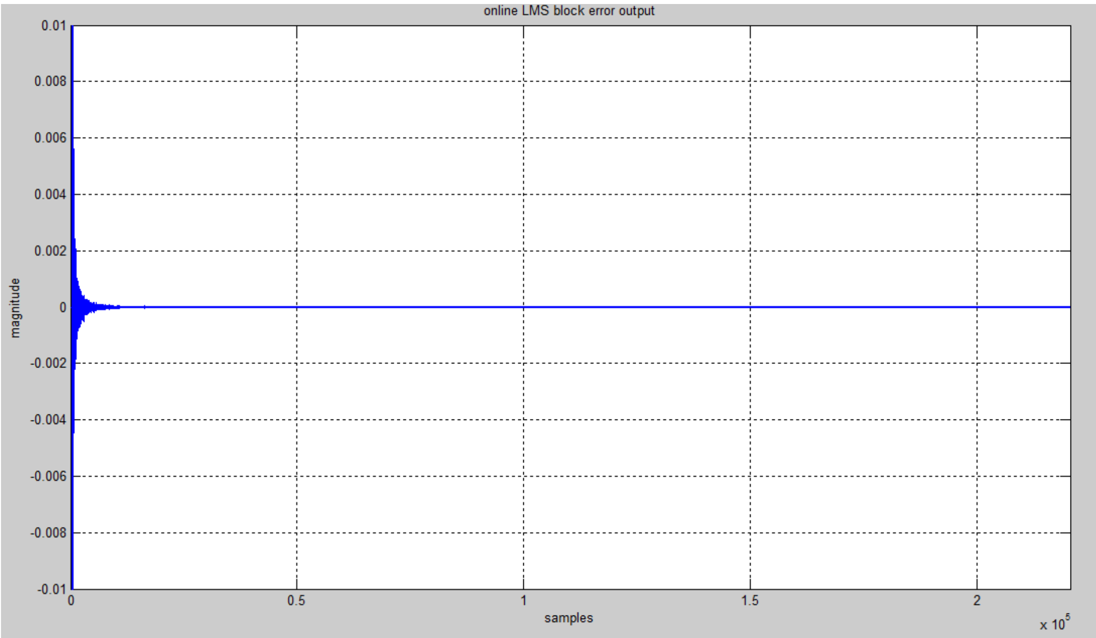

Figure 3.8 Error output of a simple online NLMS algorithm

Even though this basic online model is satisfying, real time processes require more complex models. In order to understand this complexity and to overcome it, FxLMS algorithm was introduced. The crucial part is the effects of the secondary path and finding the path parameters. The reason why secondary path exists is the existence of digital and analog converters, and speaker characteristics. Two methods can be applied in order to find the path, namely offline and online. In this project, Eriksson Model (Eriksson & Allie, 1989), which is a primary example of online secondary path estimations, was used. The main purpose is to design another LMS filter in order to eliminate a white noise. This filter compensates the effects of secondary path. Figure 3.9 is the block diagram representation of online secondary Eriksson Model.

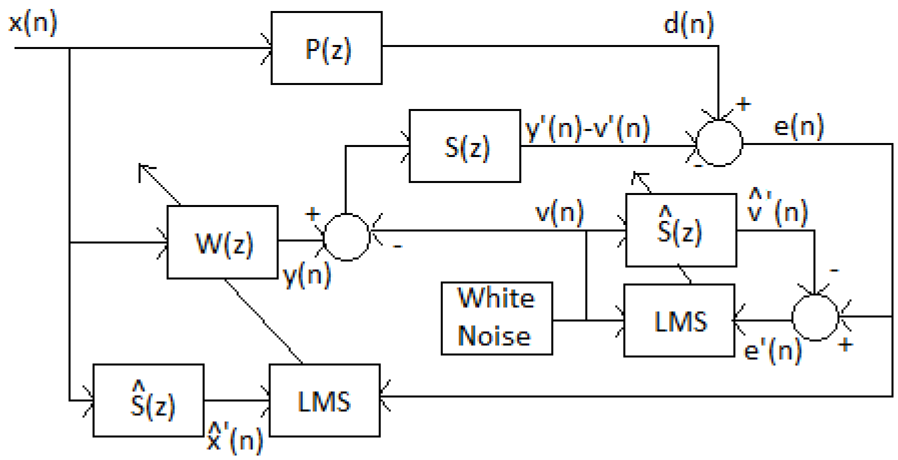

Figure 3.9 Eriksson secondary path model integrated in the FxLMS algorithm

In Figure 3.9, y(n) is the output of the main LMS filter. v(n) is the white noise signal, d(n) is the desired signal, e(n) is the error signal, $y^{'}(n)$ is the filtered version of y(n), and $v^{'}(n)$ is the filtered version of v(n). In the second LMS filter, v(n) is the input, e(n) is the desired signal and $e^{'}(n)$ is the error signal. Hence, Equation (3.5) and Equation (3.6) can be written.

$$e(n) = [d(n) - y^{'}(n)] + v'(n) \tag{3.5}$$

The error signal for the second LMS filter is $e^{'}(n)$.

$$e^{'}(n) = e(n) - \hat{v}^{'}(n) = [d(n) - y^{'}(n)] + [v^{'}(n) - \hat{v}^{'}(n)] \tag{3.6}$$

Weight update rule for the second LMS filter becomes (3.7)

$$\hat{s}(n+1) = \hat{s}(n) + \mu_s e^{'}(n)v(n) =$$
$$\hat{s}(n) + \mu_s v(n)[d(n) - y^{'}(n)] + \mu_s v(n)[v^{'}(n) - \hat{v}^{'}(n)] \tag{3.7}$$

$\hat{S}(z)$ is the estimated version of $S(z)$. Therefore, $v^{'}(n) - \hat{v}^{'}(n) \rightarrow 0$ and whole secondary system becomes a simple LMS system, which eliminates the desired signal d(n) with the output signal $y^{'}(n)$. This situation is the one desired for the whole FXLMS filter system. Therefore these second filter weights are fed back to

$\hat{S}(n)$, as indicated in Figure 2.5. The reason of the usage of white noise is the fact that it contains all frequency values and prepares the ANC system for all possible secondary path effects

In the offline secondary path, on the contrary, these filter weights are fixed. The system is not robust and even if some small change occurs, whole system corrupts and it requires new weight calculations for the correct filter system. The calculation of adaptive secondary path coefficients, which is an online method, is one of the specialties of this project.

In this chapter, drill characteristics were derived. According to these characteristics, the boundaries and optimum values of NLMS filter parameters and LMS filter type were determined. Using these parameters, simulation results were documented. Afterwards, the project NLMS block diagram was introduced. MATLAB NLMS block and project NLMS block were compared. Simulation results of the system after connecting a microphone to the computer were also given. Finally, FxLMS algorithm was studied in a mathematical manner with derivations and equations.

CHAPTER FOUR
HARDWARE STRUCTURE

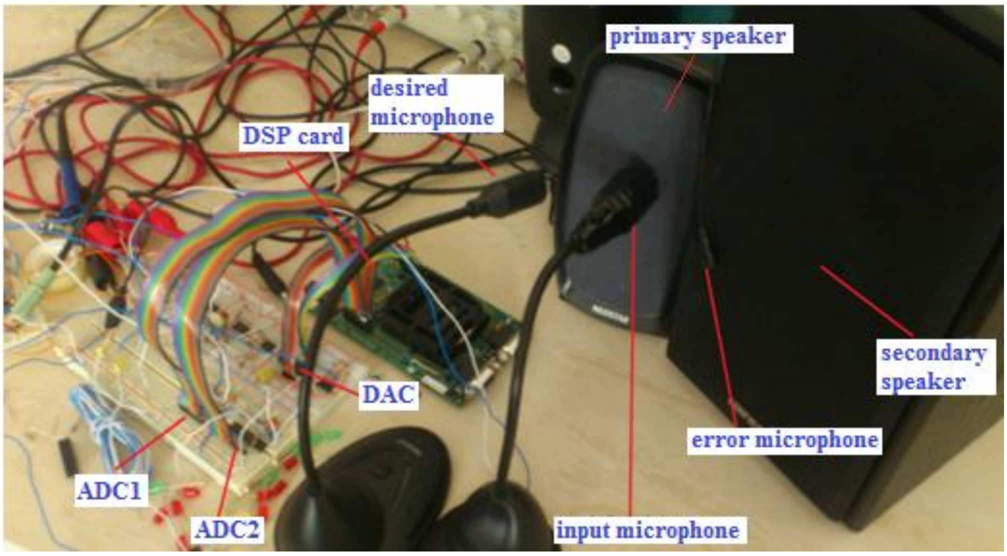

Figure 4.1 Hardware system, DSP card, speakers, microphones and other components

As the project hardware, 3 microphones, 2 speakers, 1 DSP board, and many electrical components were used as it is seen in Figure 4.1. The speakers represent the primary and secondary noise sources. While the first speaker generates the noise previously recorded drill sounds, second speaker output eliminates this noise. The first microphone was used as the input microphone, which records the original drill sound, and was placed right in front of the speaker in order to receive the sound directly. For the desired signal to be recorded, second microphone was placed on the left side of input microphone. Error microphone, on the other hand, used just to measure the signal magnitude in desired noiseless area. The distance representations are shown in Figure 4.2.

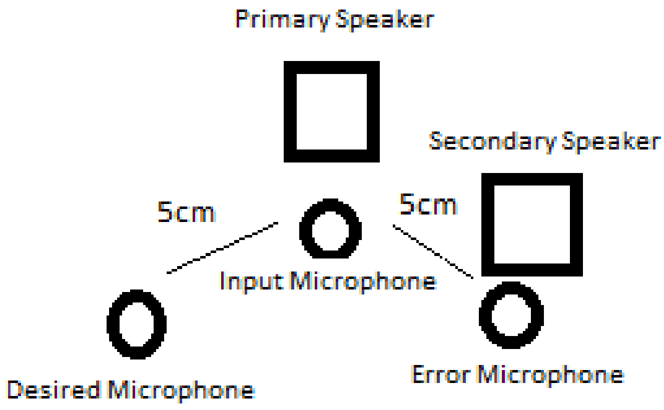

Figure 4.2 Hardware system, speakers and microphones positions

The error microphone must be placed in the mirror position of the desired signal because the noises at these microphones are supposed to be exactly same. This system was tested with 3 different distances, and the result is shown in Figure 4.3.

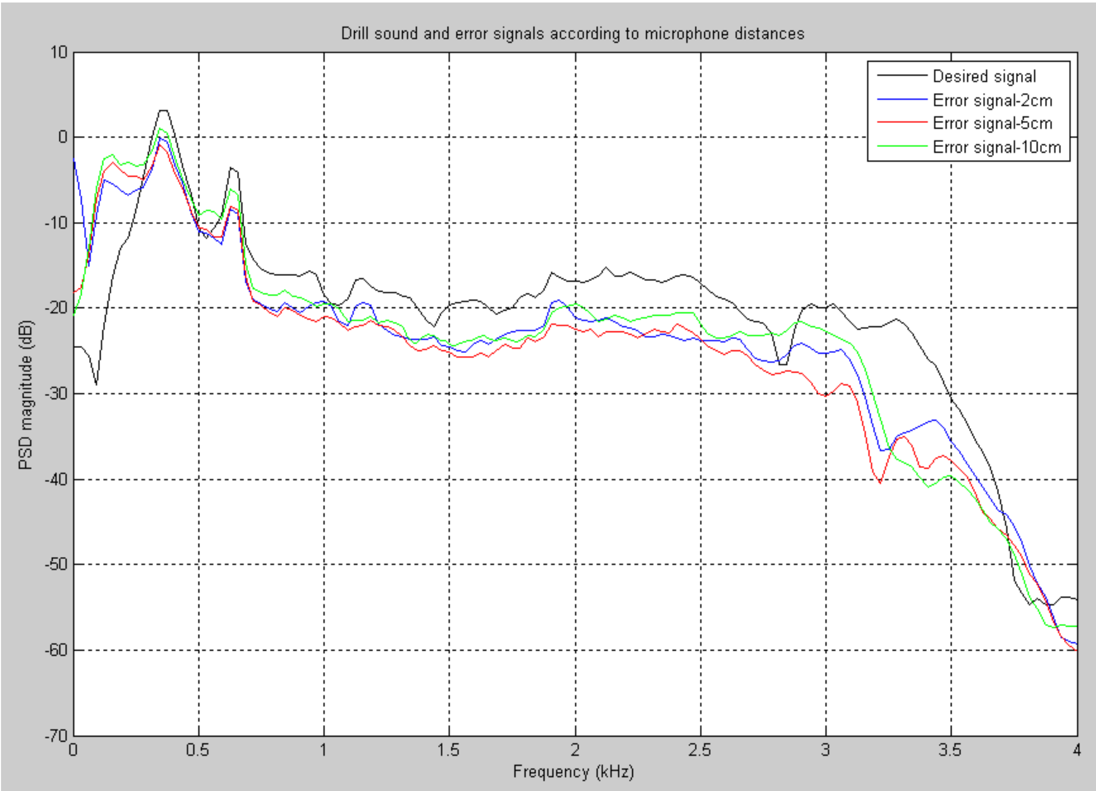

Figure 4.3 Power spectral densities of drill sound and error signals according to microphone positions

Too small distance gives bad performance because of proximity (boost) effect (Ciletti, 2004) while large distance also gives bad performance because of the signal loss. The optimum distance for this project is 5cm. Furthermore, according to a study (Mareau, Cazzolato, Zander & Peterson, 2008), noise-free area restricted with the signal wavelength, which results in 1cm noise-free area with high frequency drill sound components. Because of this, error microphone was selected so that its diameter was smaller than 1cm.

With this microphone positioning, noises were supposed to be exactly same on desired and error microphones in terms of magnitude and phase.

Secondary speaker receives analog signal. DAC system receives the output signal from DSP board in digital form, converts it into analog form and sends to the secondary speaker as the anti-noise. Other speaker (primary speaker), on the other hand, was used to create the main drill noise.

4.1 Microphone Input System

Project DSP board is digital based and all variables must be in digital form. As the analog to digital converter, internal ADC of the TMS320F28335 eZdsp was used (Tomar, 2011). The voltage range of this ADC is 0-3V. However, microphone read is much smaller than these ranges. Hence, Figure 4.4 was used as preamplifier to amplify the microphone input signal to 0-3V range.

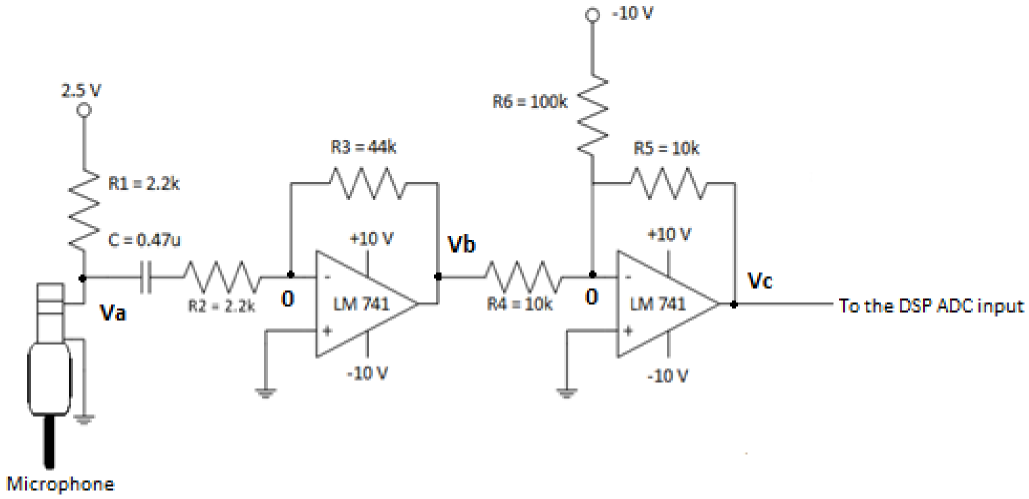

Figure 4.4 Preamplifier for amplification of the microphone input signal

V_a, V_b, and V_c show the voltage values at the specific nodes. Gain calculations are written in Equation (4.1) and Equation (4.2).

$$\frac{Va}{2.2+0.47s10^{-9}} = \frac{-Vb}{44} \tag{4.1}$$

$$\frac{Vb}{10} = \frac{10}{100} + \frac{-Vc}{10} \tag{4.2}$$

While calculating gain, term $\frac{10}{100}$ is neglected. Then Equation (4.3) and Equation (4.4) can be written.

$$Vb = -Vc \tag{4.3}$$

$$\frac{Vc}{Va} = \frac{44}{2.2+0.47s10^{-9}} \tag{4.4}$$

In order for the s term to affect the decimator, it should at least ten times smaller than 2.2.

$$0.47jw10^{-9} \overset{\sim}{\Rightarrow} 0.22 \tag{4.5}$$

Minimum required frequency for this is (4.6).

$$w = 468MHz \tag{4.6}$$

The capacitor used in much smaller frequencies has no effect to the gain. Therefore, system gain is (4.7).

$$Gain = \frac{44}{2,2} = 20 \tag{4.7}$$

4.2 TMS320F28335 DSP Card

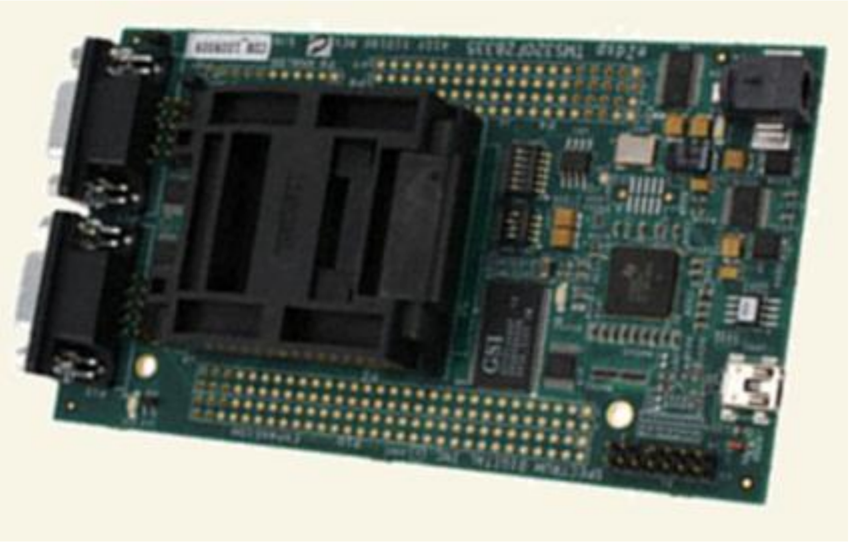

Figure 4.5 TMS320F28335 eZdsp board (Tomar, 2011)

Figure 4.5 is a clear image of the TMS320F28335 eZdsp board. It has 32-bit architecture. CPU is up to 150MHz. It has 87 I/O feeding with 3.3V. These IOs can be used for PWM outputs, timers, and event catchers. The board has no analog output pin. It has 16 of each 256K, 128K, and 64K flash memories. These memories were used to store project data. Besides the board has 128-bit encryption key (Texas Instruments, n.d.).

After the operations performed in DSP board, the resulting signal is in digital form. It should be converted to the analog form in order for the speaker to sound it. This process was done by R-2R ladder DAC system (The R/2R DAC, n.d.).

4.3 Speaker

Secondary noise speaker is Edifier R1900T II. It is a professional speaker. The most important thing in the speakers is its frequency response, which is very similar in both primary and secondary noise speakers. Edifier Cruiser (n.d.) indicated the frequency response of the edifier speaker.

The speaker has no particular response for the frequencies below 20Hz, and its response is stronger for frequencies higher than 1 kHz. This non-stable frequency response results in signal distortion. For example, a pure sinusoidal signal was recorded and its PSD was calculated. Then this signal was transmitted to the speaker and the speaker output signal and its PSD was recorded. Figure 4.6 shows both the pure sinusoidal signal PSD and PSD of a sinusoidal signal coming out from the speaker.

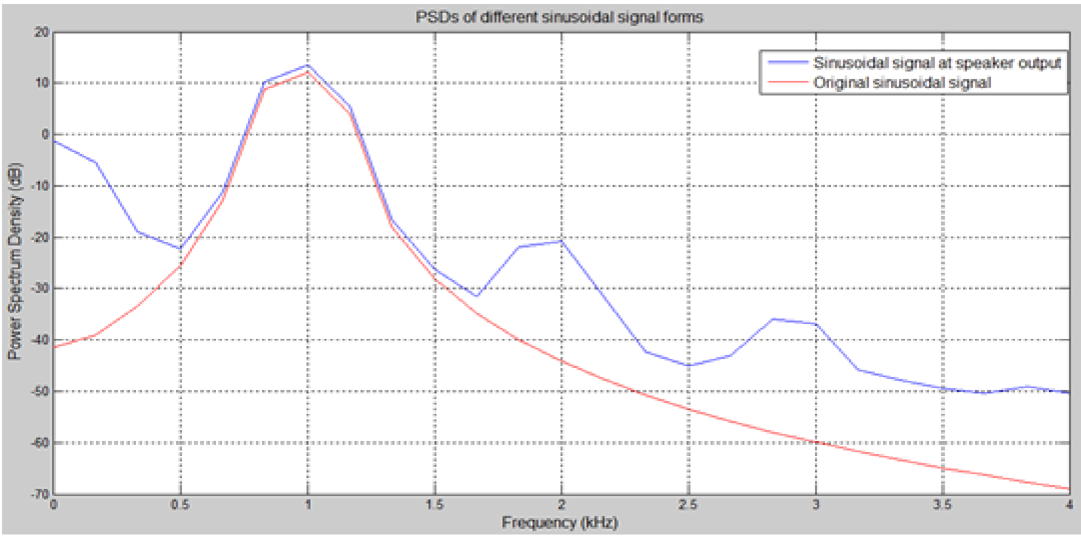

Figure 4.6 Power spectral densities of pure sinusoidal signal and a sinusoidal signal recorded at the output of a speaker

The speaker has large response in both low and high frequencies; therefore, the speaker output sinusoidal signal differs from the pure sinusoidal signal in the indicated manner.

The hardware components and their characteristics were studied in this chapter. Hardware positioning of speakers and microphones were calculated. DSP card, speaker, microphone preamplifiers, their features and effects on the system were stated. Preamplifier gain was calculated, and speaker effect on a pure sinusoidal signal was plotted.

CHAPTER FIVE

PROJECT DESIGN AND RESULTS

5.1 Offline Normalized LMS Algorithm Results

In the offline normalized LMS algorithm, all simulations were done in the MATLAB environment, no real time component included. As indicated before in offline NLMS algorithm the step size and the filter length variables are very crucial. Here, step size and filter order were determined as 0.001 and 256 respectively. This offline normalized LMS algorithm were applied to 40 different drill noises, 20 of them are free running and 20 remaining are while drilling an obstacle. The frequency and time domain performances in one sample from these categories are shown in Figure 5.1. Since the signal magnitudes are smaller than 1, PSDs of these signals are negative.

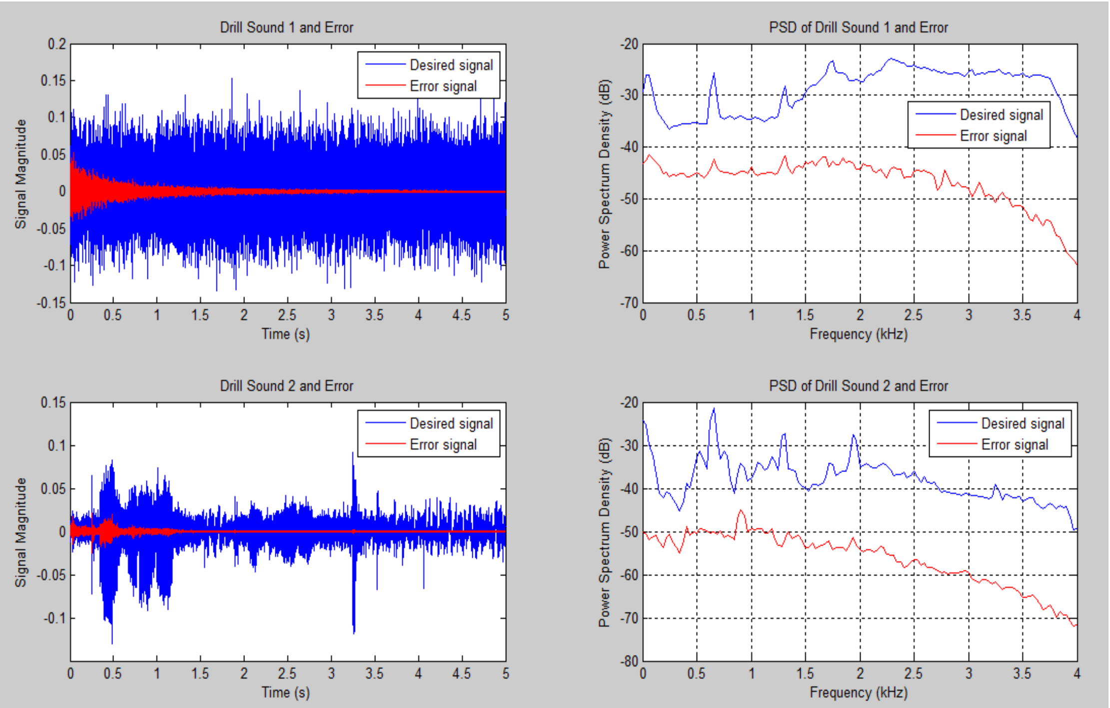

Figure 5.1 Time domain and power spectral densities of input and error signals for two different drill sounds in offline normalized LMS algorithm

Also the average performance for these 40 different drill sounds was calculated and Table 5.1 was created.

Table 5.1 Offline normalized LMS algorithm performance at different frequencies

	100Hz	**500Hz**	**1000Hz**	**2000Hz**	**3000Hz**	**Average**
Offline NLMS	24.6dB	17.1dB	11.6dB	16,6dB	18.5dB	17.5dB

5.2 Simple Online Normalized LMS Algorithm Results

To see the microphone and DSP card performances, simple online NLMS algorithm was used. The noise was eliminated in the card. Therefore, secondary speaker is not used, and there is no secondary path effect. The noise is received by a microphone continuously into the DSP board and NLMS algorithm runs in the board. It is the most basic form of online NLMS algorithm. Again 40 different drill sounds were used with the primary speaker for performance test. Figure 5.2 represents the one of these simple online NLMS algorithm performances.

The reason why the signal doesn’t have a particular peak is the fact that RTDX channels, which are used to read the internal DSP board memory, cannot store large amount of values because of small buffer size. Figure 5.2 shows only the 256 samples from the drill sound. Average performances were calculated by using 10 different buffer contents for a single noise. The RTDX buffers get full after these 10 data stores. The overall average performance for these 40 different drill sounds was calculated and Table 5.2 was created.

Table 5.2 Simple online normalized LMS algorithm performance at different frequencies

	100Hz	**500Hz**	**1000Hz**	**2000Hz**	**3000Hz**	**Average**
Basic Online normalized LMS	8dB	9dB	12dB	8dB	7dB	10dB

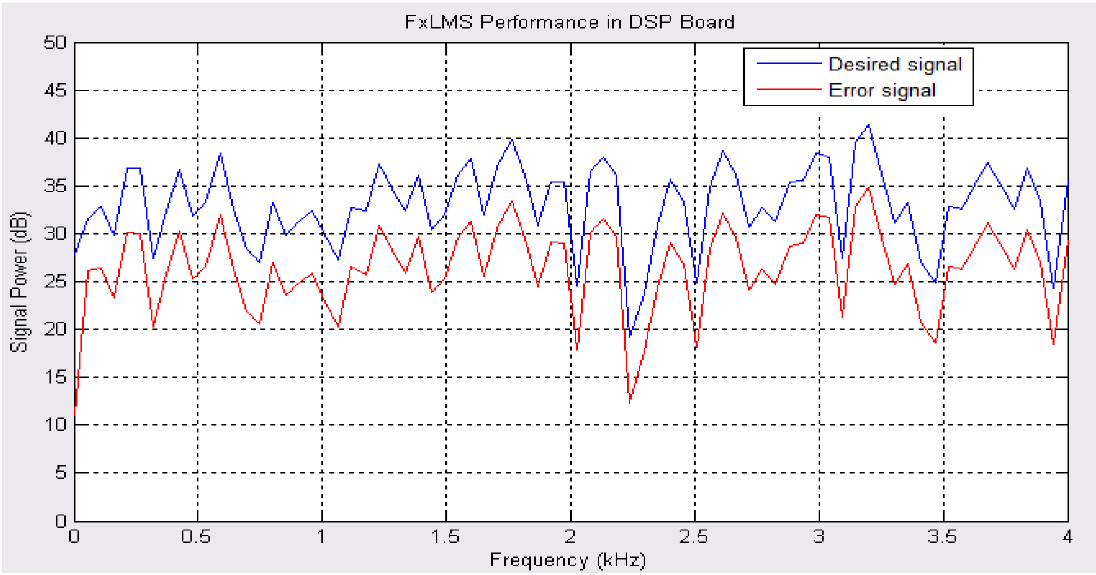

Figure 5.2 Power spectral densities of input and error signals in the most basic online normalized LMS algorithm

Note that if a drill sound has a peak at some frequencies, the elimination of the noise components which has this peak frequency is greater than the remaining ones. This elimination differs from 5dB to 30dB. The results differ from the offline NLMS algorithm because input microphone and primary speaker is included to the system, and whole adaptive system runs in the DSP board.

5.3 Online Normalized FxLMS Algorithm

Simple LMS algorithms fail when dealing with real time active noise controls. Therefore, FxLMS filters are used in real time implementations. Normalized FxLMS algorithm is an advanced version of FxLMS algorithm. Input noise is normalized so that error signal and desired signal have better synchronization than they have in normal FxLMS filter design. In Figure 5.3 and Figure 5.4, MATLAB normalized FxLMS block diagrams are shown.

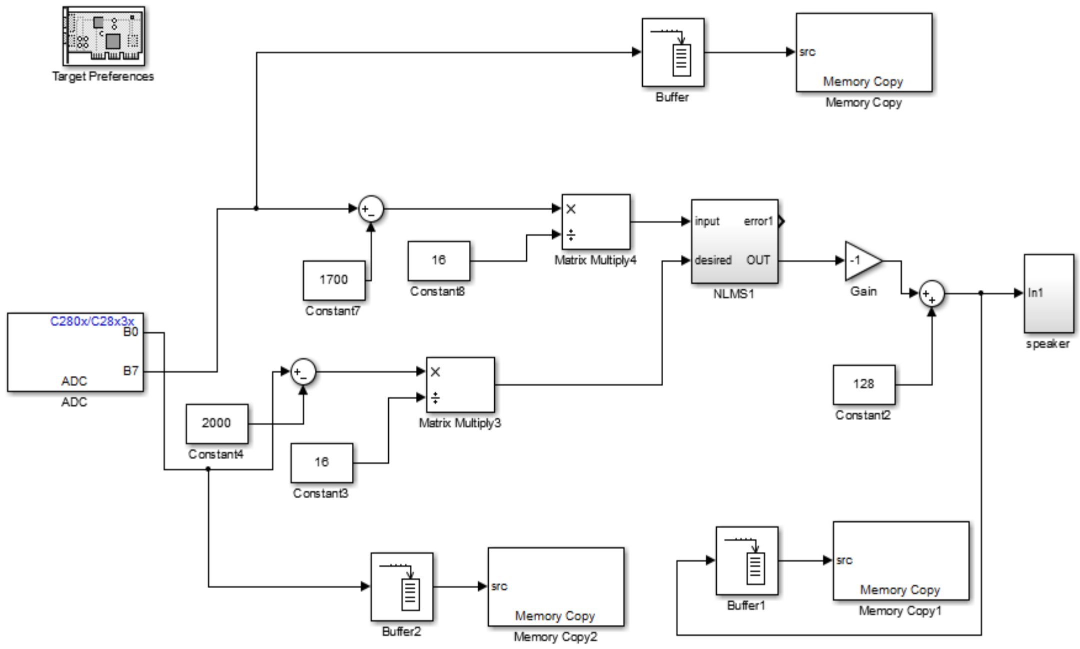

Figure 5.3 MATLAB block diagram of normalized FxLMS algorithm

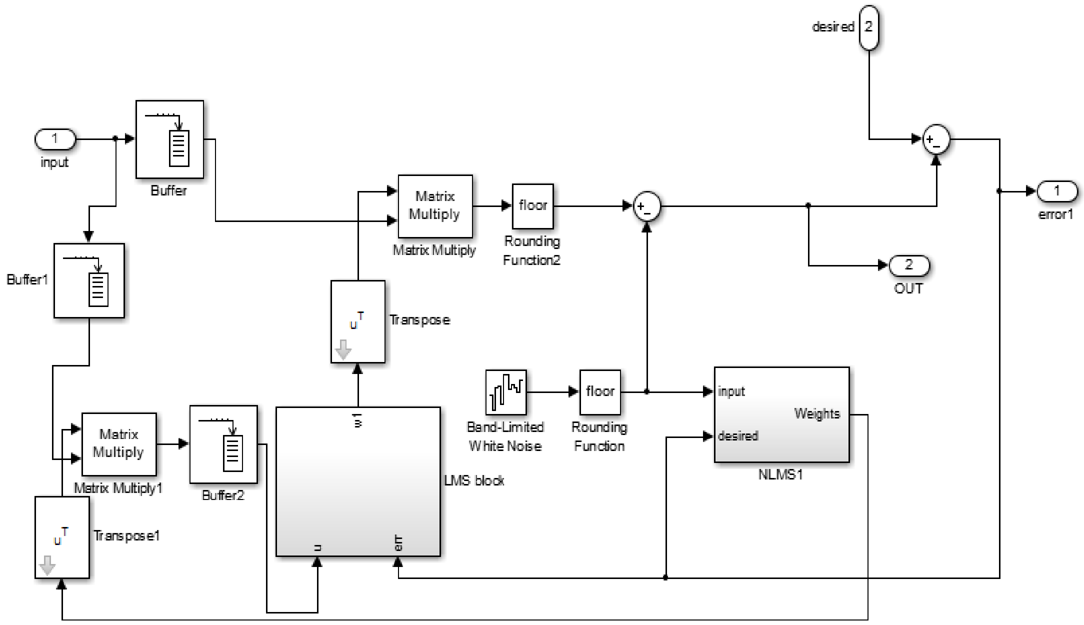

Figure 5.4 Inside of the NLMS1 subsystem

In the block diagram, **memory copy** blocks were used in order to observe the input, desired, and error signal characteristics. ADC block represents the DSP board analog to digital converter, and it has 12-bit resolution. Its range is between 0 and 4096. Hence, **constans3, constant4, constant7,** and **constant8** were used in order to change 12-bit bias input and desired signals to 8-bit non-bias (0 DC) form. **Constant2,** on the other hand was used in order to add bias to the output signal. Afterwards, the output signal is transmitted to the DAC converter, which is a R-2R ladder.

In the NLMS1 sub block, 64, 128 and 256 filter orders were tested and there was negligible different beyond 128. Hence, 128-order online secondary path (Erickson's Model) was used. In this model, a second LMS filter is placed inside the existing LMS filter. Then, its weights convolve with the input signal, continuing with normal LMS block. This block is same as the Figure 3.6.

5.4 Online Normalized FxLMS Algorithm Results

Whole online NFxLMS was setup. The noise was tried to be minimized in an open area. Now there is a secondary path included because of the secondary noise speaker. Overall performances were also tested with 40 drill sounds. One sample of each category (free running and drilling) was plotted in Figure 5.5 and Figure 5.6 in order for a better understanding of the performance. Since the whole system cannot adapt for large frequency range, the performance differs from frequency to frequency. Optimum filter parameters and hardware specifications, microphone and speaker frequency responses, cannot be the same for all frequencies. Indeed, at some frequency, especially low frequencies, the error has more power than the desired signal. Remembering speaker has unwanted response at low frequencies, it is reasonable that error signal have more power than the desired signal at these frequencies.

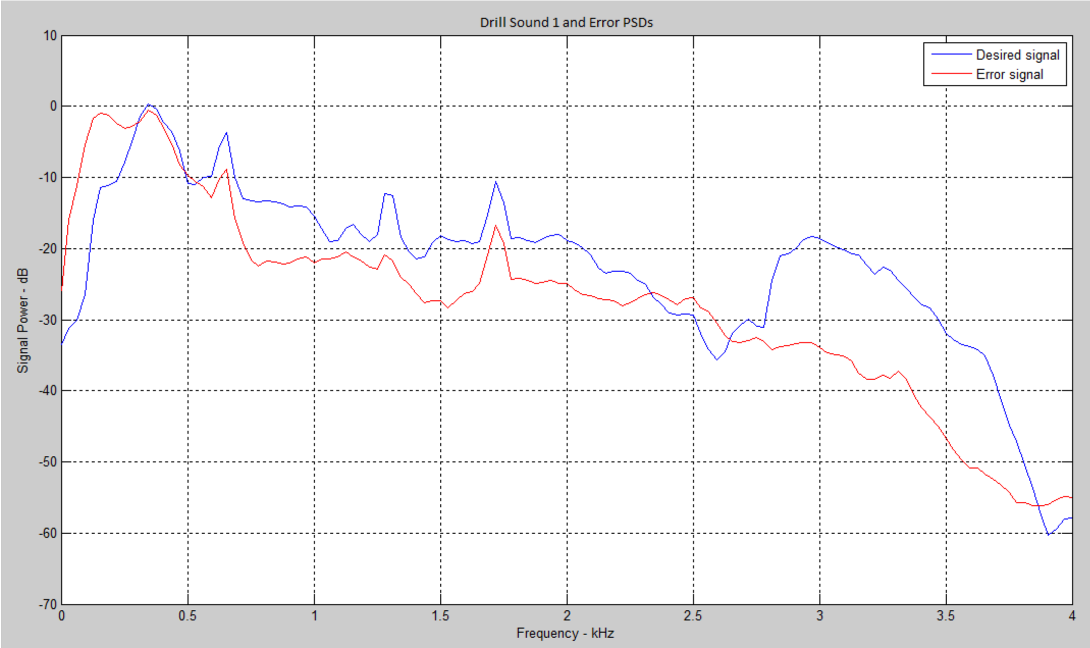

Figure 5.5 Power spectral densities of free-running drill sound and corresponding error signal in normalized online FxLMS algorithm

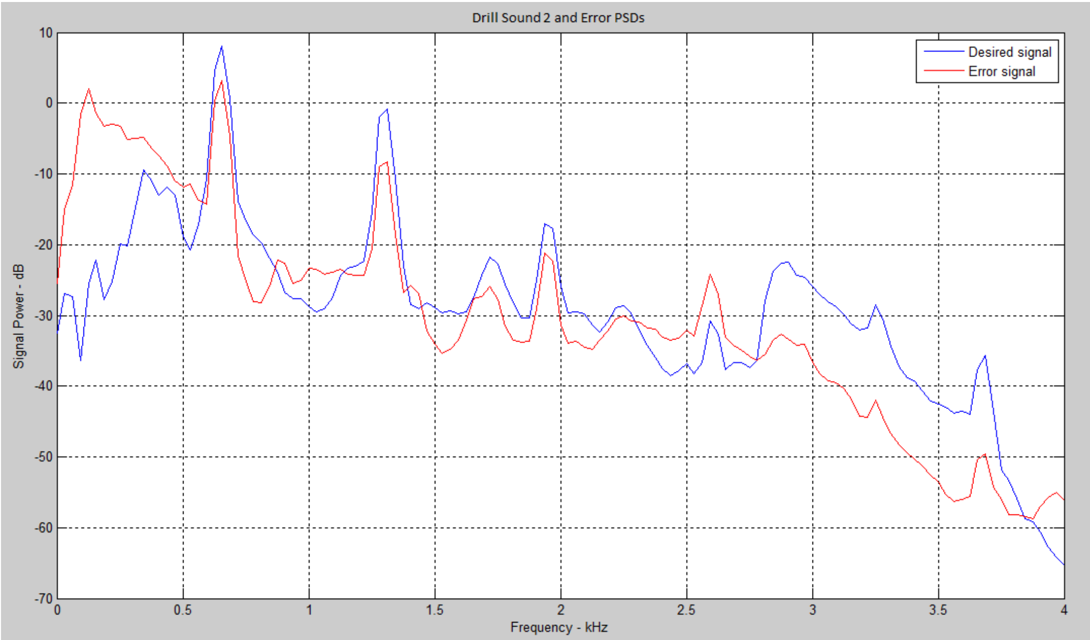

Figure 5.6 Power spectral densities of drilling drill sound and corresponding error signal in normalized online FxLMS algorithm

Average SNR performance for all these 40 drill sounds were calculated as 6dB, which is a good performance considering general FxLMS performances (Zou, Antila, Lankila & Kataja, 2009). It should be noted that the negative SNR values at low frequencies are because of the speaker effect.

5.5 Narrowband Normalized FxLMS Algorithm

In the most studies, FxLMS filters are used for band limited signals. Normally, the noise to be eliminated may have different frequency components as drill sounds. A common NLMS algorithm cannot work flawless for all these frequencies. Considering periodic noise PSD characteristics, the noise has some peaks at particular frequencies. At these frequencies the noise has more power than the remaining range densities. Therefore, signal can be band limited to these peak frequencies. Figure 5.7 shows the band pass filter used for band limiting process for this project, knowing that the noise has one of the main peaks at 1.9 kHz.

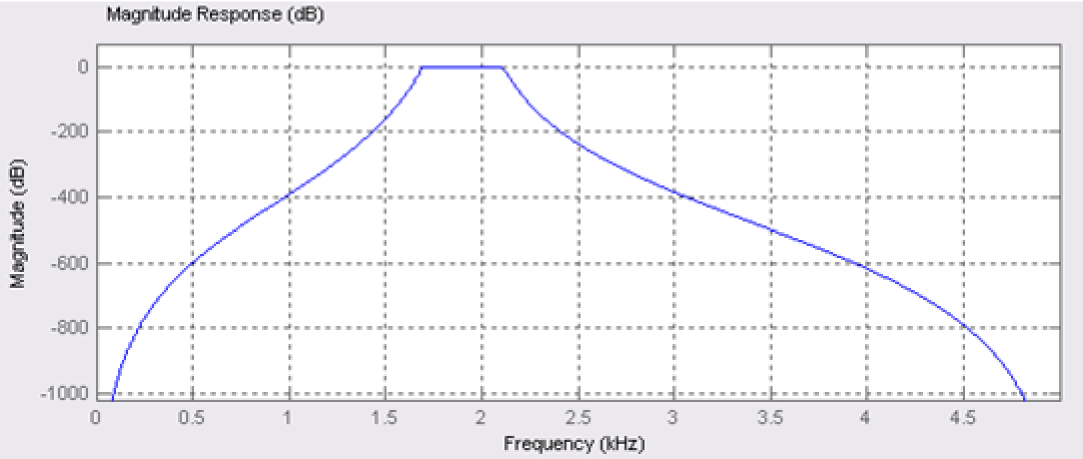

Figure 5.7 Butterworth band pass filter centered between 1.7 kHz and 2.1 kHz

This filter was generated in the MATLAB ‘digital filter design’ block. The performances in DSP board and in the environment are documented in the results section.

5.6 Narrowband Normalized FxLMS Algorithm Results

As it is mentioned before, broadband FxLMS algorithm is hard to implement because one common algorithm cannot fit for all frequency component characteristics. Therefore, this algorithm is applied to frequencies which have peak magnitudes. This results in band limited filtering. Figure 5.8 and Figure 5.9 were formed considering there was a peak value at 1.9 kHz. The first one is PSD performance in the MATLAB simulation, and the second one is PSD performance in the real environment. The filter range was between 1.7-2.1 kHz.

It is clearly seen that in MATLAB simulations, there is no frequency component other than the filtered ones in the input signal. This is because of butterworth filter. NLMS adaptive system only deals with the desired frequency range. Therefore, the desired signal and the error signals are almost same at out of butterworth frequency range. It should be noted that the distortion is because of the NLMS filter parameters, such as step size and regulation factor. To conclude, the performance is around 20dB at the peak frequency, which is an expected performance since the system runs in simulation environment.

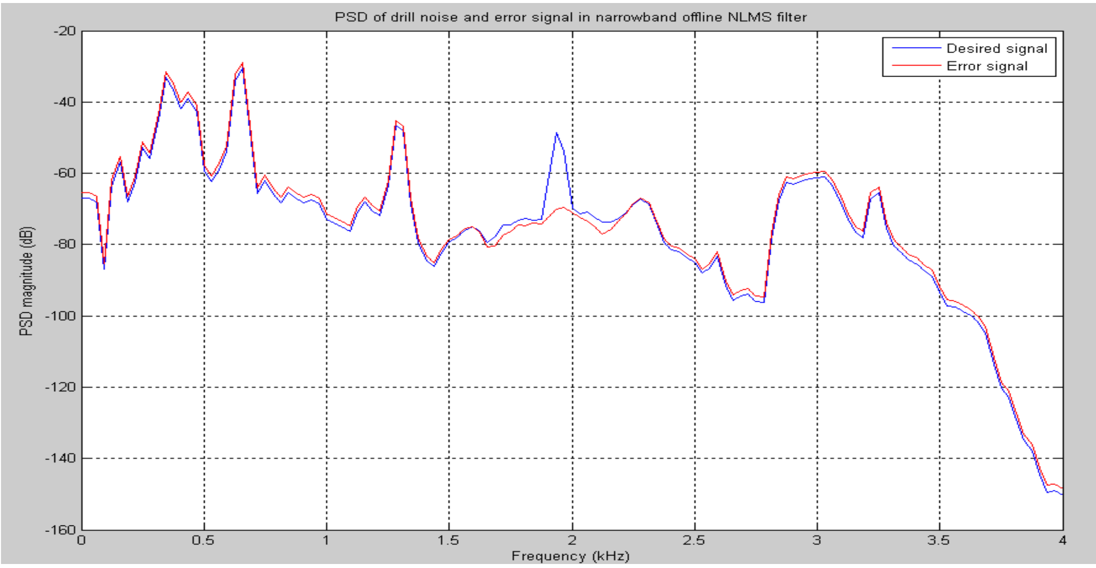

Figure 5.8 Power spectral densities of input and error signals in normalized FxLMS algorithm inside MATLAB simulation environment

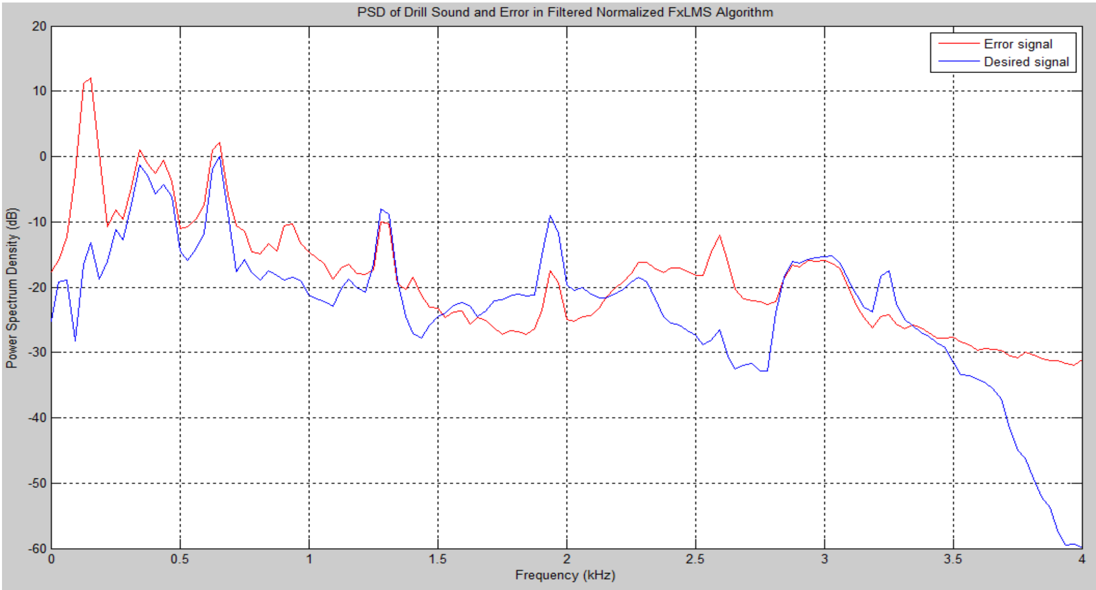

Figure 5.9 Power spectral densities of input and error signals in normalized FxLMS algorithm inside real environment

In Figure 5.9, the power spectrum density of desired signal is greater than the one in Figure 5.8. This is because in real time applications, normalization method is used. In the real environment, there is an obvious reduction in the frequency range 1.7-2.1 kHz around 8dB, better performance than the broadband FxLMS at this frequency range, 5dB. Other than these range, the error signal power usually exceeds the desired signal power. The reason why the signals do not exactly fit each other is the fact that the secondary speaker distorts the filter output, which has actually no frequency values other than 1.7-2.1 kHz range in the DSP board. The clearest proof of this distortion is the unrelated peak occurred at 150 Hz component of error signal. It should also be noted the signal magnitudes are less than 1; therefore, PSDs of these signals are negative.

5.7 Normalized FxLMS Algorithm with Acoustic Feedback

Bare hardware system has bad effects on the project because the secondary speaker output distorts the desired signal because of the open environment. This effect is called as ***acoustic feedback.*** To eliminate the negative effect of this feedback, an additional FIR filter was designed. However, the acoustic feedback filter is not common in ANC applications because of its complexity.

In Figure 5.10, F(z) represents the acoustic feedback effect, while $d^{'}(n)$ is the received desired signal and d(n) is the actual desired signal. In the real environment, received signal is calculated as (5.1).

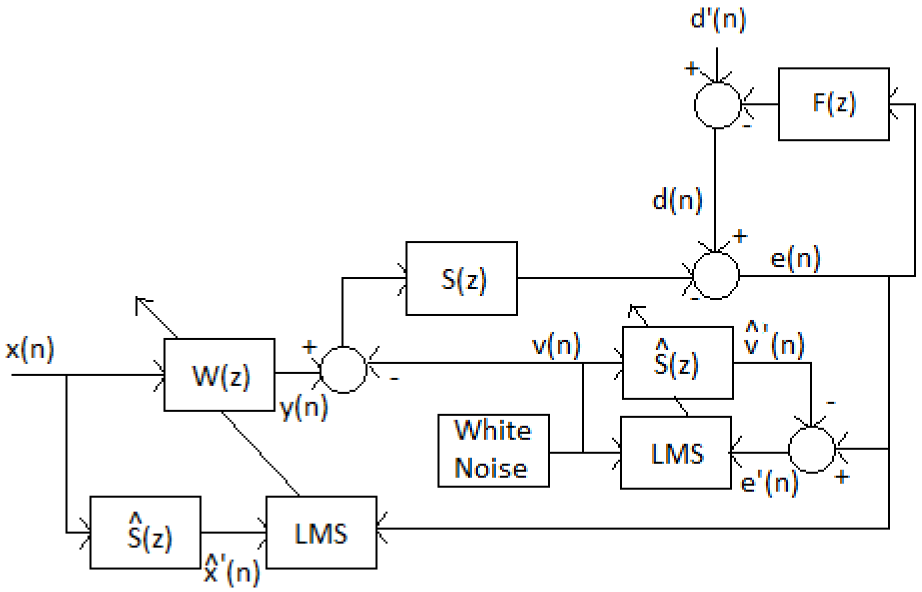

Figure 5.10 Block diagram of normalized FxLMS system with Acoustic Feedback Filter

$$d^{'}(n) = d(n) + f(n) * e(n) \tag{5.1}$$

Where '*' is the convolution process and f(n) is the time coefficients of the acoustic feedback. F(z) block in Figure 5.10 eliminates this distortion occurred in desired signal.

The filter length of F(z) is important. The boundaries are determined by the system complexity. The system complexity is mainly related with the maximum filter length of the overall system. Therefore, the first normalized FxLMS filter length, which is 256, is the overall system complexity. The secondary path filter length is negligible compared to the first one. The DSP card clock frequency is 150 MHz. Then the minimum time required for the consecutive samples is calculated as (5.2).

$$t = \left(\frac{1}{150} . 10^{-6}\right) . (256) \cong 0.00000171 \tag{5.2}$$

If the equation in (5.1) is considered with a filter length n, then the additional filter results in 2n complexity because the convolution process has n number of summation and n number of multiplication. The sampling rates of the digital blocks are 0.0001 second. As a result of (5.3), the filter length of the acoustic feedback can be 29. However, it should be multiple of 2. Therefore, it was determined as order 16.

$$Filter\_Order_{max} = 2n = \frac{0.0001}{0.00000171} \cong 58 \tag{5.3}$$

f(n) parameters now can be fixed at optimum values. Let f(n) be as (5.4).

$$f(n) = f_0 + f_1 z^{-1} + f_2 z^{-2} + \ldots + f_{15} z^{-15} \tag{5.4}$$

Now the matrix equation of feedback filter coefficients can be written as (5.5).

$$\begin{bmatrix} f_0 & 0 & 0 & \\ f_1 & f_0 & 0 & \ldots \\ f_2 & f_1 & f_0 & \end{bmatrix}_{16x16} \begin{bmatrix} e_0 \\ e_1 \\ \vdots \end{bmatrix}_{16x1} = \begin{bmatrix} d'_0 - d_0 \\ d'_1 - d_1 \\ \vdots \end{bmatrix}_{16x1} \tag{5.5}$$

To digitize d(n), $d^{'}(n)$, and e(n) values, a pure sinusoidal waveform was implemented from the secondary speaker. All these values were determined and then f(n) parameters were derived.

5.8 Normalized FxLMS Algorithm with Acoustic Feedback Results

The implementation of this algorithm was defined in chapter five. The algorithm was applied to two different drill sounds. One of them was free running, and the other was drilling. The Figure 5.11 and Figure 5.12 show the PSDs of these drill sounds and corresponding error signals respectively. The effects of the acoustic path are clearly seen in these figures.

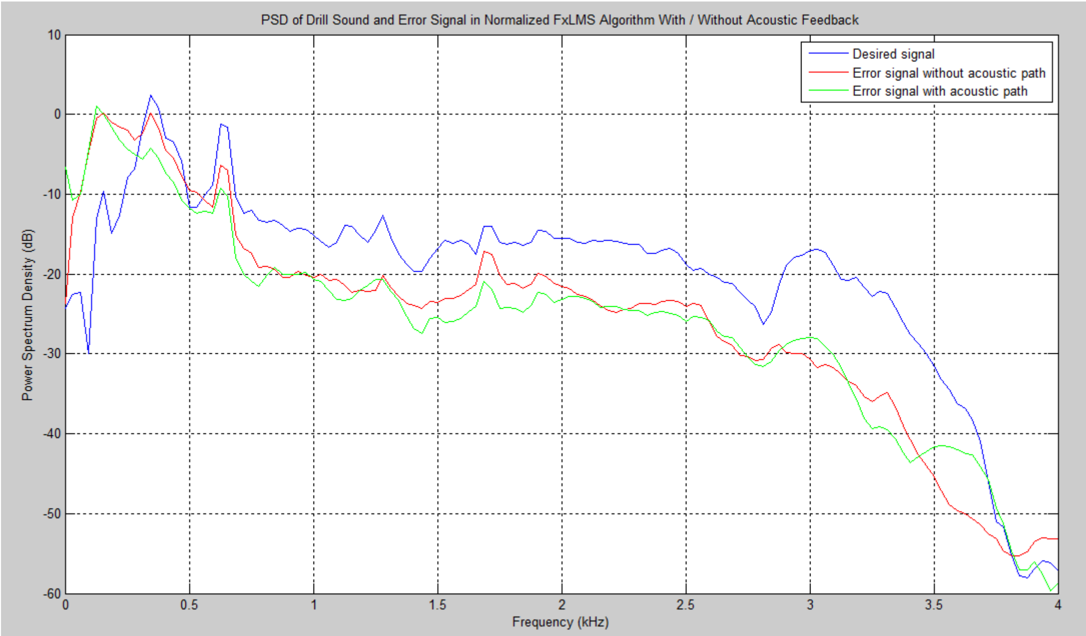

Figure 5.11 The PSDs of free-running drill sound and system error signal in normalized FxLMS algorithm with / without acoustic feedback

When this algorithm is compared to the normalized FxLMS algorithm, one can say that the overall performance is slightly better in adaptive system with acoustic feedback, 7dB. The elimination at peak values is better. The range, in which the error signal has higher power than desired signal, is narrower. These are the advantages of the acoustic feedback. The disadvantages are the fact that the system is not robust because the computational time is at its boundary, as indicated in chapter 5.7. Besides, the overall performance is not same for all drill sounds since the filter length is too small, which is 16. Sometimes, these parameters cannot initiate the overall system for some drill sounds. As a future work, the acoustic feedback filter length can be increased over 100 by using a fast DSP card. In this way, the total FxLMS performance can be increased in a robust manner.

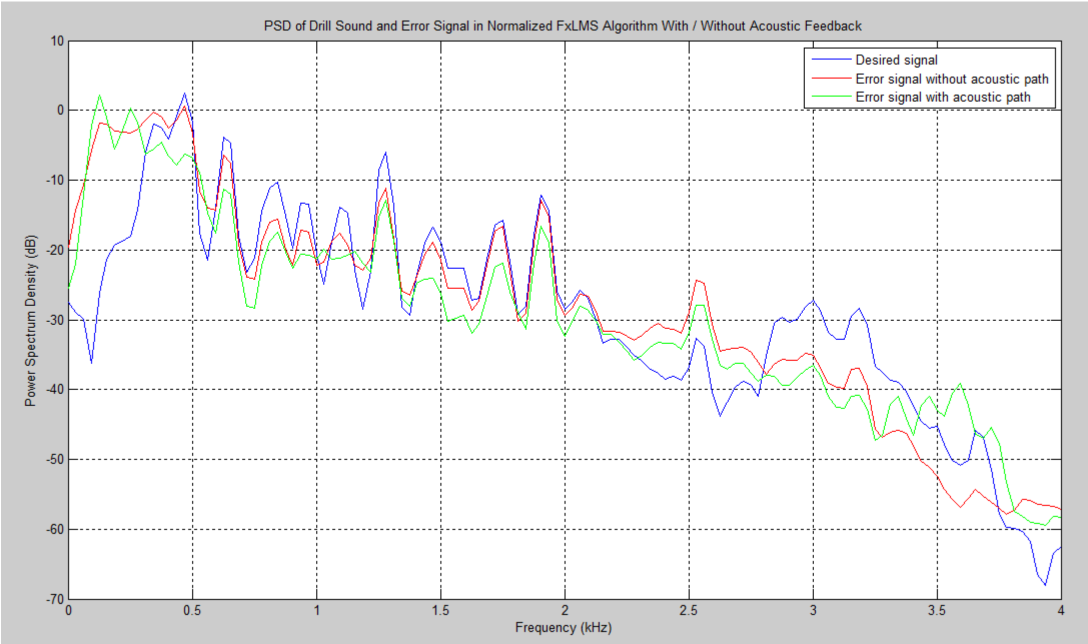

Figure 5.12 The PSDs of drilling drill sound and system error signal in normalized FxLMS algorithm with /without acoustic feedback

In this chapter, the block diagrams of the main ANC system, normalized FxLMS, was introduced with detailed information. Also the narrowband normalized FxLMS algorithm and the butterworth filter characteristics used in this algorithm were mentioned. The acoustic feedback effect was introduced. Its reason and possible solutions were defined. The acoustic feedback filter was realized in the project design. The optimum parameters, system complexity and the time boundary were deeply investigated. As the result part, 5 different algorithm results were documented. In the offline NLMS algorithm, 40 different drill sounds were applied as the input noise to the simulation, and the performance results of two different drill sounds were plotted. Also the average of the simulation results was added as a table. Then, a microphone was included to the system and whole system was transferred to the DSP board, called as simple online NLMS filter. The performance results were given in both table and graph formats. Using the main algorithm FxLMS and working in real-time, main system performances were obtained. Two different graph performances were plotted. Besides, average SNR performance of the algorithm for 40 different drill sounds was also plotted for easier comprehension. Then, narrowband filtering was mentioned. The offline and online results were obtained. Afterwards, the acoustic feedback was introduced and the overall system performance was recalculated. The advantages and disadvantages of this feedback were determined. To conclude, all filtering methods were used, their performances were calculated and plotted, and their comparisons with each other were done according to the performances, advantages, and disadvantages.

CHAPTER SIX
CONCLUSION

Considering active noise control, there are various application areas including all kinds of noises. In these areas, the software and hardware systems, methods, algorithms and environments differs from each other. Most of these applications are related with low frequency noises. In this project, dental drill noises, which have high frequencies, were in the focus area.

First of all, dental drill noise characteristics were investigated. The crucial parameters were determined and simulation blocks were prepared with these findings. Then different simulations were performed with different ANC methods and system parameters, such as step size and filter lengths. It was seen that these parameters affect the system performance and convergence speed. Finally, NLMS algorithm parameters were optimized as 0.001 for step size and 256 for filter length.

All simulations were performed in MATLAB environment. Then, to see the effect of the microphone and DSP card, the noise was received by a microphone and the adaptive normalized LMS filter was performed in the DSP card.

After extensive simulations, hardware system was designed according to the software needs. This design includes a DSP board, TMS320F28335 eZdsp, a primary noise speaker, a secondary noise speaker, a desired microphone, an input microphone, an error microphone and several electrical components. The positions of hardware components were determined, optimized and fixed according to calculations and experiments. The effects of the components to each other were also considered for better performance. To examine the speaker effect on the output signal, pure sinusoidal PSD was observed. The PSD of a pure sinusoidal waveform and the sinusoidal waveform at the output of the speaker were different.

Afterwards, an online secondary path was added in order to deal with magnitude and phase change occurred in filter output signal. As the secondary path, Eriksson method, which simply uses another LMS filter with white noise as its input signal, was used. This filter was developed as normalized LMS filter. Its order was optimized as 128. The step size is same as the first NLMS, which is 0.001.

Narrowband filter algorithms are more common in the literature, including the previous dental drill ANC research. Therefore, for a performance test and comparison, the drill noise was narrowed down with a butterworth filter. This method was characterized for a particular frequency range and filter parameters. Therefore, even if it gave a better performance than normalized FxLMS algorithm, it cannot guarantee this performance for all signals.

To overcome the acoustic feedback effect, which is the effect of error signal on the desired microphone, an additional FIR filter was designed. The system complexity and time boundaries were determined so that the filter length was defined. After defining the length, the coefficients of the filter were calculated. Then the whole system was run for some drill sounds. Although the performance was better than normalized FxLMS, the robustness and generalization was worse. The usage of better hardware setup was determined as the possible solution.

As the result part, the system was mainly tested in 3 different cases with 40 different drill sounds, namely offline NLMS, simple online NLMS, online normalized FxLMS. Average performance is 6dB in online normalized FxLMS, 10dB in simple normalized LMS, and 17.5dB in offline normalized LMS. As an extra, the band limited normalized FxLMS, and normalized FxLMS with acoustic feedback methods were applied. For a specific drill noises, 7dB-SNR in normalized FxLMS with acoustic feedback, and 8dB-SNR in narrowband normalized FxLMS were obtained. While the maximum performance of offline normalized LMS is 33dB, the maximum performance of online normalized FxLMS is 12dB. These performances were calculated for a particular frequency by using one of the drill sounds. According to Zou, Antila, Lankila & Kataja (2009), average normal FxLMS performance, in an open environment, is around 3.2dB, and this value was increased to 4.3dB by using an enhanced FxLMS algorithm on a truck cabin. To conclude, project has a better performance regarding signal to noise ratios.

Although there is a research about dental drill noise cancellation (Kaymak, Atherton, Rotter & Millar, 2007), this project has lots of specialties. Frequency range is not limited with a particular value. It is a broadband implementation. ANC algorithm method is normalized FxLMS, which has a better performance compared to a simple FxLMS. Secondary path filtering is online instead of offline. This can adapt all the environment changes, such as system transportation, environment noise & temperature changes and so on. Besides, system acoustic effect was also considered for better system performance.

6.1 Future Work

As the future work, a better DSP card with audio cable inputs and outputs may be used. The clock frequency of this card should be higher than the one used in this project so that the filter (first LMS, second LMS, and acoustic feedback filter) lengths can be increased without lagging problem. For the speakers and microphones, more professional ones may be used. These hardwares should have frequency response for low frequencies as well. In this way, signal characteristics do not corrupt at low frequencies. Finally, an additional decorrelated system (Yifeng, 1997) may be implemented in order to eliminate the effects of speakers on unassociated microphones.

REFERENCES

Abdel, M., Ahmadi, G., & Loo, F. T. C. (1990). A study on the mechanics of fatigue-dominated friction noise. *Journal of Vibration and Acoustics, 112* (2), 222-229.

Acoustic attenuators (n.d.). Retrieved September 20, 2012, from http://noisecontrolengineering.com.au/acoustic-attenuators/.

Adaptive signal processing lecture 2 (n.d.). Retrieved November 1, 2012, from http://www.eit.lth.se/fileadmin/eit/courses/ett042/LEC/notes2.pdf.

Arulampalam, M. S., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *Signal Processing, IEEE Transactions on, 50* (2), 174-188.

Barner, K. E. (2009). *Adaptive filtering method of steepest descent*. Retrieved November 2, 2012, from http://www.eecis.udel.edu/~barner/courses/eleg636/notes/pdf8.pdf

Bi, R., Li, K. M., Chen, J., & Liu, Z. S. (2011). *Silencing your car with a hybrid muffler*. Retrieved September 20, 2012, from http://www.acoustics.org/press/161st/Rong.html.

Bodson, M., Jensen, J. S., & Douglas, S. C. (2001). Active noise control for periodic disturbances. *Control Systems Technology, IEEE Transactions on, 9* (1), 200-205.

Carme, C. (1988). A new filtering method by feedback for ANC at the ear. *Proceedings of Inter-Noise*, 1083-1086.

Ciletti, E. (2004). *From rhythm track to the mix: The art of sonic translation*. Retrieved December 1, 2012, from http://www.tangible-technology.com/articles/loudness/bassmanage.htm.

Debye, P. (1909). Nӓherungsformeln für die Zylinderfunktionen für grosse Werte des Arguments und unbeschrӓnkt verӓnderliche Werte des index. *Mathematische Annalen, 67*, 535-558.

Dmitriev, V. I. (2012). *Wiener-Hopf equation*. Retrieved November 1, 2012, from http://www.encyclopediaofmath.org/index.php/Wiener-Hopf_equation.

Edifier Cruiser (n.d.). *R1900TII06 2.0-channel multimedia speaker product feature*. Retrieved January 8, 2013, from http://www.dinodirect.com/cruiser-edifier-r1900tii06-2-0-channel-multimedia-speaker-black-p32676405.html.

Elliott, S. J., & Nelson, P. A. (1993). Active noise control. *IEEE Signal Processing Magazine, 10* (4), 12-35.

Eriksson, L. J., & Allie, M. C. (1989). Use of random noise for on-line transducer modeling in an adaptive active attenuation system. *The Journal of the Acoustical Society of America, 85* (2), 797-802.

Forsgren, F. (2011). *Active noise control in forest machines*. Retrieved October 10, 2012, from http://umu.diva-portal.org/smash/get/diva2:451753/FULLTEXT01.pdf

Gan, W. S., & Kuo, S. M. (2003). Integrated active noise control communication headsets. *Proceedings of the International Symposium, 4*, 1-4.

Goines, L., & Hagler, L. (2007). Noise pollution: A modern plague. *Southern Medical Journal, 100* (3), 287-294.

Homma, K. (2004). *Compact integrated active-passive approach for axial fan noise control*. Retrieved October 9, 2012, from http://scholar.lib.vt.edu/theses/available/etd-09232004-204237/.

Kaymak, E., Atherton, M., Rotter, K., & Millar, B. (2007). Active noise control on high frequency narrow band dental drill noise: Preliminary results. *14th International Congress on Sound & Vibration*, Cairns, 26.

Klippel, W. (n.d.). *Prediction of speaker performance at high amplitudes*. Retrieved November 14, 2012, from http://www.klippel.de/uploads/media/Prediction_of_speaker_performance_at_high_amplitudes_01.pdf.

Kuo, S. M., & Chakravarty, S. R. (2006). Application of active noise control for reducing snore. *Acoustics, Speech and Signal Processing, ICASSP 2006 Proceedings. 2006 IEEE International Conference on, 5* (1).

Kuo, S. M., & Morgan, D. R. (2008). Active noise control. *Springer Handbook of Speech Processing* (1st ed.) (1001-1018). Berlin: Springer Berlin Heidelberg.

Least Mean Squares (LMS) algorithms (adaptive filter toolkit) (n.d.). Retrieved November 19, 2012, from http://zone.ni.com/reference/en-XX/help/372357A-01/lvaftconcepts/aft_lms_algorithms/.

Mareau, D., Cazzolato, B., Zander, A., & Petersen, C. (2008). A review of virtual sensing algorithms for active noise control. *Algorithms, 1*, 69-99.

Maybeck, P. S. (1990). *The kalman filter: An introduction to concepts*. Autonomous Robot Vehicles (3rd ed.) (194-204). New York: Springer New York.

Olson, H. F., & May, E. G. (1953). Electronic sound absorber. *Acoustical Society of America Journal, 25* (6), 1130-1136.

Passive flue gas silencer with mineral fibre absorber (n.d.). Retrieved September 20, 2012, from http://www.kutzner-weber.de/gb/products/silencers/passive-silencers.html.

Ramachandran, V. R., Panahi, I. M. S., & Perez, E. (2008). Active reduction of high-level acoustic noise on a fmri test-bed using labview and FPGA platforms. *Acoustics, Speech and Signal Processing, 2008, ICASSP 2008, IEEE International Conference on*, 1517-1520.

Raphael, V. R. (n.d.). *Algorithms for ANC*. Retrieved October 29, 2012, from http://www.geocities.ws/ranjit_raphael/algos.html.

Raphael, V. R. (n.d.). *The FxLMS algorithm*. Retrieved October 29, 2012, from http://www.geocities.ws/ranjit_raphael/FXLMS.html.

Schmidt, G. (n.d.). *Adaptive filters – algorithms (part2)*. Retrieved Novermber 14, 2012, from http://www.dss.tf.uni-kiel.de/en/teaching/lectures/lecture-slides/adaptive-filters/af_06_algorithms_2.

Schumacher, T., Krüger, H., Jeub, M., Vary, P., & Beaugeant, C. (2011). Active noise control in headsets: A new approach for broadband feedback ANC. *Acoustic, Speech and Signal Processing*, 417-420.

Tahir, A. M., Wataru, M., & Akinori, N. (2011). *Applications of adaptive filtering: Recent advancements in active noise control*. Adaptive Filtering Applications, Dr. Lino Garcia (Ed.), ISBN: 978-953-307-306-4, InTech. Retrieved November 20, 2012, from http://cdn.intechopen.com/pdfs-wm/16113.pdf.

Texas Instruments (n.d.). *Digital signal controllers data manual*. Retrieved December 19, 2012, from http://www.ti.com/lit/ds/symlink/tms320f28335.pdf.

Thanigai, P., Kuo, S. M., & Yenduri, R. (2007). Nonlinear active noise control for infant incubators in neo-natal intensive care units. *Acoustics, Speech and Signal Processing, ICASSP 2007, IEEE International Conference on, 1*, 109-112.

The R/2R DAC (n.d.). Retrieved December 19, 2012, from http://www.allaboutcircuits.com/vol_4/chpt_13/3.html.

Tomar, A. (2011). *Spectrum digital: TI TMS320F28 based eZdsp F28335 starter kit*. Retrieved December 18, 2012, from http://www.element14.com/community/docs/DOC-39394/l/spectrum-digital-ti-tms320f28-based-ezdsp-f28335-starter-kit

Veit, I. (1988). A lightweighted headset with active noise compensation. *Proceedings of Inter-Noise*, 1087-1090.

Vicente, L., Elliott, S. J., & Masgrau, E. (n.d.). *Fast active noise control for robust speech acquisition*. Retrieved November 25, 2012, from http://diec.unizar.es/~lvicente/personal/investigacion/Eurospeech99-v019.pdf.

Wang, L. (2008). *Active control of fluid-borne noise*. Retrieved October 9, 2012, from http://opus.bath.ac.uk/16742/2/Thesis_Lin_Wang.pdf.

Wang, L., Gan, W. S., & Kuo, S. M. (2005). Bass enhancement for automobile multimedia system using active noise equalizer. *Consumer Electronics, ISCE 2005, Proceedings of the Ninth International Symposium on*, 247-252.

Wheeler, P. D., & Smeatham, D. (1992). *On spatial variability in the attenuation performance of active hearing protectors*. England: Applied acoustics, Elsevier Science Publishers.

Wiener, N. (1964). *Extrapolation, interpolation, and smoothing of stationary time series: With engineering applications* (1st ed.). Canada: M.I.T. Press.

Yifeng, T. (1997). *Multiple reference active noise control*. Retrieved October 9, 2012, from http://scholar.lib.vt.edu/theses/available/etd-52597-0349/unrestricted/etdlast.PDF.

Zou, G., Antila, M., Lankila, A., & Kataja, J. (2009). Equalized algorithm for a truck cabin active noise control system. *Proceedings of Akustiikkapäivät*. Retrieved January 10, 2013, from http://www.akustinenseura.fi/wp-content/uploads/2013/08/44_Zou.pdf.