TWO CHANNEL ADAPTIVE SPEECH ENHANCEMENT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ERMAN ZAIM

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

FEBRUARY 2014

Approval of the thesis:

TWO CHANNEL ADAPTIVE SPEECH ENHANCEMENT

submitted by **ERMAN ZAIM** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Gönül Turhan Sayan
Head of Department, **Electrical and Electronics Engineering**

Prof. Dr. Tolga Çiloğlu
Supervisor, **Electrical and Electronics Engineering, METU**

Examining Committee Members:

Prof. Dr. Gözde Akar
Electrical and Electronics Engineering, METU

Prof. Dr. Tolga Çiloğlu
Electrical and Electronics Engineering, METU

Assoc. Prof. Dr. Cüneyt F. Bazlamaçcı
Electrical and Electronics Engineering, METU

Assoc. Prof. Dr. İlkay Ulusoy
Electrical and Electronics Engineering, METU

Tanın Afacan, M.Sc.
HBT ETD TYM, ASELSAN

Date: **7.2.2014**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: ERMAN ZAIM

Signature :

ABSTRACT

TWO CHANNEL ADAPTIVE SPEECH ENHANCEMENT

Zaim, Erman
M.S., Department of Electrical and Electronics Engineering
Supervisor : Prof. Dr. Tolga Çiloğlu

February 2014, 66 pages

In this thesis, speech enhancement problem is studied and a speech enhancement system is implemented on TMS320C5505 fixed point DSP. Speech degradation due to the signal leakage into the reference microphone and uncorrelated signals between microphones are studied. Limitations of fixed point implementations are examined. Theoretical complexities of weight adaptation algorithms are examined. Moreover, differences between theoretical and practical complexities of weight adaptation algorithms due to the selected DSP hardware are studied. Effects of the acoustic characteristics of recording environment on the performance of adaptive algorithms are examined. Computer simulations are performed on SAD source separation and Widrow's speech enhancement systems based on LMS, sign LMS and NLMS adaptive weight algorithms under both artificial and natural noises in order to compare their performances and decide filter length and step size selections. Speech enhancement systems based on LMS, SE-LMS and NLMS algorithms are implemented real time on TMS320C5505 fixed point DSP. Performances of these systems are evaluated by performing subjective listening tests. It is shown that implemented speech enhancement system works consistently and it increases the intelligibility of the speech transmitted to other party under various types of real noises.

Keywords: Speech Enhancement, Noise Cancellation, Fixed Point LMS

ÖZ

İKİ KANALLİ ADAPTİF KONUŞMA İYİLEŞTİRME

Zaim, Erman
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Tolga Çiloğlu

Şubat 2014 , 66 sayfa

Bu tezde, konuşma iyileştirme problemi çalışılmış ve konuşma iyileştirme sistemi TMS320C5505 sabit nokta sayısal sinyal işleyicisinde uygulanmıştır. Referans mikrofondaki sinyal kaçağı ve mikrofonlar arasındaki ilintisiz sinyaller nedeniyle oluşan konuşma bozulması çalışılmıştır. Sabit nokta uygulamalarının sınırlamaları incelenmiştir. Ağırlık adaptasyon algoritmalarının teorik karmaşıklığı incelenmiştir. Üstelik, ağırlık adaptasyon algoritmalarının teorik ve pratik karmaşıklıkları arasında seçilen DSP donanımı nedeniyle oluşan farklar incelenmiştir. Kayıt ortamının akustik özelliklerinin adaptif algoritmaların performansı üzerindeki etkileri incelenmiştir. LMS, sign LMS ve NLMS adaptif ağırlık algoritma tabanlı SAD kaynak ayırma ve Widrow'un konuşma geliştirme sistemlerinin performanslarını karşılaştırmak ve filtre uzunluğu ve adım boyutu seçimlerine karar vermek için hem yapay hem de gerçek sesler altında bilgisayar simülasyonları yapılmıştır. LMS, SE-LMS ve NLMS algoritmaları tabanlı konuşma geliştirme sistemleri TMS320C5505 sabit nokta sayısal sinyal işleyicisinde uygulanmıştır. Bu sistemlerin performansları kişisel dinleme testleri gerçekleştirilerek değerlendirilmiştir. Uygulanan konuşma geliştirme sisteminin sürekli çalıştığı ve çeşitli gerçek sesler altında diğer tarafa iletilen konuşmanın anlaşılabilirliğini arttırdığı gösterilmiştir.

Anahtar Kelimeler: Konuşma İyileştirme, Gürültü Giderici, Sabit Nokta LMS

To my family

ACKNOWLEDGMENTS

First of all, I would like to express sincere appreciation to my supervisor Prof. Dr. Tolga Çiloğlu for his excellent guidance and insight throughout this thesis. I am very grateful to my advisor for providing me the opportunity to do study in such an area in which I am very happy to research. I would like to specially thank to Burak Gürtunca for his help and advice during the establishments and research.

I would like to thank to TÜBİTAK for sharing their word list of Turkish Intelligibility Test.

My deepest thanks belong to my family, who were always by my side, for giving me the moral and for always believing in me.

I want to thank to ASELSAN Inc. and my employers for letting me involve in this academic study. I would like to express my special appreciation to my colleagues and seniors in HBT-ETD-TYM, for their contribution on the improvement of my engineering skills, which makes it possible to finish this thesis. In addition, I am very grateful for their patient audience to my complaining in the mornings, especially for the last months of my thesis.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

DSP	Digital Signal Processor
CPU	Central Processing Unit
I2C	Inter-Integrated Circuit
ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
RTC	Real Time Clock
PLL	Phase Locked Clock
I2S	Integrated-Interchip Sound
LMS	Least Mean Square
LS	Least Square
NLMS	Normalized Least Mean Square
SD-LMS	Sign Data Least Mean Square
SE-LMS	Sign Error Least Mean Square
SS-LMS	Sign Sign Least Mean Square
SAD	Symmetric Adaptive Decorrelation
DRE	Digital Residual Error
QE	Quantization Error
LSD	Least Significant Digit
MOS	Mean Opinion Score
DRT	Diagnostic Rhyme Test

CHAPTER 1

INTRODUCTION

Voice communication is usually performed in high noisy environments. Commercial telephone calls taking place at a cafeteria and military radios in armored vehicles, planes, helicopters can be given as examples. In such noisy environments, the intelligibility of speech transmitted to the other party decreases due to the additive background noise. In order to increase the quality and intelligibility of speech, the effect of noise on the speech signal should be reduced. For this reason, a system working as a noise-canceler is used in applications such as mobile phones [1], hands-free speaking [2], hearing aids, voice control devices [3], in-car speech.

One of the known approaches to purify speech signal from noise is based upon processing of signal coming through single channel [4]. Although single channel methods provide superiority in terms of ease of application and improve the quality of sensing, they do not contribute to intelligibility, in fact, most of the time they reduce it.

Another approach involves the use of multiple channels. Along with a microphone positioned near the mouth of speaker, other microphones are placed in other suitable places [5]. Signals picked by these microphones are used for noise reduction and such methods are known to be more effective than single channel noise reduction methods. The performance of multiple channel noise reduction algorithms is improved by increasing the number of microphones. However, computational complexity is also proportional to the number of used microphones. Due to this computational complexity, usage of two-channel systems are encouraged. In such a system one microphone is positioned close to the mouth, another microphone is positioned away from speech source. The former is called as primary whereas the latter is called as the secondary or reference microphone. Signal received by this reference microphone is used to eliminate the noise from the signal received by the primary microphone. Assuming that speech signal does not reach the reference microphone, the environmental sounds picked from the reference microphone should be modified to look like the signals reaching the primary one. Widrow's least square (LS) can be given as a well known example of this speech enhancement method [11]. Another succesful approach proposed for the solution of speech enhancement problem is to use source seperation techniques. Symmetric adaptive decorrelation (SAD) is one of the well-known source separation algorithm [8]. The performance of LS algorithm degrades mostly when the leakage from speech source into the reference source increases. SAD algorithm is suggested in order to enhance speech better when there exists leakage.

Speech enhancement algorithms are usually implemented with FIR filters. For the weight adaptation of FIR filters, there are many different types of adaptive filters. Least mean square (LMS) method is a well known and widely used adaptation algorithm [6]. It is usually preferred owing to its simplicity and its ease of application. Normalized LMS (NLMS) and sign LMS are modified LMS type weight adaptation algorithms. The performances of these adaptive algorithms usually measured with misadjustment, tracking capability and convergence speed.

Voice activity detection can be also used for speech enhancement purposes. In such applications, algorithm is adapted with a great step size parameter when there exists no speech but only background noise. Then, when speech begins, step-size of the adaptive filter is selected as a much lower value.

Field programmable gate arrays (FPGAs) and digital signal processors (DSPs) are used for adaptive filter implementations. However, DSPs are more commonly used. There are two types of digital signal processors, i.e. floating point digital signal processors and fixed point digital signal processors. Adaptive filters are implemented on both of these DSP types. Although the performance of floating point DSPs in arithmetic applications is higher and their ease of development is greater, their higher power consumption and higher prices leave them behind the fixed point digital signal processors in portable devices where power consumption has a great importance.

Therefore, mostly fixed point DSPs are preferred in industry. However, if neither power consumption nor money is a concern, then floating point digital signal processors can be used. In this thesis, adaptive filters are implemented in a fixed point DSP.

Implementation of adaptive filters on a fixed point hardware may suffer from negative impact called as slowdown phenomenon [9]. Because of this slowdown phenomenon, algorithm may stop before it goes to the optimal solution due to the dynamic range of least significant digit (LSD) of the fixed point digital signal processor. Precision also has an important effect on DSP implementation. Complexities of weight adaptation algorithms have a great importance in real time implementations since sources of DSP is limited. These factors are studied in this paper.

1.1 Purpose of This Study

The main motivation in this study is to design an effective two channel adaptive speech enhancement method on MATLAB simulation platform and then implement this method on the TMS320C5505 fixed point DSP. The theory of wiener filtering is examined and SAD and Widrow's LS method are studied. Moreover, speech enhancement algorithms are performed on MATLAB by using LMS, NLMS and sign LMS weight adaptation algorithms. These algorithms are compared according to their SNR values, computational complexity, filter sizes and step sizes. Fixed point solutions for these algorithms are studied and they are implemented on C5505 EZDSP USB STICK development kit of Texas Instruments. The performances of implemented filters are evaluated for their effects on speech intelligibility by subjective listening tests. Problems of the fixed point algorithms comparing to floating point are also studied.

1.2 Outline of This Study

In Chapter 2, theoretical background of Wiener filter and adaptive filters are given. Speech enhancement problem is examined in detail in Chapter 3. Hardware and software specifications are described in Chapter 4. In Chapter 5, effects of the fixed point

DSP selection on implemented adaptive algorithms compared to floating one and theoretical and practical complexities of adaptive algorithms are studied. In Chapter 6, MATLAB simulations of adaptive algorithms are given. In Chapter 7, subjective performance evaluation of implemented adaptive filters is given.

CHAPTER 2

BACKGROUND

In this chapter, theoretical background on Wiener filter, LMS, NLMS and sign modified LMS algorithms are given.

2.1 Wiener Filter

Wiener filter is a linear filter which would produce the optimum estimate of a signal in the MSE sense from a noisy measurement by using a noise free reference signal. The discrete form of the Wiener filtering problem is given in Figure 2.1.

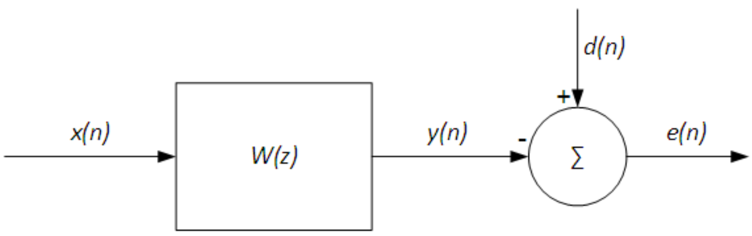

Figure 2.1: Block Diagram of Wiener Filter

x(n): input signal
d(n): desired signal
y(n): filter output signal
e(n): error signal

where

$$x(n) = d(n) + v(n) \tag{2.1}$$

However, Wiener filter requires some priori statistical information and some assumptions. It is assumed that $x(n)$ and $d(n)$ are jointly wide-sense stationary signals with known autocorrelation $R_x(k)$ and known cross-correlation $R_{dx}(k)$. It is also assumed that desired signal $d(n)$ and noise signal $v(n)$ are uncorrelated. The main purpose of Wiener filter is to estimate the desired signal $d(n)$ from noisy observations i.e. $x(n)$.

Assuming W(z) represents a transversal filter of length p, filter output $y(n)$ which is the convolution of $w(n)$ with $x(n)$ is given as

$$y(n) = \sum_{l=0}^{p-1} w_l(n)x(n-l) \tag{2.2}$$

Error signal is defined as

$$e(n) = d(n) - y(n) \tag{2.3}$$

Wiener filter is designed to filter the input signal $x(n)$ in order to produce the minimum mean square error (MSE) estimate, $y(n)$. The cost function of wiener filter is defined as

$$\xi = J(n) = E\{|e(n)|^2\} = E\{|d(n) - y(n)|^2\} \tag{2.4}$$

In order to minimize ξ, it is necessary and sufficient that derivative of ξ with respect to w_k is equal to zero for $k = 0, 1, ..., p-1$

$$\frac{\partial \xi}{\partial w_k} = E\{e(n)\frac{\partial e(n)}{\partial w_k}\} = 0 \tag{2.5}$$

Since $d(n)$ is independent of w_k,

$$\frac{\partial e(n)}{\partial w_k} = -\frac{\partial y(n)}{\partial w_k} = x(n-k) \tag{2.6}$$

Then, from (2.5) and from (2.6) error and input should be orthogonal

$$E\{e(n)x(n-k)\} = 0; \qquad k = 0, 1, ..., p-1 \tag{2.7}$$

This equation is known as orthogonality principle. Substituting (2.3) into (2.7)

$$E\{d(n)x(n-k)\} - \sum_{l=0}^{p-1} w_l E\{x(n-l)x(n-k)\} = 0 \tag{2.8}$$

Since $x(n)$ and $d(n)$ are jointly WSS then,

$$E\{x(n-l)x(n-k)\} = R_x(k-l) \tag{2.9}$$

$$E\{d(n)x(n-k)\} = R_{dx}(k) \tag{2.10}$$

Substituting (2.9) and (2.10) into (2.8)

$$\sum_{l=0}^{p-1} w_l R_x(k-l) = R_{dx}(k); \qquad k = 0, 1, ..., p-1 \tag{2.11}$$

is held. Matrix representation of (2.11) is shown as

$$R_x w = R_{dx} \tag{2.12}$$

where R_x is a Hermitian Toeplitz matrix of autocorrelations of input signal $x(n)$, w is the vector of filter coefficients solution to the Wiener-Hopf equations, and R_{dx} is the cross-correlation between desired signal $d(n)$ and the input signal $x(n)$.

2.2 Least Mean Square

The Wiener-Hopf equation given in the previous section shows that optimum filter coeeficients of Wiener filter can be obtained if some statistics about signals are known and signals are stationary. However, in practical applications, these true statistics are not known. Moreover, autocorrelation and cross correlation terms vary with time. Therefore, in real time applications instead of these true statistics, some estimated values are needed to be used. In addition, in order to deal with nonstationary signals, filter coefficients should be time varying. In this part, one of the most commonly used and well known adaptation method, LMS algorithm [6] will be investigated. LMS algorithm is a stochastic gradient approximation that is widely used. In LMS, the weight vector adaptation is given as

$$w(n+1) = w(n) + \mu E\{e(n)x(n)\} \tag{2.13}$$

As mentioned above, in practical applications, this $E\{e(n)x(n)\}$ term is generally unknown. Therefore, in LMS, instead of minimizing the square of error signal $E\{e^2(n)\}$, $e^2(n)$ is minimized. Hence, cost function is selected as

$$\xi = J(n) = e^2(n) \tag{2.14}$$

Then, weight vector update in equation (2.13) changes for LMS as

$$w(n+1) = w(n) + \mu e(n)x(n) \tag{2.15}$$

Like the wiener filter (2.3), error signal in LMS is defined as

$$e(n) = d(n) - y(n) \tag{2.16}$$

Where $y(n)$ is convolution of filter coefficients with the input signal, $x(n)$

$$y(n) = w(n)^T x(n) \tag{2.17}$$

LMS algorithm converges in mean if [11]

$$0 < \mu < 2/\lambda_{max} \tag{2.18}$$

Where λ_{max} corresponds to the largest eigenvalue of input correlation matrix R_x. λ_{max} can be approximated by the trace of input autocorrelation matrix. Therefore, bound in (2.18) can be narrowed as

$$0 < \mu < 2/tr\{R_x\} \tag{2.19}$$

2.3 Normalized Least Mean Square

In normalized LMS, the gradient step size is normalized by the energy of data vector. NLMS weight update function is defined as

$$w(n+1) = w(n) + \beta \frac{x(n)}{\|x(n)\|^2} e(n) \tag{2.20}$$

Where β is normalized LMS step size with

$$0 < \beta < 2 \tag{2.21}$$

for the convergence.

In the LMS algorithm, weight update is proportional to input vector $x(n)$. When $x(n)$ is large, LMS algorithm faces with a problem called as gradient noise amplification. In NLMS, by dividing weight update by $||x(n)||^2$, this noise amplification problem is solved.

However, now NLMS has an opposite problem that is, when $x(n)$ is too small, division by a very small number problem might be seen. Therefore, (2.20) is modified as

$$w(n+1) = w(n) + \beta \frac{x(n)}{c + ||x(n)||^2} e(n) \tag{2.22}$$

Where c is a very small value added in order to avoid from the possibility of zero division when input $x(n)$ has a very small value.

2.4 Sign Least Mean Square

Sign modified LMS algorithms are proposed in order to reduce the computational complexity of LMS. Sign-LMS algorithm replaces the multiplication of error signal $e(n)$ and input signal $x(n)$ in weight update algorithm (2.15) with the sign operator to either error $e(n)$, input $x(n)$ or both error and input. Weight update equation for sign-error algorithm becomes

$$w(n+1) = w(n) + \mu sgn\{e(n)\}x(n) \tag{2.23}$$

where

$$\begin{aligned} & 1 \quad if\ e(n) > 0 \\ sgn\{e(n)\} = & 0 \quad if\ e(n) = 0 \\ & -1 \quad if\ e(n) < 0 \end{aligned} \tag{2.24}$$

Weight update equation for sign-input algorithm is defined as

$$w(n+1) = w(n) + \mu e(n) sgn\{x(n)\} \tag{2.25}$$

and weight update equation for sign-sign algorithm becomes

$$w(n+1) = w(n) + \mu sgn\{e(n)\} sgn\{x(n)\} \tag{2.26}$$

CHAPTER 3

SPEECH ENHANCEMENT PROBLEM

In this section, speech enhancement problem and two models for the solution of this problem, symmetric adaptive decorrelation (SAD) and Widrow's least square (LS) will be presented. Moreover, difficulties of speech enhancement such as signal leakage into the reference microphone, minimum phase problem, uncorrelated signals between microphones and reverberation will be examined.

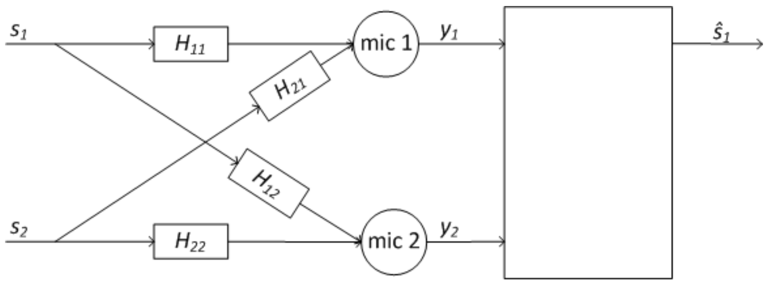

Figure 3.1: Two Channel Speech Enhancement

H_{11} : $acoustic\ transfer\ path\ of\ speech\ signal\ on\ primary\ microphone$
H_{12} : $acoustic\ transfer\ path\ of\ speech\ signal\ on\ secondary\ microphone$
H_{21} : $acoustic\ transfer\ path\ of\ noise\ signal\ on\ primary\ microphone$
H_{22} : $acoustic\ transfer\ path\ of\ noise\ signal\ on\ secondary\ microphone$

The general picture of the speech enhancement for two channel case is given in Figure 3.1. In the first part of Figure 3.1, transfer function of each channel $H_{11}(z)$, $H_{22}(z)$ and cross coupling effects $H_{12}(z)$, $H_{21}(z)$ are shown. The main purpose is to put an algorithm into the blank box in order to get a noise free output signal $\hat{s}_1$ which is close to the speech signal s_1 by using input signals y_1 and y_2, picked up from microphones.

In this chapter, signal separation by decorrelation and Widrow's least square speech enhancement methods are examined.

3.1 Widrow's Least Square Model [6]

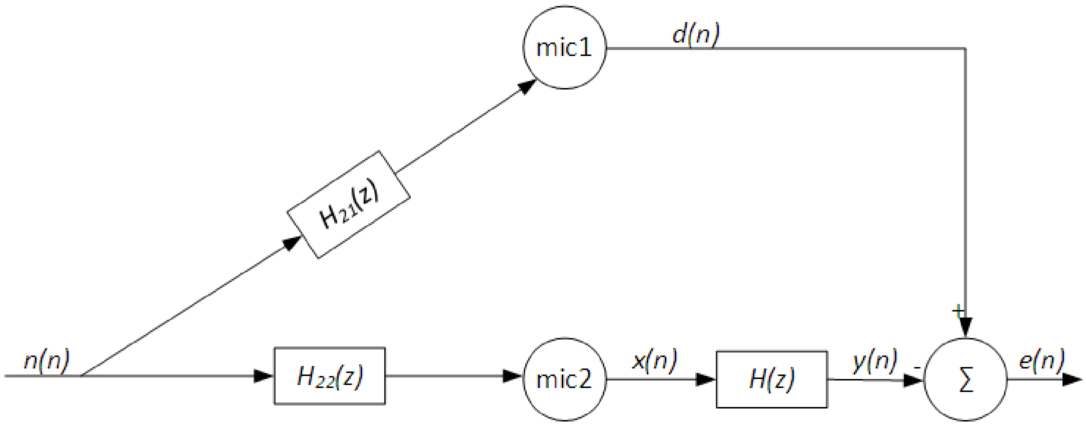

Figure 3.2: Least Square with Noise Alone

In Figure 3.2, least square noise cancellation when speech source is not available is shown. In this simplified case, transfer function representing acoustic path between two microphones is

$$H(z) = H_{22}^{-1}(z)H_{21}(z) \tag{3.1}$$

However, inverse matrix problem exists in this very simple case. In other words, if $H_{22}(z)$ is nonminimum phase, $H(z)$ will not be available.

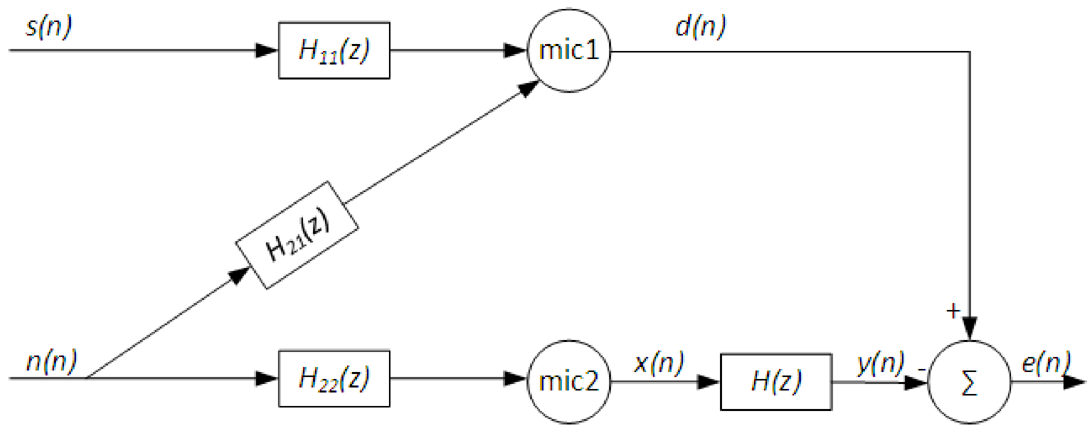

Figure 3.3: Widrow's Least Square Speech Enhancement

In Figure 3.3, Widrow's typical least square speech enhancement model is shown. Signal leakage from speech source into the reference microphone is assumed to be equal to zero that is

$$H_{12}(z) = 0\ ,\ \forall z \tag{3.2}$$

In practical applications, microphones are positioned in such a way that reference microphone picks up noise signal only while primary microphone picks up both noise and speech signals [11]. Moreover, it is also assumed that H_{11} is unitary transform. If such a perfect condition is met i.e secondary microphone picks up noise only, $H(z) = H_{22}^{-1}(z)H_{21}(z)$ still holds matrix inversion problem. While setting up microphones for practical applications in order to minimize speech leakage in the reference microphone, one may think to have a distance between two microphones by putting the first microphone next to the mouth whereas putting the second one away from it. However, this might lead to an increase in the distance between microphones.

On the other hand, the application of small separation between two microphones may give favorable effects that significantly reduce filter length required for noise cancellation and minimize the presence of reverberation [12]. Moreover, coherence of noise between two microphones is also very important for noise cancellation. If the distance between two microphones is decreased, the coherence will be increased, however, signal leakage from speech source into the reference microphone will be also increased.

In Figure 3.2, a case when speech signal does not exist and only noise signal is available is given. In this case desired signal becomes

$$d(n) = H_{21}(z)\{n(n)\} \tag{3.3}$$

input signal becomes

$$x(n) = H_{22}(z)\{n(n)\} \tag{3.4}$$

Then, error signal can be measured as

$$e(n) = d(n) - y(n) = d(n) - H(z)\{x(n)\} \tag{3.5}$$

If we put (3.3) and (3.4) into (3.5), error signal becomes

$$e(n) = \{H_{21}(z) - H(z)H_{22}(z)\}n(n) \tag{3.6}$$

and noise is cancelled if $H(z) = H_{21}(z)H_{22}^{-1}(z)$ as it was estimated before.

If speech is also available, like given in Figure 3.3, then desired signal becomes

$$d(n) = H_{21}(z)\{n(n)\} + H_{11}(z)\{s(n)\} \tag{3.7}$$

and input signal becomes

$$x(n) = H_{22}(z)\{n(n)\} + H_{12}(z)\{s(n)\} \tag{3.8}$$

Then, error signal can be measured as

$$e(n) = d(n) - y(n) = d(n) - H(z)\{x(n)\} \tag{3.9}$$

In this case, because of the leakage from speech signal into the reference microphone which is symbolized by H_{12} in Figure 3.1, $H(z) = H_{21}(z)H_{22}^{-1}(z)$ cannot be found by Widrow's least square method although if it is assumed that $H_{22}(z)$ is minimum phased.

Voice activity detection (VAD) is a method that is widely used in order to cope up with this situation. To explain better, when there is no speech activity, the adaptive filter works and sets some filter coefficients. When speech begins, the adaptation stops or continues with a smaller step-size. In that way, $H(z) = H_{21}(z)H_{22}^{-1}(z)$ is assumed to be found correctly. However, since acoustic paths are not fixed that is signals are nonstationary, for the exposure of long speech signals, this system does not work well.

If it is assumed that $H(z) = H_{21}(z)H_{22}^{-1}(z)$ is found with the help of VAD. If we put this $H(z)$ into (3.9), error signal becomes

$$e(n) = \{H_{11}(z) - H_{21}(z)H_{22}^{-1}(z)H_{12}(z)\}s(n) \tag{3.10}$$

In this equation, it is clear that noise can be completely cancelled however there is a distortion over the speech signal. This distortion will be decreased if $H_{11}(z)$ and $H_{12}(z)$ are unitary transforms.

One of the suggested methods to solve this problem is to place microphones in such a position that speech is delivered in equal distance to both microphones[15]. If this distance is kept small enough by placing microphones close enough to the mouth, making an assumption such as $H_{11} = H_{12} = 1$ will be meaningful. In that case, it is assumed that H_{21} and H_{22} are not equal to each other. This means that noise is never be directly in front of or behind the microphones. However, in practical applications $H_{21}(z) = H_{22}(z)$ is very likely to occur. Therefore, this is not a practical method.

3.2 Symmetric Adaptive Decorrelation Model

In [7], a signal separation model and its modification for speech enhancement are presented. In this approach, unlike Widrow's least square model, signal leakage from speech source into the reference microphone is also considered. Feedback symmetric adaptive decorrelation approach is given in Figure 3.4. For the simplicity, it is assumed that H_{11} and H_{22} are unitary transformations.

The observed signals $y_1(t)$ and $y_2(t)$ are the outputs of a 2 x 2 LTI system with inputs

$s_1(t)$ and $s_2(t)$ where frequency response $H(w)$ is

$$H(w) = \begin{bmatrix} 1 & H_{12}(w) \\ H_{21}(w) & 1 \end{bmatrix} \tag{3.11}$$

It is also assumed that

$$1 - H_{12}(w)H_{21}(w) \neq 0 \quad , \quad \forall w \tag{3.12}$$

If this assumption is not satisfied then H(w) will not be invertible and input signals can not be recovered. In [7], it is also assumed that input signals $s_1(t)$ and $s_2(t)$ are statistically uncorrelated WSS random processes with zero mean.

$$E\{s_1(t)s_2^T(t-\tau)\} = 0 \quad , \quad \forall \tau \tag{3.13}$$

If H_{12} and H_{21} were known, input signals can be recovered. However, in practical applications, these transformations are not known. In [7], the main purpose is to find the estimates $\hat{H}_{12}$ and $\hat{H}_{21}$ of H_{12} and H_{21} in order to find statistically uncorrelated estimated input signals by inverse filtering as shown in Figure 3.4.

$$E\{\hat{s}_1(t)\hat{s}_2^T(t-\tau)\} = 0 \quad , \quad \forall \tau \tag{3.14}$$

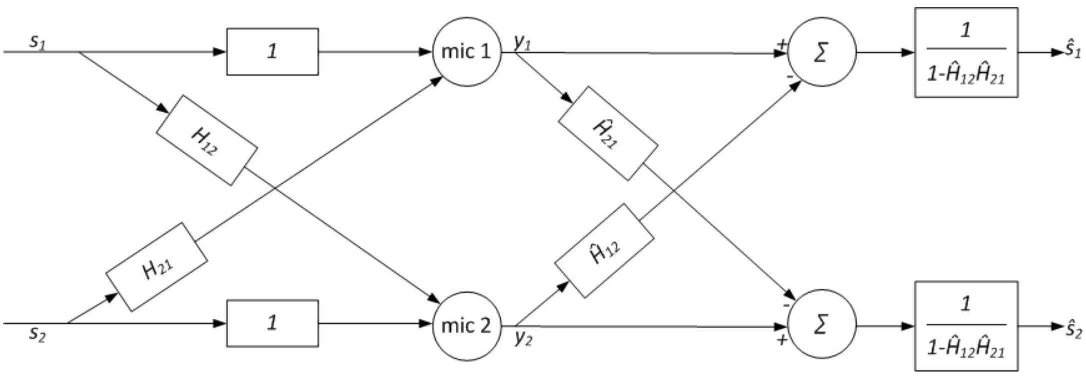

Figure 3.4: Symmetric Adaptive Decorrelation

Frequency response between input signals $y_1(t)$ and $y_2(t)$ and output signals $\hat{s}_1(t)$ and $\hat{s}_2(t)$ is given as

$$\hat{H}^{-1}(w) = \frac{1}{1 - \hat{H}_{12}(w)\hat{H}_{21}(w)} \begin{bmatrix} 1 & -\hat{H}_{12}(w) \\ -\hat{H}_{21}(w) & 1 \end{bmatrix} \tag{3.15}$$

where it is assumed that $\hat{H}(w)$ is invertible

$$1 - \hat{H}_{12}(w)\hat{H}_{21}(w) \neq 0 \quad , \quad \forall w \tag{3.16}$$

By using the power spectra between inputs and outputs of LTI system

$$\begin{aligned}\begin{bmatrix} P_{\hat{s}_1\hat{s}_1}(w) & P_{\hat{s}_1\hat{s}_2}(w) \\ P_{\hat{s}_2\hat{s}_1}(w) & P_{\hat{s}_2\hat{s}_2}(w) \end{bmatrix} = \frac{1}{|1-\hat{H}_{12}(w)\hat{H}_{21}(w)|^2} \begin{bmatrix} 1 & -\hat{H}_{12}(w) \\ -\hat{H}_{21}(w) & 1 \end{bmatrix} \\ \begin{bmatrix} P_{\hat{y}_1\hat{y}_1}(w) & P_{\hat{y}_1\hat{y}_2}(w) \\ P_{\hat{y}_2\hat{y}_1}(w) & P_{\hat{y}_2\hat{y}_2}(w) \end{bmatrix} \begin{bmatrix} 1 & -\hat{H}_{21}^T(w) \\ -\hat{H}_{12}^T(w) & 1 \end{bmatrix}\end{aligned} \tag{3.17}$$

By using the decorrelation condition in (3.14) and (3.17), (3.18) is hold.

$$P_{y_1y_2}(w) - \hat{H}_{12}(w)P_{y_2y_2}(w) - \hat{H}_{21}^T(w)P_{y_1y_1}(w) + \hat{H}_{12}(w)\hat{H}_{21}^T(w)P_{y_2y_1}(w) = 0 \tag{3.18}$$

Then $\hat{H}_{12}$ becomes

$$\hat{H}_{12}(w) = \frac{P_{y_1y_2}(w) - \hat{H}_{21}^T(w)P_{y1y1}(w)}{P_{y2y2}(w) - \hat{H}_{21}^T(w)P_{y2y1}(w)} \tag{3.19}$$

It is clear that this equation does not hold a unique solution. If cross coupling from speech source into the reference microphone $\hat{H}_{21}$ is chosen as 0, (3.19) becomes

$$\hat{H}_{12}(w) = \frac{P_{y_1y_2}(w)}{P_{y_2y_2}(w)} \tag{3.20}$$

This solution is exactly equal to Widrow's LMS solution [6]. This solution is one of many solutions of the decorrelation equation. Although this solution gives sufficient results in many situations, its performance decreases when zero coupling assumption is not satisfied. In order to measure performance, the ratio of power spectrum of the desired signal to the power spectrum of the interference signal is used. Signal estimates $\hat{s}_1(t)$ and $\hat{s}_2(t)$ are the outputs of the 2 x 2 system with inputs $\hat{s}_1(t)$ and $\hat{s}_2(t)$

and frequency response

$$H(w)\hat{H}^{-1}(w) = \frac{1}{1-\hat{H}_{12}(w)\hat{H}_{21}(w)}\begin{bmatrix} 1-\hat{H}_{12}(w)H_{21}(w) & \hat{H}_{12}(w)-H_{12}(w) \\ H_{21}(w)-\hat{H}_{21}(w) & 1-H_{12}(w)\hat{H}_{21}(w) \end{bmatrix} \tag{3.21}$$

Signal to interference ratio in the first sensor is

$$S/I = \frac{|1-\hat{H}_{12}(w)H_{21}(w)|^2 P_{s_1 s_1}(w)}{|\hat{H}_{12}(w)-H_{12}(w)|^2 P_{s_2 s_2}(w)} \tag{3.22}$$

Where the interference is the signal component involving $\hat{s}_2(t)$. In Widrow's method, signal to interference ratio is

$$S/I = \frac{1}{|H_{21}(w)|^2}\frac{P_{s_2 s_2}(w)}{P_{s_1 s_1}(w)} \tag{3.23}$$

In LS method, it is assumed that H_{12} is equal to zero. Therefore, no processing is applied to the reference signal. Hence, signal to interference ratio is limited by the interference to signal ratio at the reference microphone. However, in the decorrelation approach, higher S/I ratios can be hold by finding close estimate of H_{12}. As mentioned before, H_{11} is assumed to be unitary transform. It can not be distinguished by using neither decorrelation nor least square approach, however, it can be made close to unity by placing microphones appropriately.

In Figure 3.5, feedforward SAD algorithm is shown. LMS, NLMS and sign LMS algorithms mentioned in Chapter 2 can be used for the weight adaptation of SAD. NLMS based SAD will be examined here.

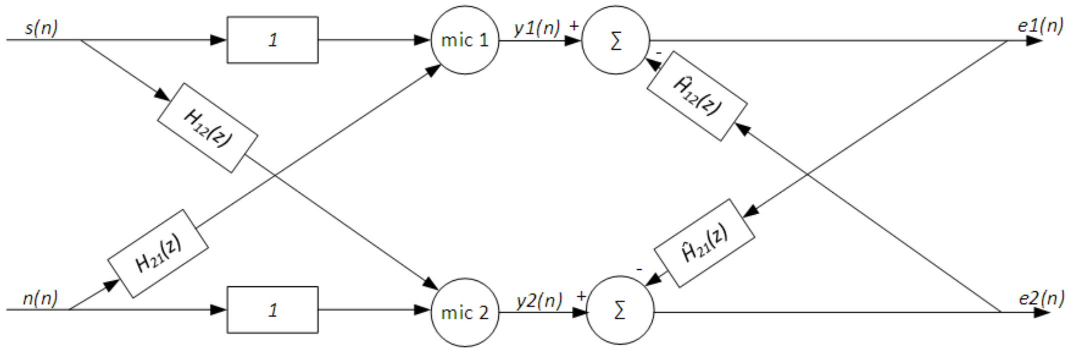

Figure 3.5: Symmetic Adaptive Decorrelation

NLMS weight update equation for SAD becomes

$$\begin{aligned} w1(n+1) &= w1(n) + \frac{\mu 1}{c + \|e2(n)\|^2} e1(n)e2(n) \\ w2(n+1) &= w2(n) + \frac{\mu 2}{c + \|e1(n)\|^2} e2(n)e1(n) \end{aligned} \tag{3.24}$$

where

$$\begin{aligned} 0 < \mu 1 < 2 \\ 0 < \mu 2 < 2 \end{aligned} \tag{3.25}$$

and error signals are defined as

$$\begin{aligned} e1(n) &= y1(n) - e2(n)w1(n) \\ e2(n) &= y2(n) - e2(n)w2(n) \end{aligned} \tag{3.26}$$

CHAPTER 4

HARDWARE COMPONENTS AND SOFTWARE SPECIFICATIONS

As a hardware development tool, C5505 EZDSP USB STICK development kit of Texas Instruments is used. Two main hardwares on this platform are TMS320C5505 low power fixed point DSP and TLV320AIC3204 stereo audio codec. Code Composer Studio V5.4.0 is used as a code compiler tool in order to develop the software.

The configuration of TMS320C5505 DSP is done by setting its central processing unit (CPU) registers and peripheral registers. For instance, CPU clock is set by CPU registers whereas Inter-Integrated Circuit (I2C) [22] and Integrated Interchip Sound (I2S) [20] protocols used between DSP and codec are set by peripheral registers. The detailed description of TMS320C5505 DSP, its registers and settings are given in Appendix A.

Similarly, all configurations of TLV320AIC3204 audio codec are set through its registers [21]. The control data is transferred between DSP and audio codec through I2C serial interface in order to set the registers of the audio codec. With this control interface, sampling rate, clock rates, ADC, DAC gains, microphone and speaker gains, input and output line switching, I2S data format, etc. are set. The audio data is transferred between DSP and audio codec via I2S bus protocol. The detailed description of TLV320AIC3204 audio codec, its registers and settings are given in Appendix B.

The sampling rate of the audio codec is configured as 8 kHz. If it is chosen higher, time that algorithm spends will be also proportionally higher. Since the primary consideration in this thesis is human speech signal which is usually below 4 kHz, 8 kHz sampling rate is chosen. In addition, resolution of the codec is chosen as 16 bit since higher resolutions will increase the computational complexity. The codec has two ADCs and two DACs. ADCs are used in order to convert analog signals coming from selected input line into digital ones. Similarly, DACs are used to sample digital signals into analog ones, and then these signals are passed through selected line outputs of codec.

The input RTC oscillator crystal clock in DSP is 32.768 kHz. With the help of a PLL unit, this frequency is increased to 120 MHz. In other words, one clock cycle is 0.0083 microseconds (1/120 MHz). As mentioned before, the sampling rate of the audio codec is configured as 8 kHz. In other words, new audio data will come in every 0.125 milliseconds. Theoretically, maximum 125/0.0083 = 15060 clock cycle length algorithm can be run on DSP with these configurations. However, there are other tradeoffs such as time spent in the entry and exit of interrupts.

For reading data from codec and writing data to codec, two interrupts or one single interrupt can be used. When two interrupts are used, one of them is used for reading data from audio codec i.e. RINT and other one is used to write data to audio codec i.e. XINT. In such implementations, timing management within the interrupt subroutines will be hard. Therefore, one must be very careful about time spent in RINT and XINT interrupts. Moreover, time spent in the entry and exit of interrupts will also increase. However, it has an advantage of dividing the work load in the interrupt subroutine. It is not preffered to have long time consuming algorithm in single interrupt in multi task projects. However, in this project since only one task, adaptive filtering, is run, a single interrupt i.e. RINT is used.

Two different inputs are needed for adaptive algorithms used in this thesis. Hence, both ADCs of audio codec are used. Basically, in every 0.125 milliseconds two different analog inputs (noisy signal and noise reference signal) are sampled into 16 bit signed digital signals on the ADCs. These digitalized signals are given to DSP through I2S bus protocol with an I2S software interrupt. In this interrupt, adaptive algorithm is performed with these two signals. Then, adaptive filter output and noisy input signals are given back to the audio codec through I2S. These 16 bit signed digital signals are sampled to analog ones on different DACs and they are passed through separate audio lines to the speaker jack of the board. Finally, these signals can be played through speakers connected to this jack. The block diagram of the physical relation between the ADC and DAC of audio codec and DSP is given in 4.1.

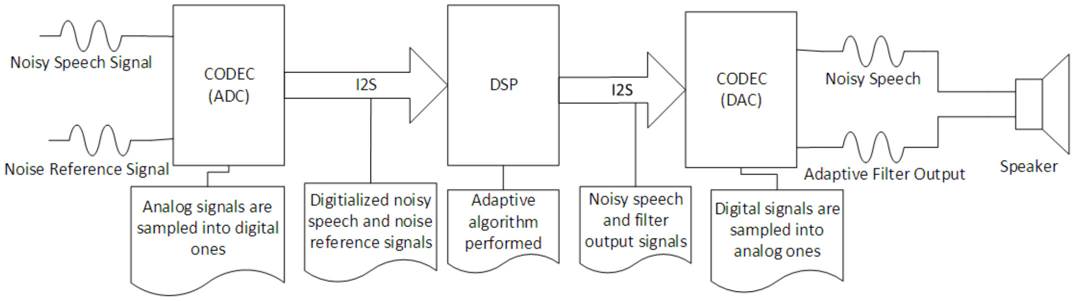

Figure 4.1: Physical Interfaces Between Codec and DSP

The general software block diagram of adaptive algorithms implemented on DSP is given in Figure 4.2.

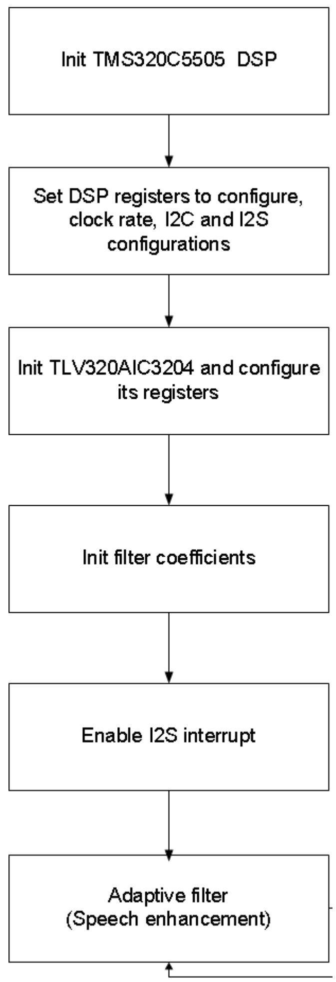

Figure 4.2: Software Block Diagram of Adaptive Filters on DSP

Firstly, DSP is initialized and its CPU and peripheral registers are set in order to configure clock rate, I2C, I2S and interrupts. Then, codec is initialized and its configurations such as sampling rate, ADC and DAC gains, selected lines for microphones and speakers, etc are done. After that, coefficients of adaptive filter are cleared. Finally, I2S interrupt is enabled and speech enhancement algorithm begins running in interrupt subroutine.

As mentioned before, in every 0.125 milliseconds, interrupt subroutine is called. Every entry and exit of this interrupt also spends some time, i.e. clock cycles. In order to decrease the time spent in these entries and exits, the feature of I2S protocol which is called as packed mode is enabled. With this mode, instead of every 0.125 milliseconds, in every 0.250 milliseconds the interrupt subroutine is called with double number of data. With the help of this feature, time spent in the entry and exit of interrupt subroutine is decreased. To explain better, instead of running adaptive algorithm once in every 0.125 milliseconds, it is called twice in every 0.250 milliseconds. In other words, instead of entering the interrupt subroutine twice, it is entered once in every 0.250 milliseconds.

TMS320C5505 DSP has four 40 bit accumulators [23]. Accumulators consist of 16 high order bits, 16 low order bits and 8 guard bits. These guard bits are used in order to prevent overflow in multiplications. In this study, adaptive filters are completely implemented in assembly because of the high computational needs. LMS assembly command of TMS320C5505 is used in order to perform LMS based weight adaptations [19]. The usage of accumulators, precision and complexities of weight adaptation algorithms will be examined in detail in the next chapter.

CHAPTER 5

FIXED POINT LIMITATIONS AND ALGORITHM COMPLEXITIES

In this chapter, theoretical and practical complexities of LMS, NLMS and sign LMS algorithms are examined. Digital errors, slowdown phenomenon and precision are studied.

5.1 Complexities of Adaptive Algorithms

In Table 5.1, theoretical complexities of sign error LMS, LMS and NLMS weight adaptation algorithms are given in terms of additions, multiplications and divisions. However, these complexities depend on the implemented hardware environment. Therefore, these theoretical complexities changes in practical applications. As mentioned before, in this study, TMS320C5505DSP is used and effect of this DSP on computational complexities are discussed in this section.

Table5.1: Theoretical Complexities of Adaptive Weight Algorithms

Operation Type / Algorithm Type	Multiplication	Addition	Division	Sign
LMS	2N+1	2N		
SE-LMS	2N	2N		1
NLMS	3N+1	3N	1	

N : length of filter

The transversal FIR filter structure is used in implementations. This type of filter structures require the use of a delay line of input samples x(n). The samples x(n), x(n-1), ..., x(n-N+1) are needed to be stored in data memory. They are stored in a circular buffer in reverse order. Similarly, filter weight coefficients $w_0, w_1, \ldots, w_{N-1}$ are stored in a circular data memory buffer in forward order. Since TI 55x DSP structure can access two different data memory address in one cycle, there is no need to store one of these variables in program memory. By using two inputs, the input sample x(n) and filter weight coefficients w(n), filter output y(n) is calculated (2.17).

$$y(n) = w(n)^T x(n) \tag{5.1}$$

Then, filter taps are updated according to the equation

$$w_i(n+1) = w_i(n) + weightupdate \tag{5.2}$$

(5.1) and (5.2) are same for all adaptive filters mentioned in this study. In TMS320C5-505, these two operations (5.1) and (5.2) are performed in one clock cycle by using LMS instruction. The main differences between complexities of algorithms are due to the measurement of the weight update equation.

Let's remember the weight update equation of LMS algorithm (2.15)

$$w(n+1) = w(n) + \mu e(n)x(n) \tag{5.3}$$

Multiplication of input signal $x(n)$ with the error signal $e(n)$ is measured in one clock cycle. One more clock cycle is needed to multiply the result with step size, μ. Another clock cycle is needed in order to save filter coefficients. Therefore, 4N clock cycle work load is performed in LMS loop.

The weight update equation of NLMS algorithm is given in (2.22)

$$w(n+1) = w(n) + \beta \frac{x(n)}{c + ||x(n)||^2} e(n) \tag{5.4}$$

The division part in this weight adaptation algorithm (5.4) has a great effect on both computational complexity and the precision. It is a general rule in fixed point environment to do multiplication first and then do division in order to have higher precision.

Since C5505 DSP does not have any hardware for division, divisions are performed by using subtraction and it requires high computational load.

If $x(n)/\|x(n)\|^2$ is firstly done, then precision will be lost since the result will be zero for the small values of $x(n)$. The best solution is to perform multiplication first i.e. $x(n) * e(n)$, then do division. This is a general rule for division in fixed point arithmetic in order to increase the precision. Twenty more clock cycles are needed in order to do division. Therefore, 22N clock cycle work load is performed in NLMS loop.

Sign LMS algorithms (SE-LMS, SD-LMS, SS-LMS) are presented in order to reduce the computational complexities of LMS algorithm by decreasing the number of multiplications. However, this approach does not have any computational benefits for DSP implementations because sign of the error calculation does not have computational advantage over classical LMS algorithm. In fact, sometimes it might increase the computational load. The weight update equation of SE-LMS algorithm is given in (2.23)

$$w(n+1) = w(n) + \mu sgn\{e(n)\}x(n) \tag{5.5}$$

For the sign checking, six clock cycles are needed. In SE-LMS algorithm sign of error is calculated only once outside the loop. Therefore, similar to LMS algorithm, 4N clock cycle work load is performed in SE-LMS loop. However, in SD-LMS algorithm, sign of error is calculated inside the loop. Therefore, complexity of SD-LMS increases compared to LMS. Six more clock cycles are needed in order to do sign checking. Therefore, 10N clock cycle work load is performed in SD-LMS loop.

The complexities of these weight adaptation methods linearly depend on filter length. Complexity of Widrow's LS method is half of that of SAD algorithm for every case since LS method updates one channel where SAD method updates two channels.

5.2 Slowdown Phenomenon and Digital Errors

Weight update equation for LMS is given in [6]

$$w(k+1) = w(k) + 2\mu e(k)x(k) \tag{5.6}$$

Magnitude of adaptation term in (5.6) can never be smaller than LSD. When $2\mu e[k]x[k]$ is smaller than LSD, adaptation of the filter stops for that coefficient.

$$|2\mu e_{k_0} x_{k_0} - p| < LSD \tag{5.7}$$

This phenomenon in (5.7) is called as digital termination [10]. Rms value of $|x_{k_0}|$ can be written as

$$X_{rms} = \sigma_x^2 + m_x^2 \tag{5.8}$$

As an approximation in (5.7), $|x_{k_0}|$ is replaced with its rms value given in (5.8). Then, (5.7) becomes

$$|e_{k_0}| < \frac{LSD}{2\mu X_{rms}} \triangleq e_d(\mu) \tag{5.9}$$

where $e_d(\mu)$ is defined as digital residual error (DRE). (5.9) clearly shows than DRE is inversely proportional to step size. On the contrary, for floating point i.e high precision case, error is reduced by decreasing step size. If the stopping phenomenon does not exist and only source of error is the quantization of coefficients, quantization error QE can be defined as

$$QE = E\{(y_k - \hat{y}_k)^2\} \tag{5.10}$$

From (5.10), it is clear that QE decreases when the filter length gets larger whereas DRE is decreasing with the increasing step size values. In the floating point case, decreasing step size minimizes the mean square error. Therefore, while chosing time varying step sizes, it is reasonable to select an algorithm which decreases step size within time in floating point case. For the fixed point case, DRE is inversely proportional to μ whereas steady state error increases with μ. Therefore, this optimal strategy for the floating point case is not suitable for the fixed point case. In [13], it is shown that this phenomenon does not stop adaptation as previously believed, but instead severely reduces the convergence rate. Thus, instead of stopping phenomenon, slowdown phenomenon exists. [9] examines slowdown phenomenon in detail. It is shown that steady state MSE is determined mostly by the data bits length and for smaller values of step size, effect of the slowdown phenomenon can be decreased by using more coefficients bits [9].

5.3 Fixed Point Precision

Multiplication of 16 bit signed integers is realized in all of the implemented filters. In DSP 40 bit accumulators are used in order to this multiplication. 16 high order bits are selected for the result in order to prevent overflow. However, underflow condition might occur in this case. To explain better, if two small 16 bit signed integers are multiplied, the result will be in the low order 16 bits of the accumulator and high order bits will be zero. When these high order bits are picked for the result, underflow will occur which is not a case in the high precision floating point simulations. Therefore, performance of the adaptive filters in fixed point case will depend on these selections.

Let's remember the error equation defined in (2.3)

$$e(n) = d(n) - y(n) \tag{5.11}$$

In order to get this error signal in fixed point implementations, 16 bit filter output signal $y(n)$ is extracted from 16 bit desired signal. In order to prevent the overflow, this process performed on the 40 bit accumulator and the result is shifted to the right one bit (5.12) and low order 16 bits are selected for result.

$$\hat{e}(n) = e(n) >> 1 \tag{5.12}$$

Because of this one bit shift called as s_e, it is expected to have an output signal having smaller amplitude compared to desired signal.

Now, let's remember the filter update equation of the LMS algorithm given in (2.15)

$$w(n+1) = w(n) + \mu e(n)x(n) \tag{5.13}$$

First of all, in order to decrease the clock cycle used by the algorithms, μ is always selected as powers of 2. By selecting the μ as powers of 2, the multiplications can be performed as right shifts which is less complex in the practical implementations. The multiplication of error signal and input signal is performed on a 40 bit accumulator. Now, let's remember the filter convolution equation of LMS algorithm given in (5.14)

$$y(n) = w(n)^T x(n) \tag{5.14}$$

These multiplication and addition series are also performed on a 40 bit accumulator. Then, high order 16 bits of the accumulator are taken in order to prevent overflow and get more precision. If there were not any guard bits, this scaling should be performed before the multiplication and the precision will be lost. Thanks to 8 guard bits of the accumulator, higher precision can be achieved. When DSP settings for overflow is set by SATD, then number in the accumulator will be set to maximum positive number if the result is positive and it will be equal to minimum negative number if the result is negative in the occurrence of overflow. To explain better, with additions of positive numbers, negative numbers will not be held as a result. Bit length equation of convolution of filter coefficients with input can be given as

$$b_y = log_2(p) + b_x + b_w \tag{5.15}$$

where

b_w : $bit\ length\ of\ the\ input$
b_w : $bit\ length\ of\ the\ filter\ coefficient$
b_y : $bit\ length\ of\ the\ output$
p : $filter\ length$

Therefore, it is safe to say that no overflow will occur although a filter with 256 taps is selected in 40 bit accumulators. In addition, when shorter filter length is selected it will be more precise to select lower bits. In other words, it will cause loss in the computation of the precision if the high order 16 bit is always chosen although it is guaranteed that higher order two bits will be always zero. Therefore, shifting symbolized by s_y is performed in the convolution process.

$$\hat{y}(n) = y(n) >> s_y \tag{5.16}$$

When we put (5.16) and (5.14) into (5.12), we get

$$\hat{e}(n) = (d(n) - [w^T(n)x(n)]2^{s_y})2^{s_e} \tag{5.17}$$

when we put (5.17) into weight update equation (2.15)

$$w(n+1) = w(n) + \mu[x(n)[d(n) - [w(n)x^T(n)]2^{s_y}]2^{s_e}] \tag{5.18}$$

with further calculation (5.18) becomes

$$w(n+1) = w(n) + \mu 2^{s_y} 2^{s_e} [[x(n)d(n)2^{s-y}] - [w(n)x(n)x^T(n)]] \tag{5.19}$$

As a result, $2^{s_y}2^{s_e}$ is the difference between the floating point and fixed point step sizes.

CHAPTER 6

COMPUTER SIMULATIONS

There are many changing parameters in the adaptive filters, therefore, some high precision tests are performed on the computer by using MATLAB before studying in fixed point environment. For this purpose, filter lengths, step sizes, and performances of various adaptive algorithms will be examined. Moreover, effects of acoustic characteristics of recording environments will be studied. However, it would be difficult to compare the performance of algorithms with mathematical expressions if neither the noise reference signal nor the original speech signal is known.

In order to do some studies in high precision MATLAB environment, there are several estimated methods. One way to achieve this goal is to add noises artificially to the known desired signal. In this method, audio path between microphones are simulated. However, it has a great importance to work with original recordings instead of simulating the audio path between microphones. Therefore, this method is not preferred.

Another method is to work on a case where no speech signal is available. In this case, adaptive algorithms mentioned up to now will try to get rid of the noise. Therefore, error signal e(n) will be attenuated to zero in the ideal case. Hence, in order to understand the effect of filter sizes and step sizes of the filters on the performance of algorithms, the attenuation value between desired signal and error signal can be compared. This method gives a great advantage to understand the performances of different adaptive filters with different lengths and different step sizes under various types of real signals however; this simplified case will not simulate the case where both speech and noise exists.

Finally, the last method is to

• play the speech signal alone and record it in both microphones

• play the noise signal alone and record it in both microphones

• then sum noise and speech part of both microphones on MATLAB and perform the adaptive algorithms

Comparing with other methods mentioned above, this method simulates better since both speech and noise signals are taken into account and characteristics of the environment where recordings are made are also taken into account.

Therefore, a test set up is constructed in order to compare the performances of adaptive filters. The test setup consists of two microphones, Edirol UA-1000 USB audio capture shown in Figure 6.1, a speaker and a MATLAB code recording the data in wave format. Sennheiser ME 104 and Sennheiser ME 64 microphones both of which have a cardioid acoustic pick up pattern are used as primary and secondary microphones respectively. Speaker is used in order to simulate the noise. Experimental studies are done in order to decide the optimum location of the microphones and it is experimentally seen that the location of the microphones might have significant effects on the results. Therefore, once it is decided, the position of the microphones does not change during the recordings.

Figure 6.1: Edirol UA-1000 USB Audio Capture

6.1 LMS Filter Length and Step Size Experiments

In order to understand the effect of filter length and step size choices and acoustic characteristics of recording environments on performances of LMS based adaptive speech enhancement algorithms, SNR measurements are performed with various types of noises in two different environments. The first environment is a moderate-reverberant room with dimensions 6.5m * 2.3m * 2.5m. In Figure 6.2 and Figure 6.3, front and back views of test setup are given. The second one is an anechoic chamber with dimensions 0.8m * 0.8m * 2.0m. Sampling rate is chosen as 8 kHz. LMS filter is run for 100000 samples and both input and output SNR values are measured from last 50000 samples.

Figure 6.2: Front View of Recording Test Setup

Figure 6.3: Back View of Recording Test Setup

6.1.1 Single 500Hz Tone Noise

In this part noise signal consists of single 500Hz sinusoidal signal. In Table 6.1 and Table 6.2, SNR values of the output signal for two different acoustic environment are given in dB where input SNR is equal to 0.87 dB for all cases. From both Table 6.1 and Table 6.2, it is clear that high improvement in SNR values can be achieved with very low filter lengths. It is an expected result because of the periodicity of the noise signal. The effect of the step size selection is also seen from this table. Approximately 13 dB increase in SNR value is held at most in the moderate-reverberant room whereas 25dB increase is held in anechoic room. This difference clearly shows the effect of the acoustic characteristics of environments on the performance of speech enhancement algorithm.

Table6.1: Output SNR (in dB) of LMS Algorithm for Single 500Hz Sinusoidal Noise in a Moderate Reverberant Room

Filter Length \ Step Size	0.0001	0.001	0.005	0.01	0.05	0.1
16	1.1	3.17	9.94	12.58	12.99	12.83
32	1.34	5.32	12.85	13.27	12.96	12.51
64	1.82	8.92	13,6	13.55	12.87	12.11
128	2.75	12.61	13.64	13.5	12.22	10.94
256	4.53	13.67	13.59	13.3	11.18	9.37
512	7.61	13.74	13.41	12.9	9.58	7.43
1024	11.64	13.69	12.95	12.01	7.65	5.63
2048	13.57	13.5	12.01	10.46	5.73	3.96

Table6.2: Output SNR (in dB) of LMS Algorithm for Single 500Hz Sinusoidal Noise in an Anechoic Room

Filter Length \ Step Size	0.0001	0.001	0.005	0.01	0.05	0.1
16	0.95	1.55	4.17	7.32	22.92	23.86
32	1.01	2.21	7.32	13.13	24.18	23.73
64	1.15	3.53	13.20	21.66	24.23	23.39
128	1.42	6.09	21.75	24.97	23.53	22.05
256	1.94	10.90	24.99	24.62	22.00	19.83
512	2.98	18.82	24.61	23.82	19.81	16.96
1024	5.02	24.73	23.83	22.52	16.95	13.57
2048	8.89	24.81	22.52	20.56	13.57	10.11

6.1.2 Multiple Tone Noise

In this part 500Hz, 1000Hz and 2000Hz tones are added and multiple tone noise is held from their compositions.

Table6.3: Output SNR (in dB) of LMS Algorithm for Multiple Tone Sinusoidal Noise in a Moderate Reverberant Room

Filter Length \ Step Size	0.0001	0.001	0.005	0.01	0.05	0.1
16	-6.06	3.07	11.14	16.39	18.01	17.26
32	-3.76	5.43	17.11	18.92	18.12	16.5
64	-0.447	9.63	20.06	20.21	17.41	14.83
128	2.35	16	20.14	19.55	15.05	11.85
256	4.52	20.02	19.82	18.37	12.08	8.4
512	7.95	20.51	18.41	16.19	8.5	4.29
1024	13.86	20.16	16.22	13.3	4.32	-0.59
2048	19.43	18.98	13.31	9.8	-0.62	diverge

Table6.4: Output SNR (in dB) of LMS Algorithm for Multiple Tone Sinusoidal Noise in an Anechoic Room

Filter Length \ Step Size	0.0001	0.001	0.005	0.01	0.05	0.1
16	-8.06	-4.00	9.77	17.34	18.60	18.31
32	-7.58	0.01	17.35	18.60	18.47	17.75
64	-6.65	6.84	19.00	19.13	18.08	16.52
128	-4.86	15.76	19.39	19.22	16.58	14.03
256	-1.52	19.09	19.18	18.51	14.00	10.77
512	4.21	19.34	18.50	17.14	10.77	7.39
1024	13.01	19.27	17.12	14.87	7.38	4.29
2048	18.74	18.76	14.84	11.81	4.29	1.81

In Table 6.3 and Table 6.4, SNR values of the output signal are given in dB where input SNR is equal to -8.73 dB for all cases. Unlike the single tone case, increasing filter lengths lead to more serious increment in the attenuation levels. However, relatively small filter lengths still lead to sufficient attenuation levels. This is expected since the stationarity of the noise signal is still reserved with a longer period. Therefore, very small filter lengths does not lead as good results as ones of single tone case. There is not any significant differences between SNR values of Table 6.3 and Table 6.4. Maximum approximately 28 dB increase is held in the SNR value. This is not expected result since same algorithm increases SNR approximately 13 dB in single tone experiment. However, it becomes clear when Fourier Transforms of the desired signal given in Figure 6.4 is examined. The desired speech signal does not have any dominant component in 1000Hz and 2000Hz. However, 500Hz has a dominant effect on this desired speech signal. Because of the correlation between noise and speech signals, performance is degraded. However, it is also expected to see higher increase in SNR if single 1000Hz tone was used instead of 500Hz. It is experimentally seen that the output SNR value of single tone experiment with 1000Hz is higher than those of both single 500Hz tone and multiple tone experiments.

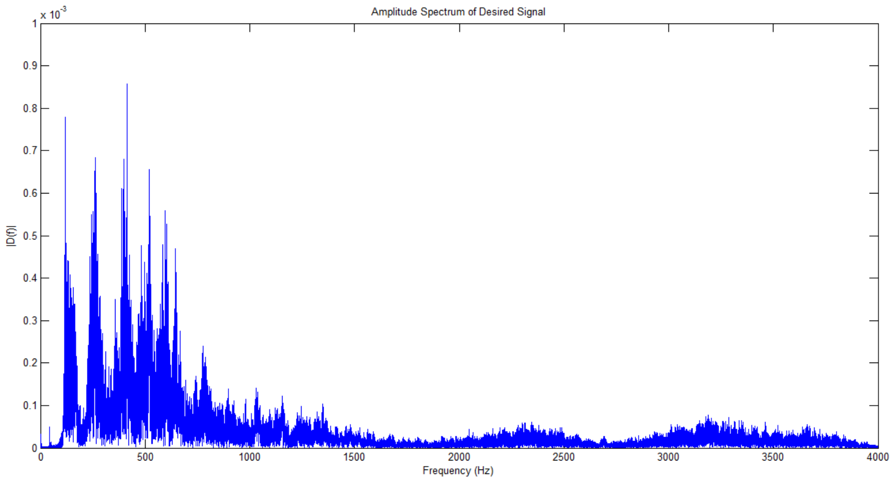

Figure 6.4: Fourier Transform of Desired Signal

6.1.3 Armored Military Vehicle Noise

In Table 6.5 and Table 6.6, SNR values of the output signal are given in dB where input SNR is equal to -5.09 dB for all cases. Comparing with the previous results, it is clear that this recorded armored military vehicle noise is more difficult one because of its non stationarity. In order to have a greater output SNRs, very large filter lengths are necessary. A filter having less than 512 taps do not contribute significantly to the output SNR. This is also an expected result because stationarity in the noise signal compared to single or multiple tone experiments is lost. Effect of the acoustic characteristics of environments on the performance is also seen from the differences between SNR values in Table 6.5 and Table 6.6.

Table6.5: Output SNR (in dB) of LMS Algorithm for Nonstationary Armored Military Vehicle Noise in a Moderate Reverberant Room

Filter Length \ Step Size	0.0001	0.001	0.005	0.01	0.05	0.1
16	-5.04	-4.86	-4.63	-4.61	-4.6	-4.59
32	-4.97	-4.46	-4.23	-4.21	-4.2	-4.19
64	-4.91	-4.16	-3.8	-3.77	-3.76	-3.75
128	-4.86	-3.89	-3.35	-3.3	-3.26	-3.23
256	-4.77	-3.4	-2.65	-2.57	-2.49	-2.44
512	-4.65	-2.75	-1.07	-0.84	-0.77	-0.86
1024	-4.45	-1.64	1.9	2.58	1.98	1.16
2048	-4.41	-1.35	3.12	3.98	2.21	1.01

Table6.6: Output SNR (in dB) of LMS Algorithm for Nonstationary Armored Military Vehicle Noise in an Anechoic Room

Filter Length \ Step Size	0.0001	0.001	0.005	0.01	0.05	0.1
16	-3.99	-3.47	-3.39	-3.36	-3.22	-3.14
32	-0.37	0.24	0.29	0.27	-0.20	-0.68
64	3.94	5.28	5.32	4.67	1.73	0.53
128	5.09	6.89	6.12	4.67	0.97	-0.06
256	6.05	7.58	5.07	3.12	-0.01	diverge
512	6.12	6.40	3.01	1.44	diverge	diverge
1024	5.85	4.71	1.35	0.23	diverge	diverge
2048	5.36	2.87	0.08	diverge	diverge	diverge

6.1.4 Tank Noise

In Table 6.7 and Table 6.8 SNR values of the output signal are given in dB where input SNR is equal to -3.39 dB for all cases. Similar to the armored military vehicle noise, a filter having less than 512 taps do not contribute significantly to the attenuation value. Therefore, in order to the have a successful result, larger filter taps should be used in the implementations. Effect of the acoustic characteristics of rooms on the performance is also seen from the differences between SNR values in Table 6.7 and Table 6.8.

Table6.7: Output SNR (in dB) of LMS Algorithm for Nonstationary Tank Noise in a Moderate Reverberant Room

Step Size / Filter Length	0.0001	0.001	0.005	0.01	0.05	0.1
16	-3.34	-3.02	-2.63	-2.58	-3.39	-2.51
32	-3.29	-2.65	-2.13	-2.07	-2.53	-2.02
64	-3.2	-2.14	-1.36	-1.28	-2.03	-1.19
128	-3.03	-1.18	-0.23	0.1	0.01	0.02
256	-2.74	-0.12	1.21	1.55	1.82	1.77
512	-2.31	0.6	2.33	2.89	3.31	3.09
1024	-1.75	1.1	4.04	5.17	5.58	4.66
2048	-1.23	1.31	4.61	5.86	5.37	3.74

Table6.8: Output SNR (in dB) of LMS Algorithm for Nonstationary Tank Noise in an Anechoic Room

Step Size / Filter Length	0.0001	0.001	0.005	0.01	0.05	0.1
16	-1.10	-0.44	-0.36	-0.36	-0.49	-1.01
32	2.35	4.22	4.41	4.32	3.32	-0.66
64	4.77	8.02	8.81	8.29	4.76	2.37
128	4.73	9.25	10.07	8.58	3.30	2.75
256	4.64	9.01	8.19	6.26	1.38	1.24
512	4.42	8.30	6.23	4.14	diverge	diverge
1024	4.37	7.31	3.88	1.93	diverge	diverge
2048	4.21	5.74	1.75	0.11	diverge	diverge

From Table 6.1, Table 6.3, Table 6.5 and Table 6.7, it is clearly seen that LMS filter with larger tap is necessary in order to deal with nonstationary signals. Selection of the step size also has a great importance. A step size giving good results for one noise could give insufficient results for another type of noise. Moreover, it can also diverge. This is an expected result because autocorrelation matrix of the noisy speech signal changes with added noise. Stability of the filter according to step size depends on the eigenvalues of the autocorrelation matrix (2.19). Moreover, in the real time applications if there is not any prior knowledge about the environment that the algorithm will work, a necessity for a time varying step size algorithm which increase the computational load occurs. However, if this is a specific system for instance a system for the pilot on plane cockpit, helicopter cockpit or for tank or car driver, adaptive algorithms with fixed step sizes should also give sufficient performance. Effect of the acoustic characteristics of recording environment is also seen in these simulations. In an anechoic room higher increment in SNRs can be held by filters with smaller taps compared to ones in a reverbarent room.

6.2 Filter Performances

Performance measurements for different adaptive filter types, Widrow's LS method with LMS and modified LMS and SAD with NLMS are performed in this part. Adaptive filters are run for 100000 samples and both input and output SNR values are measured from last 50000 samples. During all the experiments step sizes are kept fixed. They are not the step sizes giving the maximum SNR for all cases. The performance of these adaptive filters with fixed step sizes under different noise conditions are tested. Only moderate-reverberant room recordings are used.

6.2.1 Single 500Hz Tone Noise

In Table 6.9, SNR values of the output signal are given in dB for various types of filters where input SNR is equal to 0.87 dB for all cases and noise signal is recorded 500Hz sinusoidal tone. As expected, all of the filters give high SNRs for this single tone noise due to the stationarity of input signal.

Table6.9: Output SNR (in dB) of Adaptive Algorithms for Single 500Hz Sinusoidal Noise

Filter Type / Filter Length	LS with LMS	LS with SD-LMS	LS with SE-LMS	LS with SS-LMS	LS with NLMS	SAD with NLMS
16	12.58	13.05	13.1	13.1	12.58	9.74
32	13.27	13.13	13.38	13.3	13.27	12.56
64	13.55	13.16	13.63	13.34	13.57	12.31
128	13.5	12.67	13.59	12.98	13.5	11.27
256	13.3	11.76	13.46	12.38	13.3	9.74
512	12.9	10.31	13.3	11.32	12.91	7.67
1024	12.01	8.37	12.8	9.73	12.03	5.76
2048	10.46	6.38	11.64	7.77	10.51	4.1

6.2.2 Multiple Tone Noise

In Table 6.10, SNR values of the output signal are given in dB for various types of filters where input SNR is equal to -8.73 dB for all cases and noise signal is recorded multi tone noise consists of 500Hz, 1000Hz and 2000Hz sinusoidal tones. As expected, all of the filters give successful results for this multiple tone noise. Figure 6.5 shows the noisy input signal, LMS based LS filter output signal for 2048 taps and desired signal in time domain.

Table6.10: Output SNR (in dB) of Adaptive Algorithms for Multiple Tone Sinusoidal Noise

Filter Type / Filter Length	LS with LMS	LS with SD-LMS	LS with SE-LMS	LS with SS-LMS	LS with NLMS	SAD with NLMS
16	16.39	6.49	18.15	7.96	16.38	16.61
32	18.92	11.6	19.2	15.27	18.92	16.48
64	20.21	17.91	20.21	16.08	20.21	15.0
128	19.55	19.82	19.41	12.04	19.59	12.21
256	18.37	19.42	17.92	8.02	18.51	8.69
512	16.19	17.72	15.63	3.63	16.57	4.28
1024	13.3	15.04	12.75	0.32	14.21	-0.86
2048	9.8	11.41	9.24	-5.22	11.72	diverge

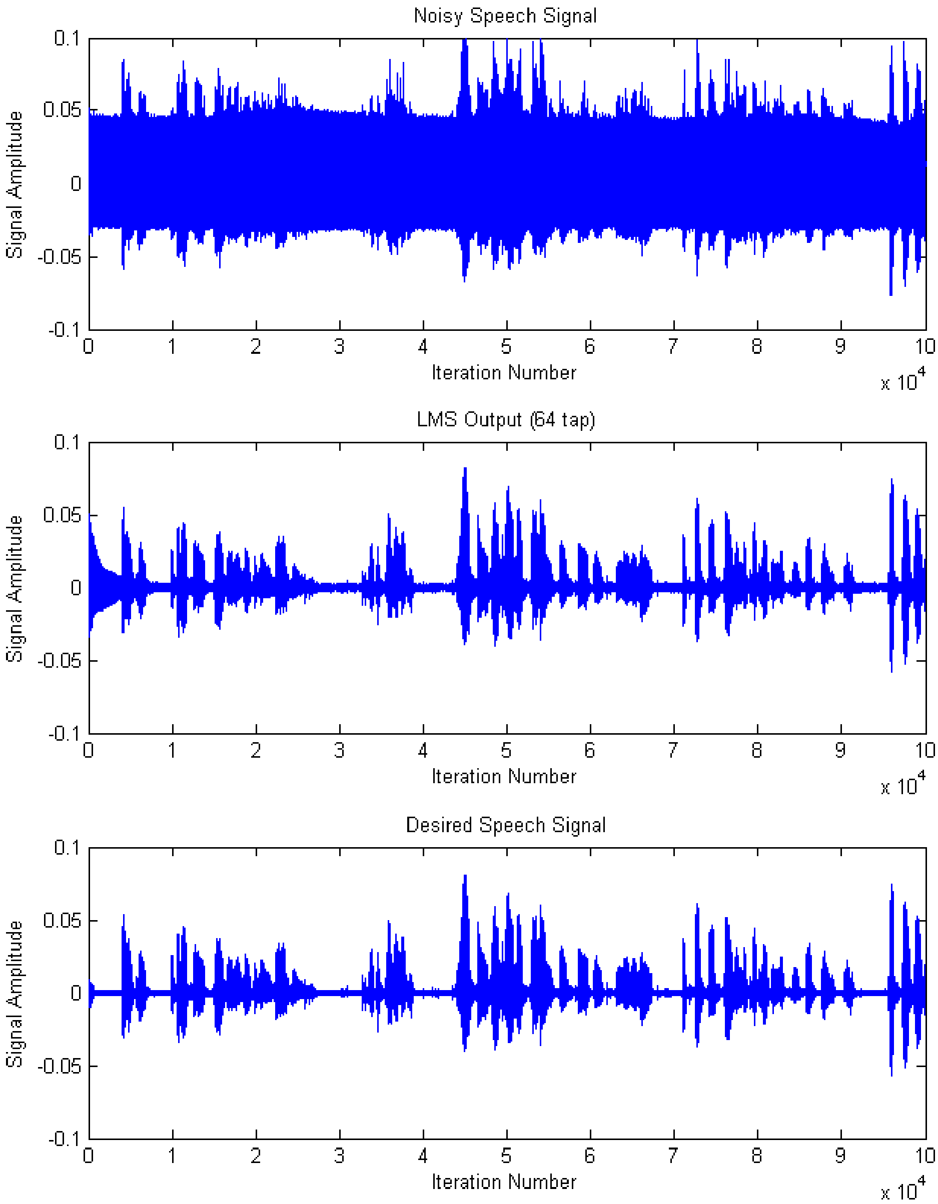

Figure 6.5: Noisy, LMS Output and Desired Signals in Time Domain

6.2.3 Armored Military Vehicle Noise

In Table 6.11, SNR values of the output signal are given in dB for various types of filters where input SNR is equal to -5.09 dB for all cases and noise signal is recorded nonstationary armored vehicle noise. As expected, performance of all of the filters degrades significantly compared to experiments performed with stationary noise signals. Figure 6.6 shows the noisy input signal, output signal of LS filter with SE-LMS weight adaptation for different taps and desired signal in time domain for all 100000 samples. From Figure 6.6 and Table 6.11, it is clearly seen that there is a need for an adaptive filter with larger taps in order to deal with this noise. In the implementations, larger filters should be implemented to get a better enhancement.

Table6.11: Output SNR (in dB) of Adaptive Algorithms for Nonstationary Armored Vehicle Noise

Filter Type / Filter Length	LS with LMS	LS with SD-LMS	LS with SE-LMS	LS with SS-LMS	LS with NLMS	SAD with NLMS
16	-4.61	-4.61	-4.63	-4.61	-4.61	-4.35
32	-4.21	-4.21	-4.23	-4.21	-4.21	-4.31
64	-3.77	-3.78	-3.79	-3.77	-3.77	-3.82
128	-3.3	-3.3	-3.33	-3.3	-3.3	-3.25
256	-2.57	-2.56	-2.62	-2.58	-2.57	-2.39
512	-0.84	-0.8	-0.9	-0.8	-0.84	-0.92
1024	2.58	2.52	2.7	2.83	2.75	1.1
2048	3.98	3.49	4.51	4.33	3.95	2.2

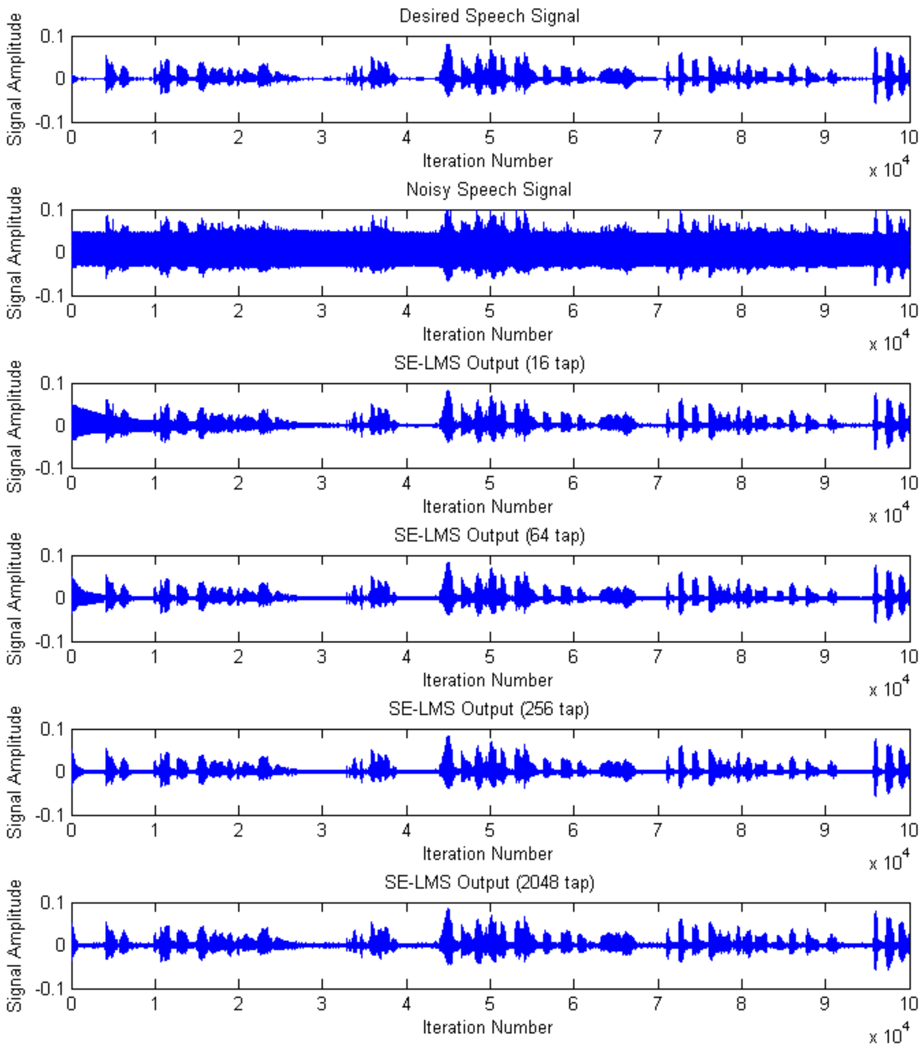

Figure 6.6: Desired, Noisy and SE-LMS Output Signals in Time Domain

6.2.4 Tank Noise

In Table 6.12, SNR values of the output signal are given in dB for various types of filters where input SNR is equal to -3.39 dB for all cases and noise signal is recorded non stationary tank noise.

Table6.12: Output SNR (in dB) of Adaptive Algorithms for Nonstationary Tank Noise

Filter Type / Filter Length	LS with LMS	LS with SD-LMS	LS with SE-LMS	LS with SS-LMS	LS with NLMS	SAD with NLMS
16	-2.58	-2.55	-2.62	-2.57	-2.58	-2.83
32	-2.07	-2.05	-2.1	-2.06	-2.07	-2.5
64	-1.28	-1.24	-1.32	-1.26	-1.28	-2.01
128	0.1	-0.05	-0.15	-0.07	-0.1	-0.59
256	1.55	1.69	1.5	1.69	1.55	0.93
512	2.89	3.14	2.9	3.23	2.88	0
1024	5.17	5.54	5.45	6.04	5.13	4.82
2048	5.86	5.72	6.45	6.67	5.77	3.78

Figure 6.7 shows the noisy input signal, SAD filter output signal for 2048 taps and desired signal in time domain.

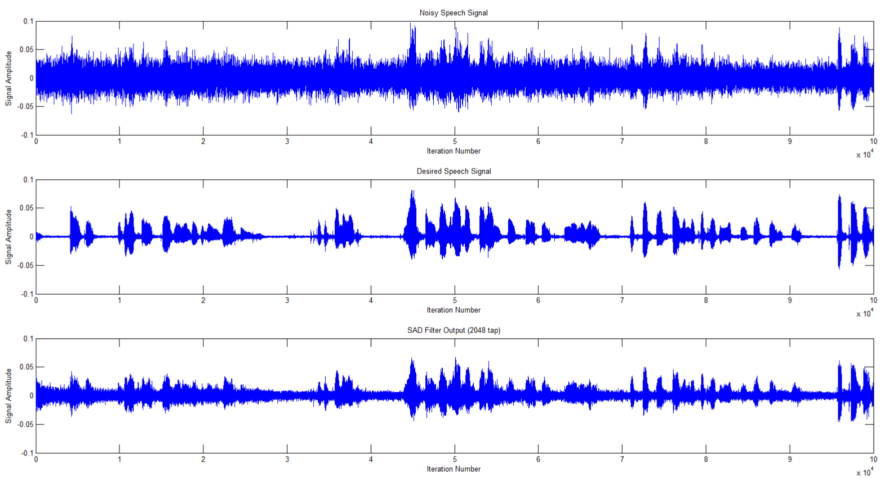

Figure 6.7: Desired, Noisy and SAD Output Signals in Time Domain

Similarly to the previous test, it is clearly seen that there is a need for an adaptive filter with larger taps in order to deal with real non stationary noises. Therefore, in the implementations, larger filters should be implemented to get a better enhancement.

6.3 Performances SAD and Widrow's LS

In this part of the thesis, SAD and Widrow's LS methods are compared. In order to compare the performances of LS and SAD algorithms, filter size is selected as 2048 and it is kept fixed during the experiment. Step sizes giving the highest output SNR are used in experiments. Recorded signals are processed further in order to increase the signal leakage. SNR of reference microphone is changed and experiments are performed with these different SNRs. LMS weight adaptation methods are used in both algorithms. In Table 6.13, differences between input and output SNRs of LS and SAD methods for different NSR at reference microphone are given in dB.

Table6.13: SNR improvement of LS and SAD

LS with LMS	SAD with LMS	NSR at Reference
6.57	6.63	38
6.55	6.61	32
6.4	6.58	24
6.36	6.52	12
5.6	5.7	0

Performances of both SAD and LS algorithms are very close. The difference between SNR values is less than 0.2dB. However, complexity of LS method is half of that of SAD method since weight adaptation is performed twice in SAD method. Therefore, LS filter having twice length of that of SAD filter can be implemented when the complexities of algorithms are kept same. Therefore, it is seen that Widrow's LS is superior to SAD especially when complexity is considered. Moreover,signal leakage is mostly prevented when microphones are placed appropriately in the test setup.

CHAPTER 7

SUBJECTIVE LISTENING TESTS

Test setup mentioned in Chapter 6 is used in order to evaluate the performances of implemented adaptive filters. Two different types of recordings are performed with this setup. In the first part, noise source is played and talker also speaks and recording is performed. In the second part, firstly recording of the talker is done. Then, recording of noise is performed by keeping the setup same. These recordings are processed with MATLAB in order to create wave format files with different SNRs.

Another test setup consists of a computer, C5505 EZDSP USB STICK development kit and a recording device is used in order to perform speech enhancement. The computer is used to play two channel wave format audio file. Line output of computer is connected into the line input of development kit and the result is given to the recording device through the line output of development kit.

In this chapter, performances of implemented adaptive filters are evaluated by using these setups. The speech quality and intelligibility can be quantified using subjective and objective measures. In this chapter, subjective speech intelligibility measures are used. In subjective listening tests, a large number of human participants are asked to listen to the speech voice and provide a rating of perceived quality of it in accordance with a predetermined opinion scale. In Absolute Category Rating (ACR), the filter output signal is rated by itself. The result is rated on a scale of 1 to 5 where 1 shows “bad” and 5 means “excellent”. This scaling is known as the Mean Opinion Score, MOS.

In Degradation Category Rating (DCR), the opposite version of MOS scale is presented. Both original speech signal and enhanced filter output signal are listened and disturbances in speech signals are rated relatively in accordance with Degradation MOS rating scale of 1 to 5 where 1 shows “very annoying” and 5 means “inaudible.

In another approach called as Comparison Category Rating (CCR), the various filter output signals are listened and these signals are compared with each other. The advantage of the CCR method over the DCR procedure is the possibility to assess speech processing that either degrades or improves the quality of the speech [16].

Diagnostic Rhyme Test (DRT) is also used in order to measure the intelligibility of speech signals. DRT is an ANSI standard for measuring speech intelligibility [17]. In DRT, listeners are given a list composed of 96 word pairs constructed from a consonant-vowel-consonant sound and differing only in the first phoneme. Test words are played and listeners are asked to select one of the words from given word pair. Carrier sentences are not used.

In order to compare the performances of systems, output of adaptive filters and noisy input signals are recorded by using the test setups mentioned above. In this part, only real noises which are armored military vehicle noise, tank noise and helicopter noise recorded in a moderate-reverberant room are used. Artificial noises like single tone or multiple tones are not used since they are not the noises encountered in practical applications. Recorded voices are listened to a group consists of ten people. Three different tests are performed. In the first part, DRT is conducted. In the second part, a paragraph consists of three sentences is listened and rated according to MOS. In the third part, three different sentences are filtered with LMS, SE-LMS and NLMS. Then, two of them are listened to a test group and listeners are asked to choose which one is better. For the simplicity SD-LMS and SS-LMS are not implemented. SAD algorithm is not implemented because its complexity is higher and according to the simulation results of Chapter 6, Widrow’s LS performs better than SAD algorithm. The simulation results of Chapter 6 are used in order to determine the implemented filter length. LMS and SE-LMS algorithms are implemented with 2000 taps. Moreover, NLMS with 400 taps is implemented. Due to the computational complexities and hardware constraints mentioned in Chapter 3, NLMS with a filter length as long as LMS cannot be implemented on 5505 DSP. The performances of LMS with 2000 taps, SE-LMS with 2000 taps and NLMS with 400 taps are compared with three different tests.

7.1 Diagnostic Rhyme Test

In Chapter 6, SNR values of algorithms are measured. However, it is known that quality and intelligibility of speech can be degraded although SNR at filter output gets higher. Moreover, SNR and segmental SNR cannot be measured with real data because of the alignment. In this part, the main concern is to evaluate the intelligibility of speech in Turkish language. Therefore, word list of Turkish Intelligibility Test (TIT) developed by TÜBİTAK-UEKAE's Acoustics laboratory is used [18]. It is developed by considering DRT standard [17]. DRT is based on how the initial constant is recognized properly. Similar consonants are selected. The word pair lists include the words that can be detected false by human ear due to voicing, nasality, sustention, sibilation, compactness and graveness effects [18]. Orders of noisy words and processed words are mixed. During the presentation process, subjects are listened rhyming pairs with Sennheiser HD 202 headphones in a silent room. Tests are conducted with native speakers of the Turkish language. The results of Turkish Intelligibility Test is given in Figure 7.1. The overall results for different input SNR values are given in Table 7.1.

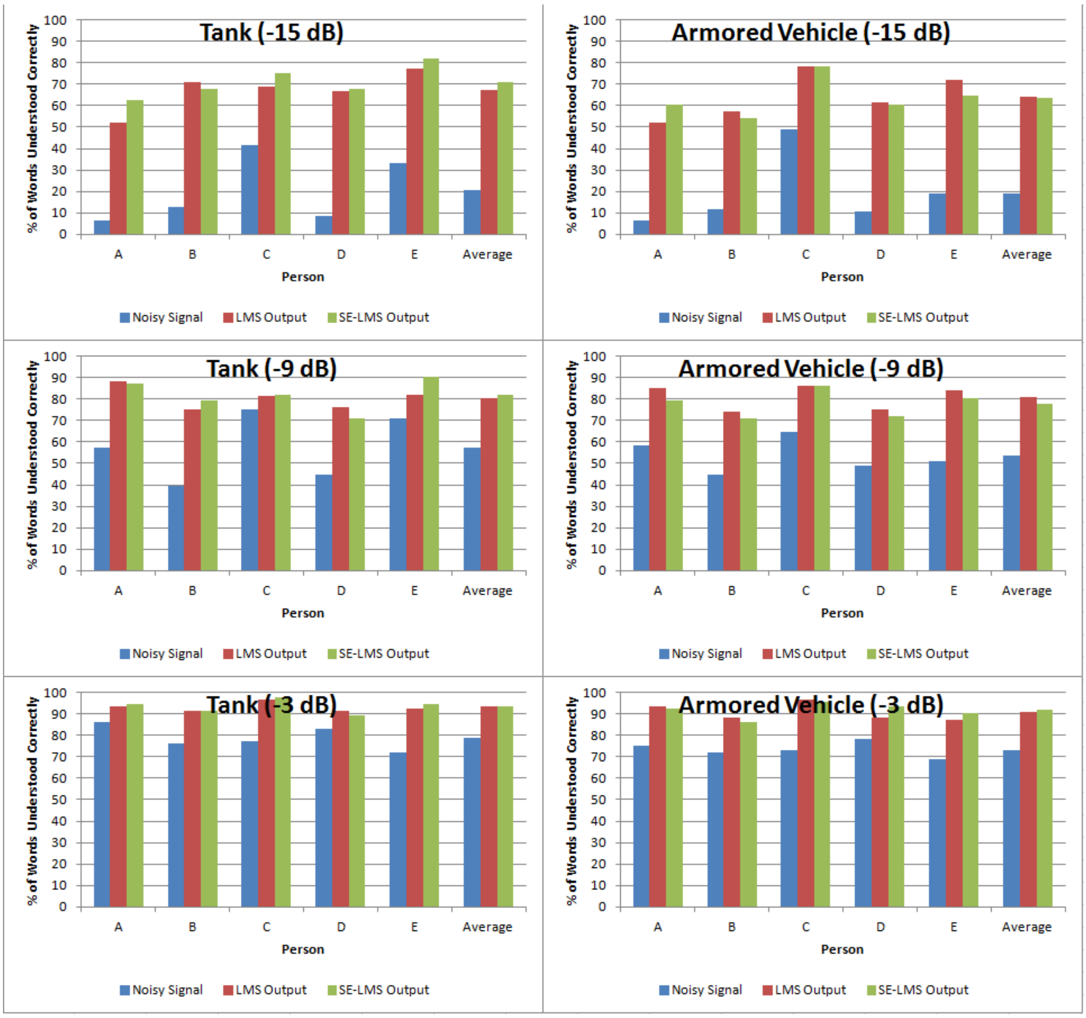

Figure 7.1: TIT Results

Table7.1: Overall TIT Results

	Armored Vehicle Noise			Tank Noise		
Input SNR	Noisy Input Signal	LMS Output Signal	SE-LMS Output Signal	Noisy Input Signal	LMS Output Signal	SE-LMS Output Signal
-15 dB	19.2	64.2	63.5	20.4	67.1	71.0
-9 dB	53.5	81.0	77.7	57.5	80.6	82.1
-3 dB	73.3	91.0	91.9	79.0	93.3	93.8

From this table, the effect of Widrow's LS speech enhancement method on the intelligibility of speech is seen. When input SNR is -9 dB, approximately 30% increase is seen in the number of words that are understood correctly. When input SNR is -15 dB, the improvement in the intelligibility of speech becomes clearer. The intelligibility of noisy speech signal is very low at -15dB SNR. Approximately two of every ten words cannot be understood. The intelligibility can be increased up to 70%. When input SNR is higher, this difference is lower since the intelligibility of noise signal is relatively higher. However, 20% increase is seen in the number of words that are understand correctly. Both LMS and SE-LMS weight adaptation algorithms give similar results.

7.2 Mean Opinion Score

The average MOS of adaptive filters is shown in Table 7.2.

Table7.2: MOS Results for LMS, SE-LMS and NLMS Based Speech Enhancement Systems

Filter Type / Noise Type	Noisy Input	LMS	SE-LMS	NLMS
Armored Military Vehicle Noise	1	3.1	3.2	1.4
Tank Noise	1	3.1	2.8	1.9
Helicopter Noise	1	2.3	2.2	1.8

From these results, it is clear that both SE-LMS and LMS algorithms show high performances in all noise types. It is an expected result under the light of the computer simulation results of Chapter 6. In other words, results achieved in MATLAB simulations are held with their fixed point implementations. Performance of NLMS is lower compared to others. This is also an expected result since in Chapter 6, it is shown that in order to deal with these natural noises in a moderate-reverberant room, larger filter taps are needed. However, compared to noisy input signal, NLMS still give improvement in the intelligibility of speech.

7.3 Mutual Comparison

As mentioned before, filter outputs of three different sentences with each noise type are recorded. Adaptive filters are run from beginning for each of these three sentences. That is, after the first sentences are recorded, filter is reinitialized and filter coefficients are set to zero. Then, the following recording is done. By this way, effect of the convergence of the filters is also seen. The results are given in Table 7.3. Here, the blank box means that corresponding filter is not evaluated.

Table7.3: Mutual Comparison of LMS, SE-LMS, NLMS Adaptive Filters

Filter Type / Noise Type	LMS	SE-LMS	NLMS
Armored Military Vehicle Noise First Sentence	6	4	
Armored Military Vehicle Noise Second Sentence	10		0
Armored Military Vehicle Noise Third Sentence	0	10	
Tank Noise First Sentence	10		0
Tank Noise Second Sentence	7	3	0
Tank Noise Third Sentence		10	0
Helicopter Noise First Sentence	5		5
Helicopter Noise Second Sentence	6	4	
Helicopter Noise Third Sentence		4	6

Both LMS and SE-LMS always perform better that NLMS especially for armored military vehicle and tank noises. It is expected since filter length of NLMS is insufficient. For the helicopter noise, performance of NLMS is closer to other algorithms. However, it is seen in the previous experiment that performances of all filters with helicopter noise is generally poor. In all mutual comparison tests, LMS is superiour to SE-LMS although the difference is very close. It could be said that convergences of LMS is a bit better. Moreover, LMS and SE-LMS show similar performances like MOS test done in the second part. As a result of this, it can be said that LMS and SE-LMS are superiour to NLMS algorithm due to the filter length limitation.

CHAPTER 8

CONCLUSIONS

In this thesis work, a speech enhancement system is implemented real-time on TMS3-20C5505 fixed point DSP. Speech enhancement problem and existing methods for the solution of this problem are examined. Two different models for this problem i.e., SAD source separation and Widrow's LS speech enhancement models are studied. Assumptions made in these models, their advantages, disadvantages and contributions to the solution of speech enhancement problem are examined.

Differences between these models for instance signal leakage into the reference microphone is studied. Some parameters like acoustic transfer path of speech signal on primary microphone and acoustic transfer path of noise signal on secondary microphone can not be recovered by both methods. However, effect of these parameters can be minimized by arranging the position of microphones. These models are compared with each other and uncorrelated noise, correlation between speech and noise are studied in experiments.

Limitations of fixed point implementation, differences between fixed point and floating point adaptive algorithms are also examined. Speech degradation due to the signal leakage into the reference microphone and uncorrelated signals between microphones are studied. Under the light of these terms, the position of microphones is experimentally studied. It was expected to see that SAD overcomes LS when signal leakage in the reference microphone increases. However, it is experimentally seen that the increment in the output SNR is less than 0.2 dB and complexity of LS is half of that of SAD since weight adaptation is performed twice in SAD method. It is experimentally seen that LS methods enhance speech signal a bit better than SAD when the complexity is considered. Nonminimum phase matrix inversion problem is also examined.

Effects of the selection of step size and filter length of adaptive algorithms on their performances under artificial and real noises are examined in MATLAB. Under the light of these studies, appropriate step sizes and filter lengths have been selected for implemented filters.

Adaptive LMS, sign LMS, NLMS weight adaptation algorithms are examined. Theoretical complexities of these weight adaptation algorithms are examined. Moreover, differences between theoretical and practical complexities of weight adaptation algorithms due to the selected DSP hardware are studied. Computation power of DSP and complexities of algorithms in terms of clock cycles are examined.

Negative effect of the correlation between original speech signal and additive noise signal on the performance of adaptive filters is simulated. It is experimentally shown that the filter performance degrades when the correlation between original speech signal and additive noise signal increases. Moreover, it is seen that performances of adaptive filters under real recorded noises degrade largely compared to stationary artificial single and multiple tone noises. Moreover, effects of the acoustic characteristics of recording environment on the performance of adaptive algorithm are experimentally shown. When the reverberant in the recording place increases, it becomes more difficult to enhance speech because length of FIR filters modeling the acoustic paths needs to be selected longer. Effects of the acoustics characteristics of environment on SNR values and filter tap selection are examined in detail.

LS and SAD algorithms are simulated in MATLAB and evaluated according to SNR values. Moreover, LMS, SE-LMS and NLMS based speech enhancement algorithms are implemented on DSP and performances of these filters under various real noises with different input SNRs are rated with DRT, MOS and mutual comparison. Speech enhancement is succeeded even at low SNR levels. In the simulations, it is seen that NLMS is superior to other adaptation algorithms. However, NLMS with enough taps in order to enhance speech in a moderate reverberant room cannot be implemented due to the hardware constraints. Intelligibility of speech is measured by using DRT. When input SNR is -9 dB, approximately 30% increase is seen in the number of words that are understood correctly. By using subjective tests, it is shown that implemented systems with LMS and SE-LMS algorithms enhance intelligibility.

As a result, an effective speech enhancement system is implemented and sufficient improvement is held on the intelligibility of the speech transmitted to the other parties under real recorded noises such as tank noise, armored military vehicle noise and helicopter noise.

8.1 Future Work

As a future work, NLMS based speech enhancement algorithm having as much taps as LMS and sign error LMS can be implemented on a DSP with different architecture or higher clock rate. With a DSP having more computational capacity, the resolution of codec can be increased. Moreover, time varying step size algorithms can be studied. Performances of these algorithms can be compared.

Another future work is to develop this study and get a self powered end product. This product can be placed into the existing radios and improvement in the intelligibility of speech transmitted to the other party can be measured.

REFERENCES

[1] Hu, Jwusheng; Le, Mingtang. Speech enhancement for mobile phones based on the imparity of two-microphone signals. In: Information and Automation, ICIA'09. International Conference on. IEEE, p. 606-611, 2009

[2] Rahmani, Mohsen, et al. A real-time architecture for dual-microphone speech enhancement systems. In: GCC Conference and Exhibition, 5th IEEE. IEEE, p. 1-5, 2009

[3] Yoganathan, V.; Moir, T. J. Multi-microphone adaptive neural switched Griffiths-Jim beamformer for noise reduction. In: Signal Processing (ICSP), IEEE 10th International Conference on. IEEE, p. 299-302, 2010

[4] Virag, Nathalie. Single channel speech enhancement based on masking properties of the human auditory system. Speech and Audio Processing, IEEE Transactions, 7.2: 126-137, 1999

[5] J. Chen, J. Benesty, Y. Huang, A Minimum Distortion Noise Reduction Algorithm With Multiple Microphones, Audio, Speech, and Language Processing, IEEE Transactions, March 2008

[6] Widrow, Bernard, et al. Adaptive noise cancelling: Principles and applications. Proceedings of the IEEE, 63.12: 1692-1716, 1975

[7] Weinstein, Ehud; Feder, Meir; Oppenheim, Alan V. Multi-channel signal separation by decorrelation. Speech and Audio Processing, IEEE Transactions, 1.4: 405-413, 1993

[8] Van Gerven, Stefaan; Van Compernolle, Dirk. Signal separation by symmetric adaptive decorrelation: stability, convergence, and uniqueness. Signal Processing, IEEE Transactions on, 43.7: 1602-1612, 1995

[9] Gupta, Riten; Hero III, Alfred O. Transient behavior of fixed point LMS adaptation. In: Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on. IEEE, p. 376-379, 2000

[10] Gitlin, R.; Mazo, J.; Taylor, M. On the design of gradient algorithms for digitally implemented adaptive filters. Circuit Theory, IEEE Transactions, 20.2: 125-136, 1973

[11] Sayed, Ali H. Fundamentals of adaptive filtering. John Wiley and Sons, 2003

[12] Harrison, W., Jae Lim, and Elliot Singer. "A new application of adaptive noise cancellation." Acoustics, Speech and Signal Processing, IEEE Transactions on 34.1: 21-27, 1986

[13] Bermudez, José Carlos M.; Bershad, Neil J. A nonlinear analytical model for the quantized LMS algorithm-the arbitrary step size case. Signal Processing, IEEE Transactions, 44.5: 1175-1183, 1996

[14] Benesty, Jacob; Chen, Jingdong; Huang, Yiteng; Microphone Array Signal Processing, Springer 2008

[15] Jeong, Jinsoo. Analysis of system identification and modified application to two-microphone speech enhancement, International Journal OF Ccircits, Systems And Signal Processing, Issue 2, Volume 3, 2009

[16] Benesty, J., Sondhi, M.M., and Huang, Y.A., Springer Handbook of Speech Processing, New York: Springer-Verlag; 2008

[17] STANDARD, A. N. S. I. S3. 2-1989. Method for measuring the intelligibility of speech over communi cation Systems. New York

[18] Palaz, H., Bicil, Y., Kanak, A., and Dogan, M. U., New Turkish intelligibility test for assessing speech communication systems. Speech communication, 47(4), 411-423, 2005

[19] Texas Instruments,TMS320C55x DSP Mnemonic Instruction Set, Literature Number: SPRU374G, October 2002

[20] Texas Instruments, TMS320VC5505/5504 DSP Inter-IC Sound (I2S) Bus User's Guide, Literature Number: SPRUFP4, September 2009 – Revised January 2010

[21] Texas Instruments, TLV320AIC3204 Application Reference Guide, Literature Number: SLAA557, November 2012

[22] Texas Instruments, TMS320C5515/14/05/04/VC05/VC04 DSP Inter-Integrated Circuit (I2C) Peripheral, Literature Number: SPRUFO1A, September 2009

[23] Texas Instruments, TMS320C5505 Fixed-Point Digital Signal Processor, Literature Number: SPRS660E, August 2010–Revised January 2012

[24] Sinha, Priyabrata Speech processing in embedded systems, Springer, 2010

APPENDIX A

TMX320C5505 CONFIGURATION

There are four PLL control registers and source clock of PLL unit, its multiplication and division values are set through them. RTC Clock in DSP is given as PLL input clock and 120 MHz is set as frequency of CPU clock. By giving external clock, TMX320C5505 can be run with 150 MHz CPU clock.

There is an I2C bus available in DSP. With this interface, DSP can control more than one codec or other devices supporting this communication protocol. In this study, control data between codec and DSP is sent through this interface. I2C has two pins SCL and SCA.

There are four I2S buses which allow serial transfer of full-duplex streaming data between DSP and an external I2S peripheral device such as an audio codec. DSP is configured as master whereas codec is configured as slave. I2S has four pins I2S Clock, I2S Frame Sync Clock, I2S Data Transmit and I2S Data Receive. I2S supports different data lengths with different data formats. Packed mode is set used. From codec to DSP, LADC data 1, RADC data 1, LADC data 2 and RADC data 2 i.e. four 16 bit words are sent. LADC data 1 corresponds to signal sampled from codec's left analog to digital converter at time t whereas RADC data 2 corresponds to signal sampled from codec's right analog to digital converter at time t+1. In this study, former one corresponds to noisy input signal whereas second one corresponds to reference signal and I2S0 bus is used for the data transmission between DSP and audio codec.

This DSP has four DMA controllers which are used to move data among internal memory, external memory and peripherals without intervention from the CPU and in the background of CPU operation. They are all identical but they cannot access every resources.

In this study, I2S is configured such that it will give notification when data is received from the codec. This notification can be configured as DMA interrupt or CPU interrupt. In this study, a CPU interrupt is used. When this interrupt occurs, four 16 bit data are read from four I2S data receive registers, then after the CPU operations, four output 16 bit data are written to four I2S data transmit registers respectively in order to give them to audio codec. I2S is configured with I2SCTRL register such that mono mode is enabled, loopback is disabled, frame-synchronization polarity is low, receive data is sampled on the rising edge and transmit data shifted on the falling edge, 1 bit data delay, packed mode is set, no sign extension, 16 bit data word, master and I2S/left-justified format.

APPENDIX B

TLV320AIC3204 STEREO AUDIO CODEC CONFIGURATION

TLV320AIC3204 has two ADCs, two DACs. It supports operations from 8 kHz mono voice playback to audio stereo 192 kHz DAC playback. In this study 8 KHZ sampling rate with 16 bit word length is used.

ADC path of TLV320AIC3204 has six analog inputs which can be mixed and/or multiplexed in single-ended and/or differential configuration, two programmable gain amplifiers (PGA) with a range of 0 to +47.5dB, digital volume control with a range of -12 to +20 dB and AGC.

DAC path of TLV320AIC3204 has 2 headphone amplifiers which are usable in single-ended or differential mode having analog volume setting with a range of -6 to +29 dB. It also has 2 line-out amplifiers which are usable in single-ended or differential mode having analog volume setting with a range of -6 to +29dB and digital volume control with a range of -63.5 to +24 dB.

Audio data can flow between DSP and the TLV320AIC3204 on the digital audio data serial interface or audio bus. This very flexible bus includes left or right-justified data options, support for I2S or PCM protocols, programmable data length options, a TDM mode for multichannel operation, very flexible master-slave configurability for each bus clock line, and the ability to communicate with multiple devices within a system. In this study I2S protocol is used in order to transfer audio data between codec and DSP. In this study, I2S0 bus of DSP is used for the data transmission between DSP and audio codec.

TLV320AIC3204 control interface supports SPI or I2C communication protocols. In this study, in order to configure codec from DSP, I2C protocol is used. In the following some important settings of the TLV320AIC3204 for this study are given.

By writing 0x01 into page 0 reg1, software reset is performed.

By writing 0x0D into page 0 reg27, I2S, 16 bit and master mode are set.

By writing 0x03 into page 0 reg4, PLL clock input is selected. By writing 0x08 into page 0 reg6, 0x07 into page 0 reg7, 0x80 into page 0 reg8, 0xA0 into page 0 reg30, 0x91 into page 0 reg5, 0x02 into page 0 reg13, 0x00 into page 0 reg14, 0x80 into page 0 reg20, 0x88 into page 0 reg11, 0x82 into page 0 reg12, 0x88 into page 0 reg18 and reg19, PLL settings, ADC and DAC clock settings for 8 kHz sampling are performed.

By writing 0x08 into page 1 reg12, LDAC is routed to HPL and similarly, by writing 0x08 into page 1 reg13, RDAC is routed to HPR.

By writing 0x02 into page 1 reg64, RDAC gain is controlled by LDAC. By writing 0xF4 into page 1 reg65, RDAC and LDAC gains are set to -35 dB. By writing 0xD4 into page 1 reg63, RDAC plays right audio interface data, LDAC plays left audio interface data. In addition, LDAC and RDAC are powered up.

By writing 0x00 into page 1 reg16 and 0x00 into page 1 reg17, HPL and HPR are unmuted with 0 dB gain. By writing 0x09 into page 1 reg30, HPL and HPR are powered up.

By writing 0x3A into page 1 reg59 and reg60, MIC PGA left and right are unmuted.

By writing 0xC0 into page 0 reg81, left and right ADC are powered up and by writing 0x00 into page 0 reg82, they are unmuted.