FPGA IMPLEMENTATION OF ADAPTIVE NOISE CANCELLATION

MAIN PROJECT REPORT

Submitted by

VISHNU BALACHANDRAN (IEANEEC052)

In partial fulfilment for the award of the Degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

Under the guidance of

Mr. ARUN N

Assistant Professor, ECE Department

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CALICUT UNIVERSITY INSTITUTE OF ENGINEERING AND TECHNOLOGY

THENHIPALAM-673636

APRIL 2017

CERTIFICATE

*This is to certify that the project report entitled **"FPGA Implementation of adaptive noise cancellation"** submitted by **Vishnu Balachandran** to the University of Calicut towards partial fulfilment of the requirement for the award of the Degree of Bachelor of Technology in Electronics and Communication Engineering is a bonafide record of the work carried out by him under my supervision and guidance.*

Guided by:

Mr. Arun N	**Mrs.Roshna**
Assistant Professor	Head of Department
Dept. of ECE	Dept. of ECE
CUIET	CUIET
Thenhipalam	Thenhipalam

Place : Kohinoor (seal)

Date :

ACKNOWLEDGMENT

Dedicating this work to the almighty God whose abundant grace and mercies enabled its successful completion, I would like to express my profound gratitude to all the people who have inspired and motivated me to make this work successful.

I express my gratitude to **Mr. Balakrishnan Pillai**, Principal of Calicut University Institute of Engineering and Technology and **Mrs. Roshna**, Head of the Department, Department of Electronics and Communication Engineering, Calicut University Institute of engineering and Technology, for their valuable suggestions and constantly inspiring me during the course of this work.

I would like wish to record my sincere thanks to the project coordinator **Mr. Arun N,** Assistant Professor, Electronics and Communication Engineering, Calicut University Institute of Engineering and Technology, for having supported the work related my project.

I extend me hear felt thanks to my project guide **Mr. Arun N** Asst.Professor, Department of Electronics and Communication Engineering, Calicut University Institute of Engineering and Technology, for her expert supervision, encouragement, constructive criticism and constant support given throughout, amidst her busy schedule.

I grateful to all other staffs of Calicut University Institute of Engineering and Technology for providing all the facilities and valuable suggestions for the success of this work. Finally, I extend my sincere thanks to my dear parents and friends who helped me build up confidence.

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Speech is a very basic way for humans to convey information, it has a bandwidth of only 4 kHz. It can convey information with the emotion of a human voice. Certain properties of the speech signal are, it is a one-dimensional signal, with time as its independent variable. It is random in nature, it is non-stationary, and the frequency spectrum is not constant in time. Although human beings have an audible frequency range of 20 Hz to20 kHz, the human speech has significant frequency components only up to 4 kHz. The most common problem in speech processing is the effect of interference noise in the signals. This noise masks the speech signal, reduces its intelligibility and also in noisy environment speech communication is greatly affected by the presence of background acoustic noise. The presence of background noise in speech significantly reduces the intelligibility of speech. Noise reduction algorithms are used to suppress such background noise and improve the perceptual quality and intelligibility of speech. Removing various types of noise is difficult due to the random nature of the noise and the inherent complexities of the speech. Noise reduction techniques usually have a trade-off between the amount of noise removal and speech distortions introduced due to processing of the speech signal. Several techniques have been proposed for this purpose in the area of speech enhancement, like spectral subtraction approach, wiener filter and kalman filter. The performances of these techniques depend on the quality and intelligibility of the processed speech signal. The improvement in the speech signal to noise ratio is the target of most techniques.

1.2 BASIC CONCEPTS

Noise Cancellation makes use of the notion of destructive interference. When two sinusoidal waves superimpose, the resulting waveform depends on the frequency amplitude and relative phase of the two waves. If the original wave and the inverse of the original wave encounter at a junction at the same time, total cancellation occur.

The challenges are to identify the original signal and generate the inverse without delay in all directions where noises interact and superimpose. We will demonstrate the solutions later in the report.

.

CHAPTER 2

LITERATURE REVIEW

2.1 SPEECH SIGNAL

A sound source can be created with either the vocal folds or with a constriction in the vocal tract. The fundamental purpose of speech is communication, i.e., the transmission of messages. According to Shannon's information theory, a message represented as a sequence of discrete symbols can be quanti- fied by its information content in bits, and the rate of transmission of information is measured in bits/second (bps). In speech production, as well as in many human-engineered electronic communication systems, the information to be transmitted is encoded in the form of a continuously varying (analog) waveform that can be transmitted, recorded, manipulated, and ultimately decoded by a human listener. In the case of speech, the fundamental analog form of the message is an acoustic waveform, which we call the speech signal. Speech signals can be converted to an electrical waveform by a microphone, further manipulated by both analog and digital signal processing, and then converted back to acoustic form by a loudspeaker, a telephone handset or headphone, as desired. This form of speech processing is, of course, the basis for Bell's telephone invention as well as today's multitude of devices for recording, transmitting, and manipulating speech and audio signals. Although Bell made his invention without knowing the fundamentals of information theory, these ideas have assumed great importance in the design of sophisticated modern communications systems. Therefore, even though our main focus will be mostly on the speech waveform and its representation in the form of parametric models, it is nevertheless useful to begin with a discussion of how information is encoded in the speech waveform.

Based on the various sound sources we proposed a general categorization of speech sounds. There are various ways to categorize speech sounds. For example, we can categorize speech sounds based on different sources to the vocal tract; we have seen that different sources are due to the vocal fold state, but are also formed at various constrictions in the oral tract. Speech sounds generated with a periodic glottal source are termed voiced; likewise, sounds not so generated are called unvoiced.

There are a variety of unvoiced sounds, including those created with a noise source at an oral tract constriction. Because the noise of such sounds comes from the friction of the moving air against the constriction, these sounds are sometimes referred to as fricatives An example of frication is in the sound “th” in the word “thin” where turbulence is generated between the tongue and the upper teeth. The reader should hold the “th” sound and feel the turbulence. A second unvoiced sound class is plosives created with an impulsive source within the oral tract. An example of a plosive is the “t” in the word “top.” The location of the closed or partial constriction corresponds to different plosive or fricative sounds, respectively. We noted earlier that a barrier can also be made at the vocal folds by partially closing the vocal folds, but without oscillation, as in the sound “h” in “he.” These are whispered unvoiced speech sounds. These voiced and unvoiced sound categories, however, do not relate exclusively to the source state because a combination of these states can also be made whereby vocal fold vibration occurs simultaneously with impulsive or noisy sources. For example, with “z” in the word “zebra,” the vocal folds are vibrating and, at the same time, noise is created at a vocal tract constriction behind the teeth against the palate. Such sounds are referred to as voiced fricatives in contrast to unvoiced fricatives where the vocal folds do not vibrate simultaneously with frication. There also exist voiced plosives as counterparts to unvoiced plosives as with the “b” in the word “boat.”

2.1.1 Speech Coding

Perhaps the most widespread applications of digital speech processing technology occur in the areas of digital transmission and storage of speech signals. In these areas the centrality of the digital representation is obvious, since the goal is to compress the digital waveform representation of speech into a lower bit-rate representation. It is common to refer to this activity as “speech coding” or “speech compression.” Figure 2.1 shows a block diagram of a generic speech encoding/decoding (or compression) system. In the upper part of the figure, the A-to-D converter converts the analog speech signal x(t) to a sampled waveform representation x[n]. The digital signal x[n] is analysed and coded by digital computation algorithms to produce a new digital signal y[n] that can be transmitted over a digital communication channel or stored in a digital storage medium as ˆy[n]. As we will see, there are a myriad of ways to do the encoding so as to reduce the data rate over that of the sampled and quantized speech waveform x[n]. Because the digital representation at this point is often not directly related to the sampled speech waveform, y[n] and ˆy[n] are appropriately referred to as data signals that represent the speech signal. The lower path in Figure shows the decoder associated with the speech coder. The received data signal ˆy[n] is decoded using the inverse of the analysis processing, giving the sequence of samples ˆx[n] which is then converted (using a D-to-A Converter) back to an analog signal ˆxc(t) for human listening. The decoder is often called a synthesizer because it must reconstitute the speech waveform from data that may bear no direct relationship to the waveform. With carefully designed error protection coding of the digital representation, the transmitted (y[n]) and received (ˆy[n]) data can be essentially identical. This is the quintessential feature of digital coding. In theory, perfect transmission of the coded digital representation is possible even under very noisy channel conditions, and in the case of digital storage, it is possible to store a perfect copy of the digital representation in perpetuity if sufficient care is taken to update the storage medium as storage technology advances. This means that the speech signal can be reconstructed to within the accuracy of the original coding for as long as the digital representation is retained. In either case, the goal of the speech coder is to start with samples of the speech signal and reduce (compress) the data rate required to represent the speech signal while maintaining a desired perceptual fidelity. The compressed representation can be more efficiently transmitted or stored, or the bits saved can be devoted to error protection.

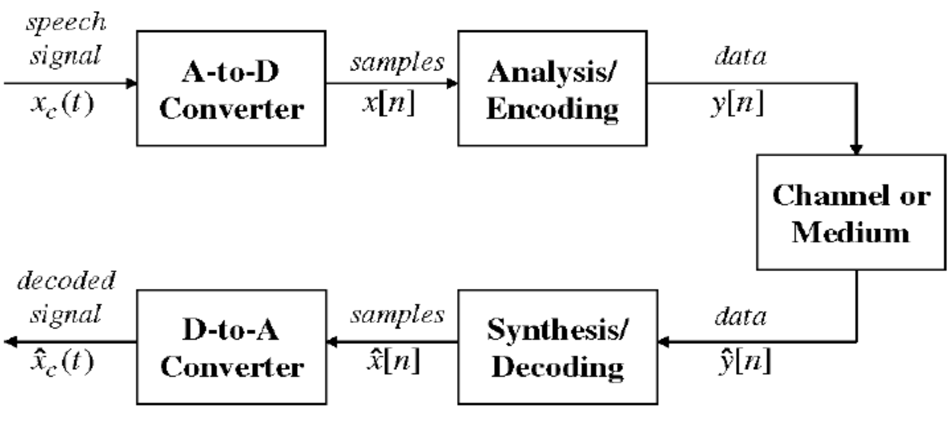


Fig. 2.1 Speech coding block diagram of encoder and decoder.

Speech coders enable a broad range of applications including narrowband and broadband wired telephony, cellular communications, voice over internet protocol (VoIP) (which utilizes the internet as a real-time communications medium), secure voice for privacy and encryption (for national security applications), extremely narrowband communications channels (such as battlefield applications using high frequency (HF) radio), and for storage of speech for telephone answering machines, interactive voice response (IVR) systems, and pre-recorded messages. Speech coders often utilize many aspects of both the speech production and speech perception processes, and hence may not be useful for more general audio signals such as music. Coders that are based on incorporating only aspects of sound perception generally do not achieve as much compression as those based on speech production, but they are more general and can be used for all types of audio signals. These coders are widely deployed in MP3 and AAC players and for audio in digital television systems.

2.1.2 Types of Audio Formats

There exist a variety of sound file formats among which an mp3 enjoys its high popularity. MP3 files are ubiquitous on the Net and it is no exaggeration to say that the MP3 is now a household name in the world. People, however, enjoy an mp3 music file without knowing what it is and its strengths and weaknesses in comparison other popular music file formats.

- MP3

The most popular sound file format these days is .MP3 for both Mac and PCs. Other file formats include .AIFF (for Mac); .AU for Mac and UNIX; .WAV for the PC; and .RA for Real Audio, a proprietary system for delivering and playing streaming audio on the Web. The MP3 format is a compression system for music. The MP3 format helps reduce the number of bytes in a song without hurting the quality of the song's sound. The goal of the MP3 format is to compress a CD-quality song by a factor of 10 to 14 without losing the CD quality of the sound. With MP3, a 32 megabyte song on a CD compresses down to 3 megabytes or so. This lets you download a song in minutes rather than hours, and it lets you store hundreds of songs on your computer's hard disk without taking up that much space. It can be broken up into pieces, and each piece is still playable. The feature that makes this possible (header less file format) also means that MP3 files can be made to stream across the net real-time.

- WAV:

The Microsoft .WAV file format is a technique for storing analog audio data in a digital format. It is capable of storing waveform data in many different formats and an array of compression types. A *.WAV file is a digital recording of the sounds made by any instrument or human voice. It basically cannot be modified. When a PC plays back a WAV file, it converts numbers in the file into audio signals for the PC's speakers. A complete tune recorded in .WAV format is always very large. A .WAV file is always true to the original instruments that produced the music. WAV files are simple and widely used, especially on PCs. Many applications have been developed to play WAV files and it is the native sound format for Windows. Later versions of Netscape Navigator (3+) and Microsoft Internet Explorer (2+) support the WAV format.

- au (AU):

AU is short for audio, a common digital sound file format used on Unix machines and the standard audio file format for the Java programming language. The file has a very simple structure: the file header specifies the basic parameters of the sound - sampling rate, sample size, number of channels and type of encoding - followed by the sound data. AU files usually employ the 8000 hertz u-Law encoding method. A simple, well-established sound format which is the most commonly supported browser sound file format.

- rm/ ram (RealAudio):

A mainstay of internet audio for many years, and a standard for streaming audio and video. RealAudio has many quality settings. The RealAudio files are still smaller than 128kb/s MP3 files, but the quality is also less. RealPlayer is the best (and only official) choice for RealAudio files.

2.2 CLASSIFICATION OF NOISE

Noise may be defined as any unwanted signal that interferes with the communication, measurement or processing of an information-bearing signal. Noise is present in various degrees in almost all environments. For example, in a digital cellular mobile telephone system, there may be several variety of noise that could degrade the quality of communication, such as acoustic background noise, thermal noise, electromagnetic radio-frequency noise, co-channel interference, radio-channel distortion, echo and processing noise. Noise can cause transmission errors and may even disrupt a communication process; hence noise processing is an important part of modern telecommunication and signal processing systems. The success of a noise processing method depends on its ability to characterise and model the noise process, and to use the noise characteristics advantageously to differentiate the signal from the noise. Depending on its source, a noise can be classified into a number of categories, indicating the broad physical nature of the noise, as follows:

- Acoustic noise:

  Emanates from moving, vibrating, or colliding sources and is the most familiar type of noise present in various degrees in everyday environments. Acoustic noise is generated by such sources as moving cars, air-conditioners, computer fans, traffic, people talking in the background, wind, rain, etc.
- Electromagnetic noise:

  Present at all frequencies and in particular at the radio frequencies. All electric devices, such as radio and television transmitters and receivers, generate electromagnetic noise.
- Electrostatic noise:

  Generated by the presence of a voltage with or without current flow. Fluorescent lighting is one of the more common sources of electrostatic noise.
- Channel distortions:

  Echo, and fading: due to non-ideal characteristics of communication channels. Radio channels, such as those at microwave frequencies used by cellular mobile phone operators, are particularly sensitive to the propagation characteristics of the channel environment.

- Processing noise:

The noise that results from the digital/analog processing of signals, e.g. quantisation noise in digital coding of speech or image signals, or lost data packets in digital data communication systems.

2.2.1 Noise Reduction

The consequences of exposing people to noise from various sources may vary from short term effects such as sleep disturbance to long term effects such as permanent hearing loss. To reduce the noise from source reaching our year involves various methods which can be categorised into

- Passive Noise Control

Passive Noise control is a method in which the noise from the source is not allowed to reach the ear of the person. This is done by blocking the path of the noise using absorbing materials or by reflecting the noise in some other direction. Thermocol or polystyrene, clothes and wood are some examples of the materials which absorb the noise and reduce the adverse effects occurring from it.

- Active Noise Control

Active Noise Control is a very effective electronic method to reduce the effect of the noise in an environment. It is basically generation of anti-noise, equal in magnitude and opposite in phase with the noise. The anti-noise and the noise are destructively interfered to remove the effects of noise from the path of the noise.

2.3 EXISTING METHODOLOGIES:

2.3.1 Analog:

- LUGE’S PATENT:

The design of acoustic ANC utilizing a microphone and an electronically driven loudspeaker was first proposed in 1936 patent by Lueg .Block diagram of his system with additional legends is shown in The Lueg system is a monopole consisting of a source). An acoustic noise is picked up by a microphone, whose signal is processed through an electronic device to create cancelling signal at a loudspeaker that opposes to the primary noise. Electronic device includes phase inverter with a time delay. Signal is further amplified and led to the loudspeaker disposed downstream from the microphone. At the position of the loudspeaker noise and “antinoise” cancel each other and form a quite zone. This type of system is called feedforward. It senses noise before it passes secondary source. Unfortunately Lueg was never able to demonstrate his idea successfully because it was oversimplified and field of electronics was not sufficiently advanced.

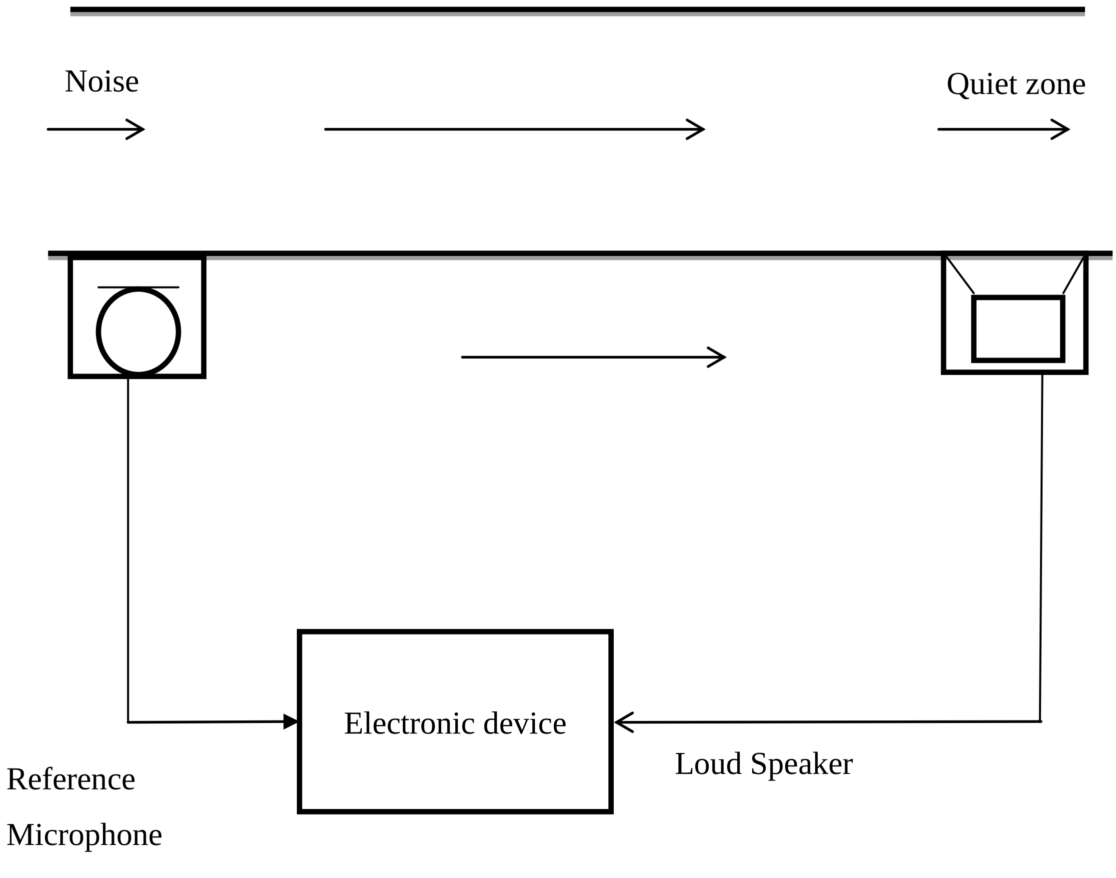


Fig 2.2 Lueg’s system

- PATENT BY OLSON AND MAY

In 1953 Olson and May carried their researches on developing an electronic sound absorber, which appeared to be successful over small volumes in a unidirectional sound field. This type of system is called feedback system. It cancels noise without the benefit of an upstream reference input. Previously described systems were not adaptive.

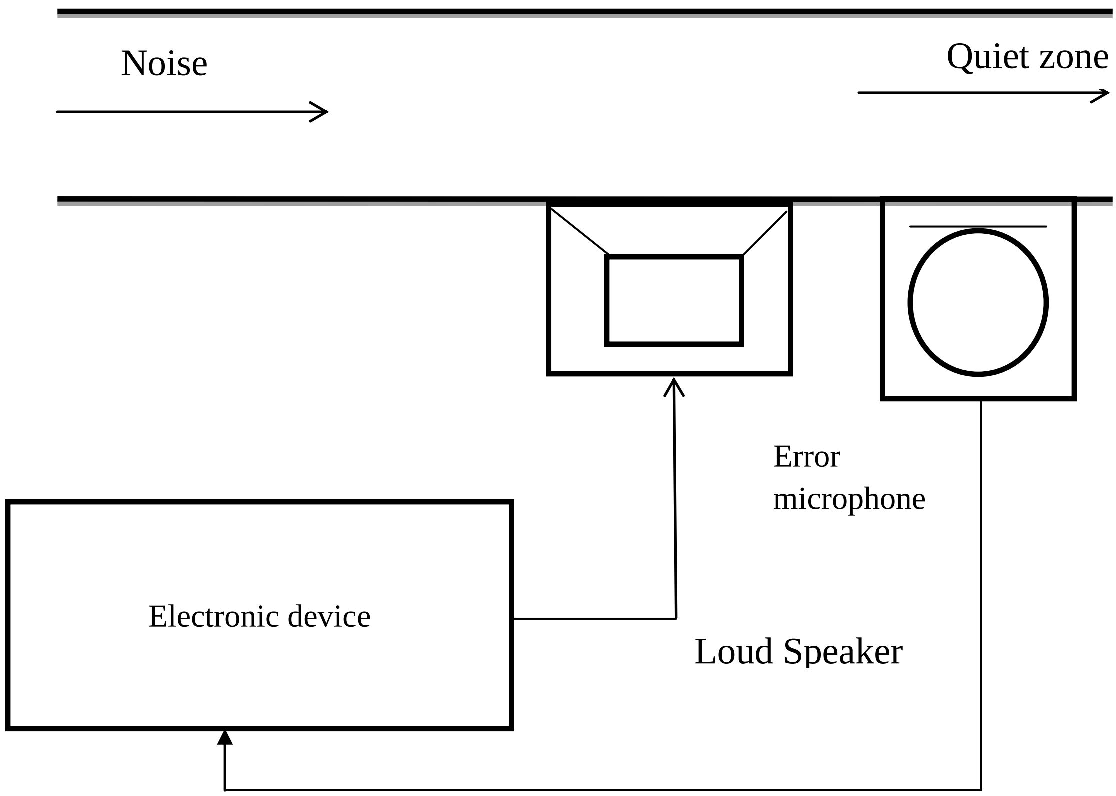


Fig 2.3 Feedback system (analog)

- CONOVER SYSTEM

In 1956, Conover described an active system for reducing noise radiated by large transformers It consisted of three parallel independent channels, realized with three band pass filters, as can be seen in a block diagram. Central frequencies of these filters correspond to the main frequency emitted by a transformer and its first two harmonics. The main frequency of noise is twice the electric grid (line) frequency because transformer magnetic core contracts twice with each period of grid frequency. To achieve precise cancelling signal Conover applied manual adjustment of phase and amplitude for each of three frequency components of the cancelling signal. Later, in 1968 Onoda and Kido developed an automated system for transformer's noise reducing.

2.3.2 Digital:

- FEEDFORWARD SYSTEM

In a feedforward system the signal from the reference microphone is processed by an adaptive digital filter to produce a cancelling signal. Essentially it is the system identification approach where the acoustic path is modelled with an adaptive filter. Real systems face two additional problems, the influence of the secondary path and an acoustic feedback.

Loudspeaker and amplifier that drives it, form the secondary path that has a transfer function S(z). The influence of the secondary path S(z) is corrected with the introduction of the filter S(z) that filters the input signal x(t). hence the name Filtered-X LMS algorithm (FXLMS), 1981. S(z) provides also accurate time alignment of referent and error signals that are used by the adaptive algorithm. In a feedforward system the signal from the reference microphone is processed by an adaptive digital filter to produce a cancelling signal. Essentially it is the system identification approach where the acoustic path is modelled with an adaptive filter. Real systems face two additional problems, the influence of the secondary path and an acoustic feedback.

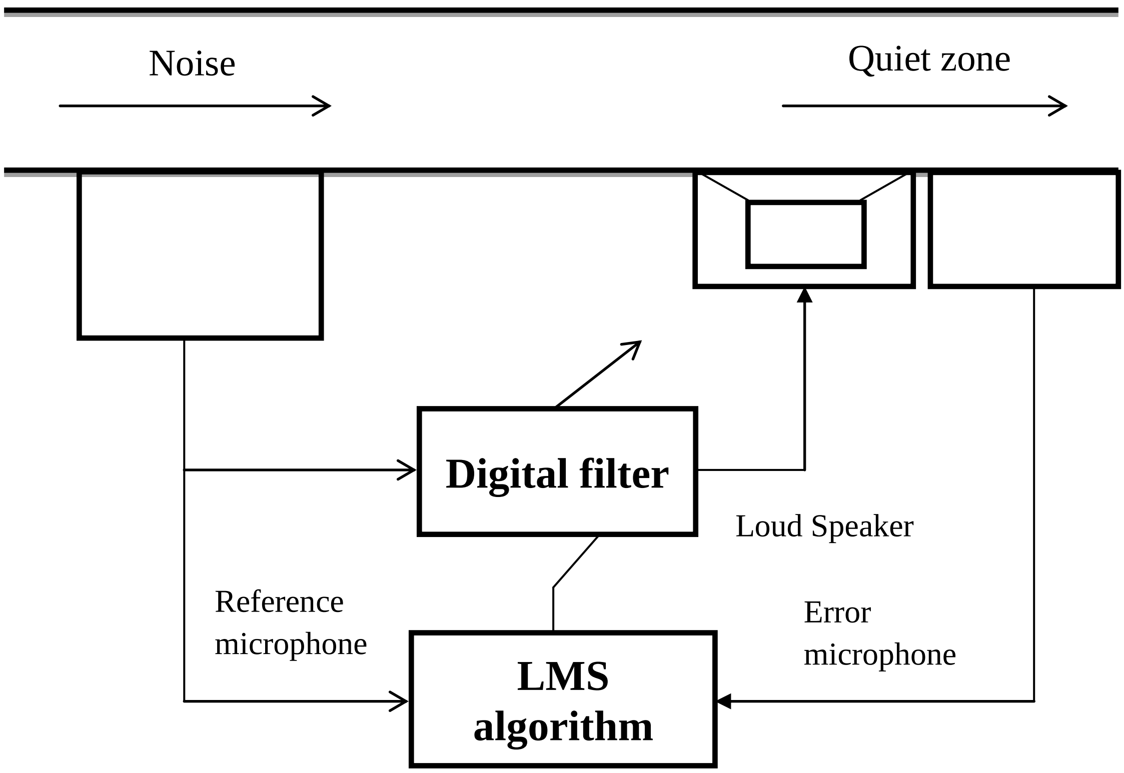


Fig 2.4 Feedforward system (digital, adaptive)

- FEEDBACK SYSTEMS

Block diagram of a feedback system is shown. In the feedback system the signal from the error microphone is processed by an adaptive digital filter to produce a cancelling signal. Under simplified (idealistic) assumption the secondary path S(z) can be approximated with a delay. It cancels noise without the benefit of an upstream reference input. Previously described systems were is not adaptive. This system will be adaptive .

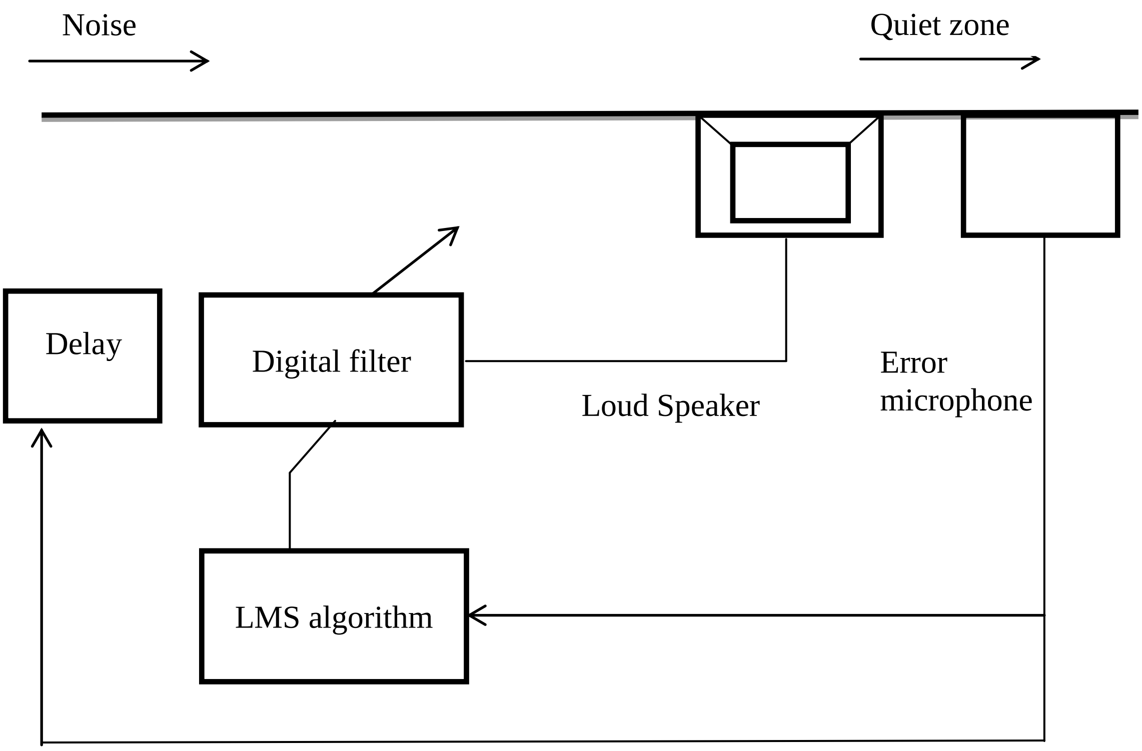


Fig 2.5 Feedback systems

CHAPTER 3

PROPOSED SYSTEM

3.1 ADAPTIVE FILTERS

As their own name suggests, adaptive filters are filters with the ability of adaptation to an unknown environment. This family of filters has been widely applied because of its versatility (capable of operating in an unknown system) and low cost (hardware cost of implementation, compared with the non-adaptive filters, acting in the same system).

The ability of operating in an unknown environment added to the capability of tracking time variations of input statistics makes the adaptive filter a powerful device for signal-processing and control applications. Indeed, adaptive filters can be used in numerous applications and they have been successfully utilized over the years.

As it was before mentioned, the applications of adaptive filters are numerous. For that reason, applications are separated in four basic classes: identification, inverse modeling, prediction and interference cancelling. These classes will be detailed in the next chapter.

All the applications above mentioned, have a common characteristic: an input signal is received for the adaptive filter and compared with a desired response, generating an error. That error is then used to modify the adjustable coefficients of the filter, generally called weight, in order to minimize the error and, in some optimal sense, to make that error being optimized, in some cases tending to zero, and in another tending to a desired signal.

3.1.1 Active Noise Cancelling

The active noise cancelling (ANC), also called adaptive noise cancelling or active noise canceller belongs to the interference cancelling class. The aim of this algorithm, as the aim of any adaptive filter, is to minimize the noise interference or, in an optimum situation, cancel that perturbation. The approach adopted in the ANC algorithm, is to try to imitate the original signal s(n).

In this study, the final objective is to use an ANC algorithm to cancel speech noise interference, but this algorithm can be employed to deal with any other type of corrupted signal, as it will be presented in the section. A scheme of the ANC can be given below.

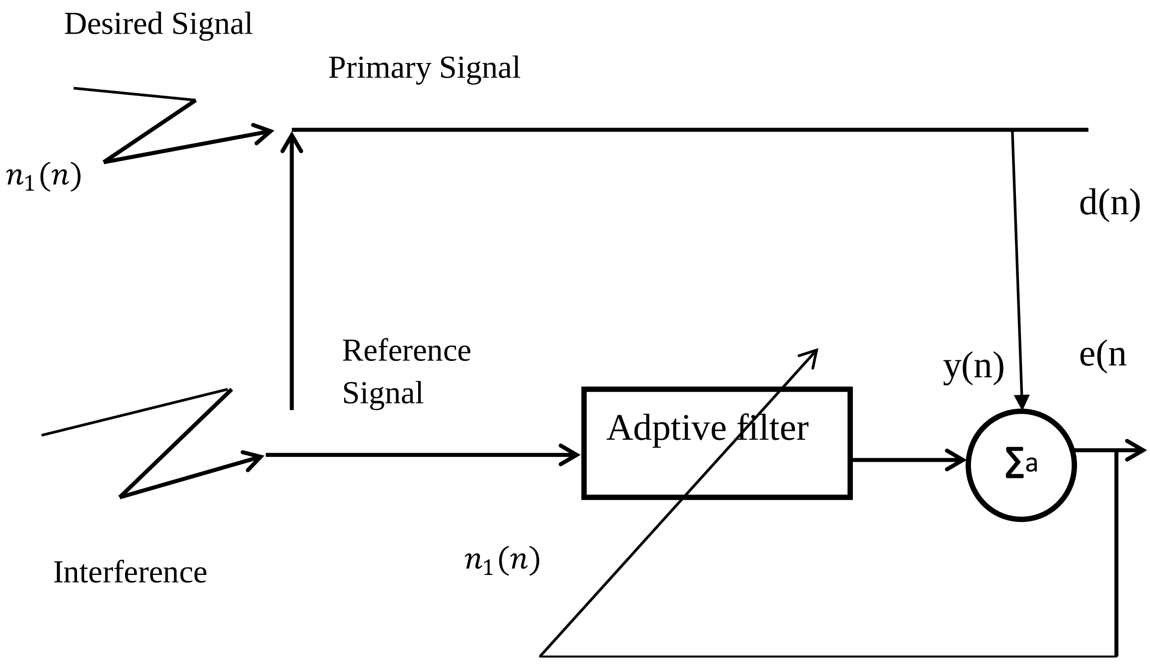


Fig 3.1 Active noise canceller

In the ACN, as explained before, the aim is to minimize the noise interference1 that corrupts the original input signal. In the figure above, the desired signal d(n) is composed by an unknown signal, that we call s(n) corrupted for an additional noise n2(n), generated for the interference. The adaptive filter is then installed in a place that the only input is the interference signal n1(n). The signals n1(n) and n2(n) are correlated. The output of the filter y(n) is compared with the desired signal d(n), generating an error e(n). That error, which is the system output, is used to adjust the variable weights of the adaptive filter in order to minimize the noise interference. In an optimal situation, the output of the system e(n) is composed by the signal s(n), free of the noise interference n2(n).

3.1.2 Four Fundamental Classes

- Adaptive Identification

The adaptive identification is an approach to model an unknown system. The unknown system is in parallel with an adaptive filter, and both are receiving the input signal. The output of the unknown system provides the reference signal for the adaptive digital filter. Applications for adaptive identification include room acoustic identification, channel estimation, echo cancellation and so on.

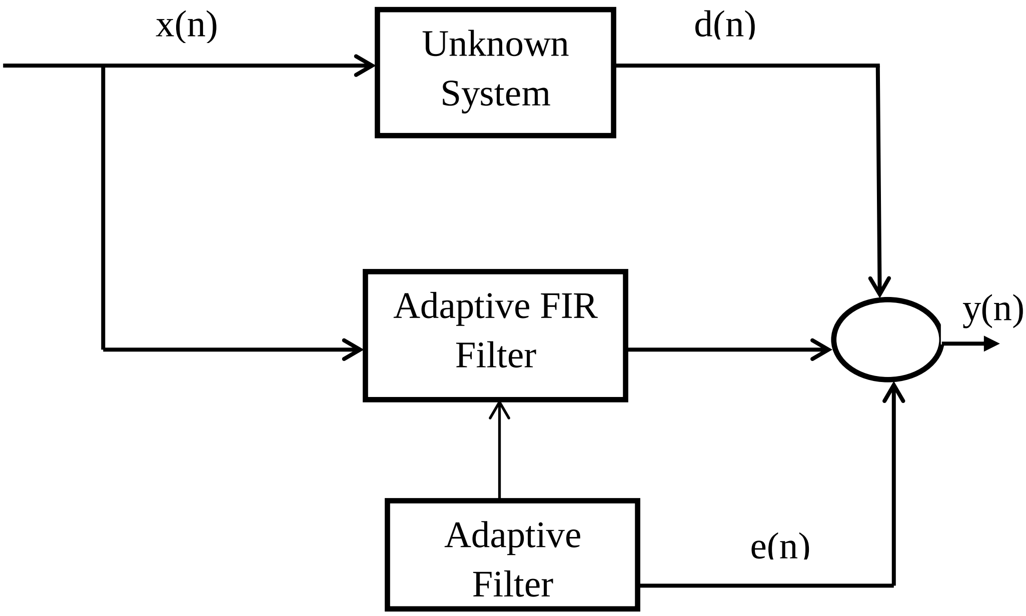


Figure 3.2 Adaptive identification

- ADAPTIVE INVERSE

In the architecture of adaptive inverse as shown in figure. The adaptive digital filter is used to provide the inverse model for an unknown system. The inverse model realizes the reciprocal of the unknown system's transfer function. The combination of the two would then constitutes an ideal transmission medium. Applications that use adaptive inverse include equalization in digital communications, predictive de convolution, blind equalization, adaptive control systems, and others.

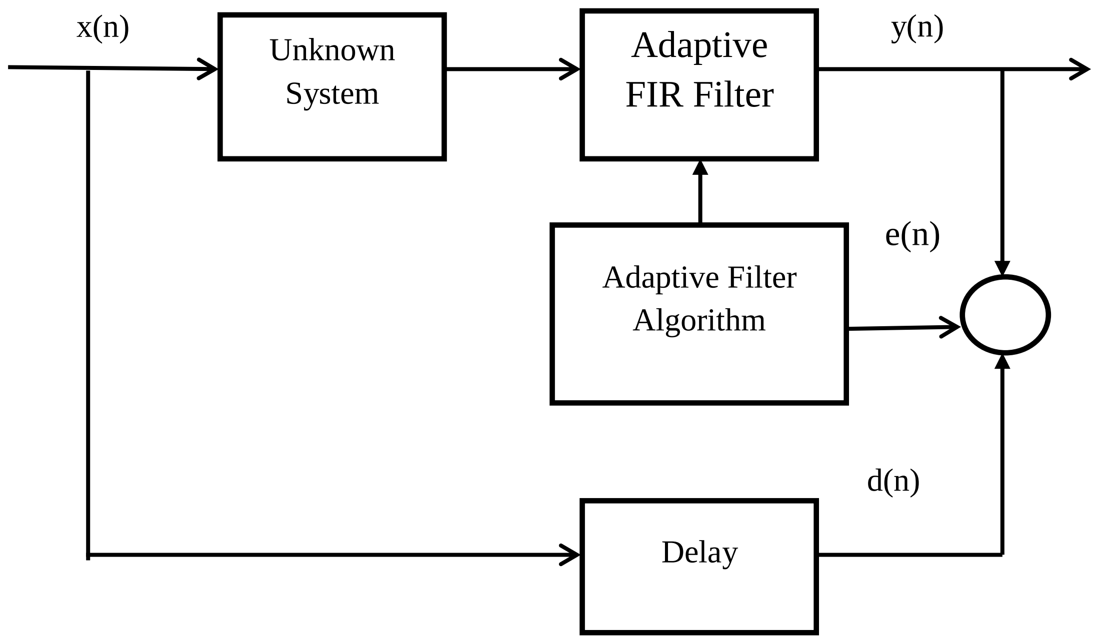


Fig 3.3 Adaptive inverse

- ADAPTIVE PREDICTOR

In the prediction architecture as shown in Figure, the adaptive filter is used to provide a prediction of the value of a random input signal. Depending on the application, the system can operate as a predictor if the output of the adaptive filter predicts the output of the system in advance. However, the system can also operate as a prediction error filter if the prediction error signal is used as the output of the system. Applications of adaptive predictors include predictive noise suppression, periodic signal extraction, linear predictive coding, and others.

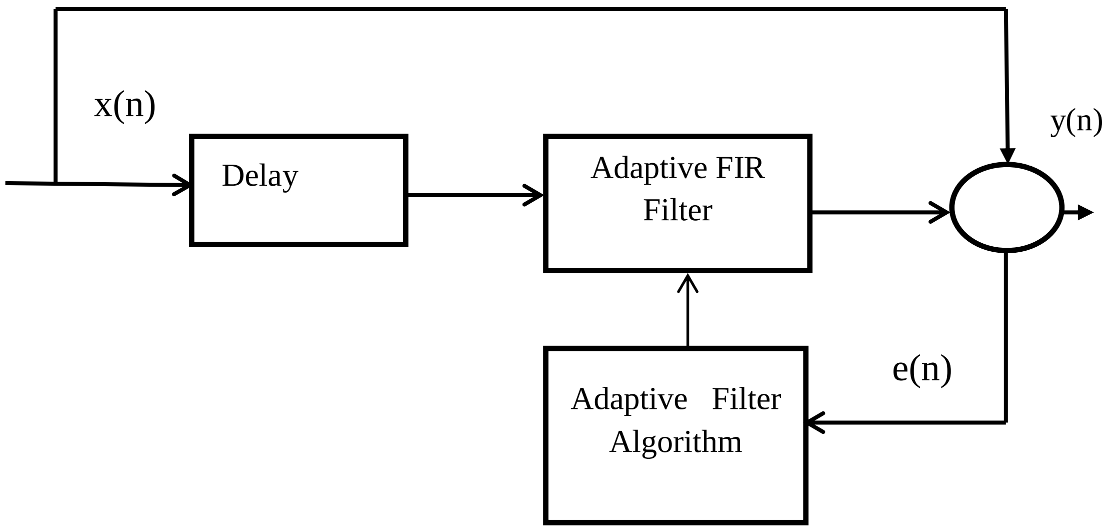


Fig 3.4 Adaptive predictor

- ADAPTIVE NOISE CANCELLATION

Active noise cancellation increases the signal-to-noise ratio of a signal by decreasing the noise power in the signal by attempting to cancel noise signals. Applications consist of adaptive noise cancellation, echo cancellation, adaptive beamforming, biomedical signal processing, and others.

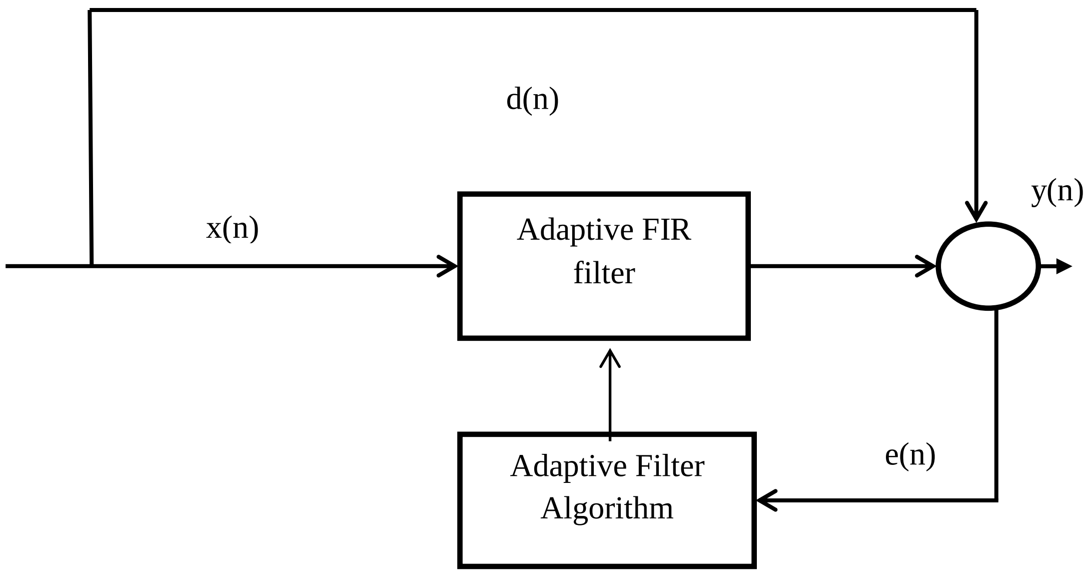


Figure 3.5 Adaptive noise cancellation

3.1.3 High Level Block Diagram

The configuration of the high level block diagram for the system. There are two inputs in the system: reference and interference signals. The reference signal, d(n), contains the target signal and an interference signal. The interference signal, x(n), contains just an interference signal similar to that contained in the reference signal. When the interference signal is passed through the adaptive filter, the output, y(n), is generated so that when it is subtracted from the reference signal the error signal, e(n), is obtained. The error signal is then used to update the coefficients of the filter

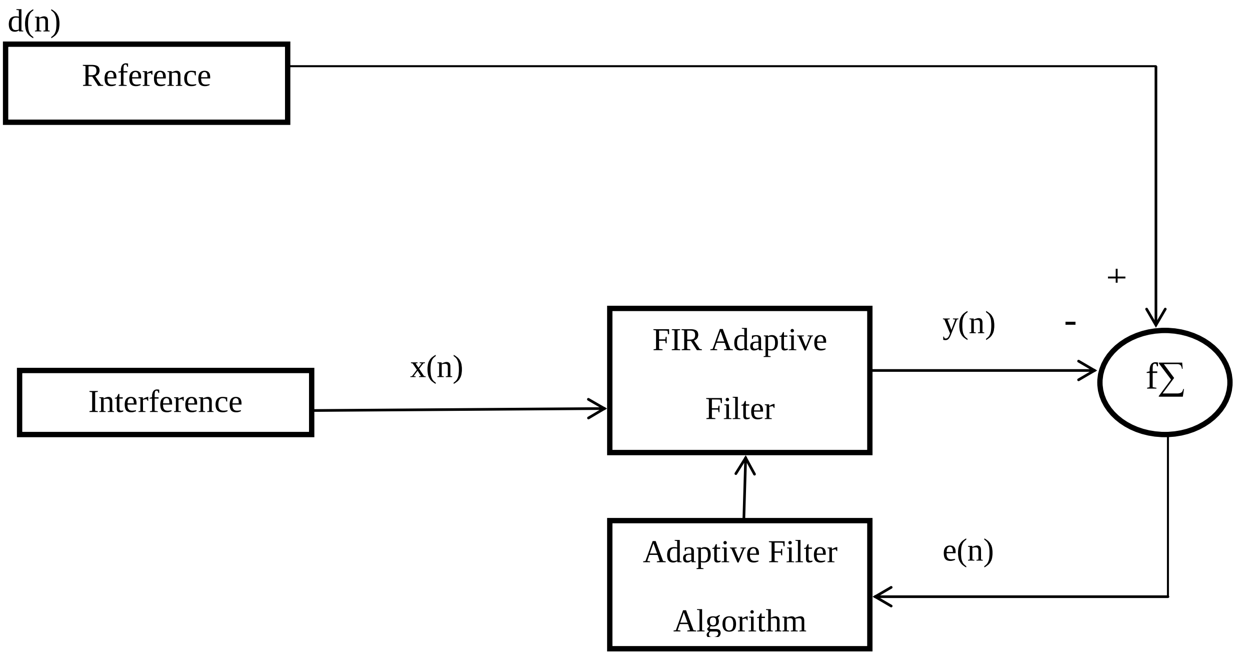


Fig 3.6 High level block diagram of an adaptive filter

3.1.4 Concept of Adaptive Noise Canceling

The input to the adaptive filter is a noise signal w1(n) that is highly correlated with the additive disturbance, w(n), but is uncorrelated with the clean signal s(n). (One can think of w1 (n) as being derived from a sensor located at a point in the noise field where the signal is undetectable.) The reference signal w1 (n) is filtered to produce the output 6(n) that is an estimate of the additive noise w(n). This output is then subtracted from the noisy signal x(n) to produce the system output z(n). The system output is used to control the adaptive filter and is an estimate of s(n). Provided s(n) is uncorrelated with both w1 (n) and w(n), and the adaptive filter is adjusted to give a system output z(n) that has the least possible energy, then z(n) is a best least-squares fit to the clean signal s(n). To prove this, we note that the power in z (n) is given by

$$E(Z^2(Y(n))) = E(S^2(n) + (w(n) - W(N))^2 + 2s(n)(w(n) - \acute{w}(n))) \quad (1)$$

where E denotes expected value. Now since the noise terms and the signal s(n) are assumed uncorrelated,

$$E(Z^2(n)) = E(S^2(n)) + E((w(n) - W(n))^2) \quad (2)$$

Since the signal energy is a fixed quantity for the frame of interest, minimizing the output energy yields

$$\min E(Z^2(n)) = E(S^2(n)) + \min E((w(n) - W(n))^2) \quad (3)$$

Thus, when the noise cancelling filter is adjusted so that E(z(n)) is minimized, $E((w(n) - W(n))^2)$ is also minimized. The filter output S(n) is then a best least-squares estimate of the primary noise w(n). Moreover, when $E((w(n) - W(n))^2)$ is minimized, $E((w(n) - W(n))^2)$ is also minimized since z(n) - s(n) = w(n) - ẃ(n).Thus, z(n) is a best least-squares estimate of the clean signal s(n).

3.1.5 Noise Canceling For Speech Inputs

The success of the adaptive noise cancelling filter is dependent on obtaining an external reference noise input that is uncorrelated with the signal and highly correlated with the additive noise corruption. In most applications, a reference noise signal is not available from a second sensor. To avoid this problem, previous investigators have tried to form a reference noise signal by assuming that the noise is stationary, and that the average signal determined during periods classified (or known to be signal free) as “silence” are representative of the noise. Unfortunately, this approach suffers because the noise is rarely stationary, a finite sample may be insufficient to estimate the noise signal, the silence decision is not error free, and finally, the technique cannot be applied for quantization noise.

Although it may be difficult to form a reference noise input, it is quite easy to form a reference input of the original speech signal. Since speech is quasi-periodic, a section of speech delayed by a small amount (one or two pitch periods) will be highly correlated with the true speech signal s(n) and will be uncorrelated with the additive noise w(n) (provided that w(n) is sufficiently broad band). we can take advantage of the quasi periodic nature of speech for effective noise removal. In this system, we take advantage of the fact that s(n) and s(n - T)are highly correlated, and w(n) and w(n - T) are not correlated with themselves and the speech signal. It can be seen that minimizing the energy in the system output w'(n) will result in a signal s'(n) that is a best least squares fit to s(n).

3.1.6 Technical Methods

- Mathematical Approach

Adaptive filters operate by attempting to reduce a cost function. One of the most popular cost functions to use is known as the Least Square Error equation. It uses the mean square error as the cost function and attempts to reduce the cost function. Various adaptive algorithms can be obtained based on how to minimize the cost function. The cost function (J) can be represented as follows:

$$J = E\{e2\ (n)\} \qquad (4)$$

The error signal of the system can be expressed as:

$$e(n) = d(n) - y(n) = d(n) - f\ T\ X\ (n) \qquad (5)$$

where f is the filter coefficients and X (n) which is a column vector of the filter input signal

The cost function becomes:

$$J = E\{e2\ (n)\} = E\{(d(n) - f\ T \times X\ (n))2\} \qquad (6)$$

$$J = E\{d(n)2 + 2d(n) \times f\ T \times X\ (n) + f\ T \times X\ (n) \times X\ T\ (n) \times f\ \} \qquad (7)$$

By setting the gradient if J equal to zero and solving, for the filter coefficient f, we find that:

$$E\{d(n) \times X(n)\} = E\{X^T(n) \times X(n)\} f_{opt} \quad (8)$$

Solving for the optimum coefficients results in the following equation:

$$Fout = Rxx^{-1}.\mathrm{rr}_{dX^1} \quad (9)$$

3.1.7 Wiener Filtering Algorithm:

Wiener filter theory provides a convenient method of mathematically analysing statistical noise cancelling problems. The Wiener filter is a popular technique that has been used in many signal enhancement methods. The basic principle of the Wiener filter is to obtain estimate of speech signal from that corrupted by additive noise. This estimate is obtained by minimizing the Mean Square Error (MSE) between the desired signal s(n) and the estimated signal s(n). It is based on a statistical approach. The Wiener filter weights noisy signal spectrum according to SNR at different frequencies typical filters are designed for a desired frequency response. However, the design of wiener filter takes a different approach. One is assumed to have knowledge of the spectral properties of the original signal and the noise, and one seeks the linear time invariant filter whose output would come as close to the original signal as possible. For this transfer function of wiener filter is used in frequency domain which is expressed as follows:

$$H(w) = \frac{Ps(w)}{Ps(w) + Pd(w)} \quad (10)$$

Where, Ps (ω) and Pd (ω) are power spectral densities of clean and noisy speech signals respectively. In wiener filter, the speech signal and noise is assumed uncorrelated and stationary, and the SNR is given by

$$SNR = \frac{Ps(w)}{Pd(w)} \quad (11)$$

Using this definition of SNR, the transfer function of Wiener filter can be given

$$H(w) = [1 + \frac{1}{SNR}]^{-1} \quad (12)$$

From the above definition of transfer function, it can be interpreted that the Wiener filter has fixed frequency response at all frequencies and also needs an estimation of the power spectral density of clean signal and noise prior to filtering.

3.1.8 Method of Steepest Descent

To solve the Wiener-Hoff equations for tap weights of the optimum spatial filter, we basically need to compute the inverse of a p-by-p matrix made up of the different values of the autocorrelation function rx(j,k) for j, k = 1, 2, . . . , p. We may avoid the need for this matrix inversion by using the method of steepest descent. According to this method, the weights of the filter assume a time-varying form, and their values are adjusted in an iterative fashion along the error surface with the aim of moving them progressively toward the optimum solution. The method of steepest descent has the task of continually seeking the bottom point of the error surface of the filter. Now it is intuitively reasonable that successive adjustments applied to the tap weights of the filter be in the direction of steepest descent of the error surface, that is, in a direction opposite to the gradient vector whose elements are defined by V,J for k = 1, 2, . . . , p. Such an adjustment is illustrated figure for the case of a single weight. Let wk(n) denote the value of weight wk of the spatial filter calculated at iteration or discrete time n by the method of steepest descent. In a corresponding way, the gradient of the error surface of the filter with respect to this weight takes on a time-varying form of its own.

3.2 THE LMS ALGORITHM

The most commonly-used algorithm to design adaptive linear filter is the least-mean-square(LMS) algorithm originally developed by Widow and Hoff. The LMS algorithm is based on the principle of Minimum Mean square error and the steepest descent algorithms . However, gradient vector, nor does it requires matrix inversion. It is often referred to as the Widow-Hoff rule. The LMS algorithm is used to search for the solving the Wiener-Hoff equation and find the optimal coefficients Wopt for an adaptive filter. The main advantages of the LMS algorithm is its computational simplicity, ease of implementation, unbiased convergence, and the existence of a proof in stationary environment. A block diagram of a typical adaptive noise cancellation system is shown in figure.

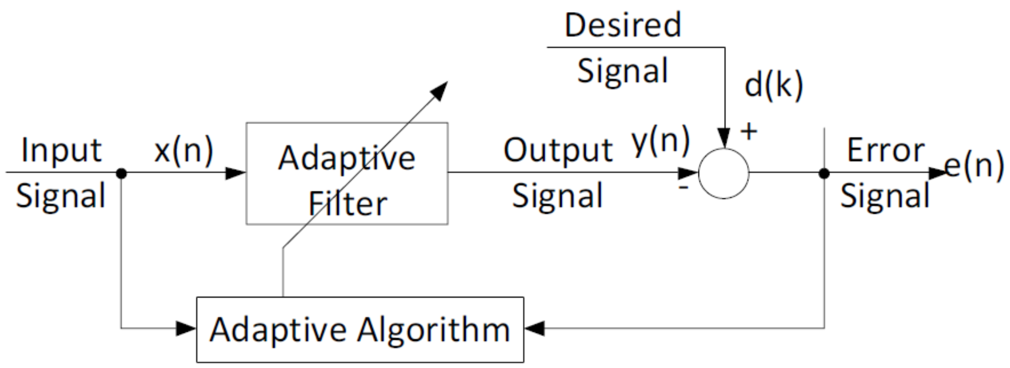


Fig 3.7 Block diagram of a typical adaptive noise cancellation system

The signals shown in above figure are described as follows:

a. The vector X(k) represents the is the input vector of time delayed input values and X(k) is the input at time .

$$\boldsymbol{X}(k) = [x(k) \quad x(k-1) \quad x(k-2) \quad \cdots \quad x(k-N+2) \quad x(k-N+1)]^T \tag{13}$$

b. The vector W(k) is used to represent the weights applied to the filter coefficients at time .

$$\boldsymbol{W}(k) = [W_0(k) \quad W_{1(\mathrm{k})} \quad W_2(k) \quad \cdots \quad W_{N-2}(\mathrm{k}) \quad W_{N-1}(k)]^T \tag{14}$$

c. The parameter μ is the step size of the adaptive filtered .e (k) is the error between the desired response d(k) and the output of the filter y(k), i.e., the filtered signal, at time .The LMS algorithm performs the following operations to update the coefficients of the FIR filter:

1. Calculate the output signal y(k) of the FIR filter. The output of the filter represents an estimate of the desired response. y(k) is the calculated as the convolution of the weight vector and the input vector:

$$y(k) = \sum_{n=0}^{N-1} W_n(\mathrm{k})x(k-n) = W^T(k)x(k) \tag{15}$$

2. The error signal e(k), is estimation error defined as the difference between the estimated response and the desired response.

$$e(k) = d(k)- y(k) \quad (16)$$

3. The error signal and the input signal are applied to the weight update algorithm to updates the filter coefficients. The LMS algorithm updates its coefficients through the minimization of the mean (expectation) of the instantaneous squared error denoted by $E[e^2(k)]$. The LMS algorithm assume that x(k) and d(k) are wide-sense stationary ergodic processes, therefore, their means and variances are constant. X(k) and W(k) are assumed to be independent. As explained in detail, the LMS iterative weight update algorithm follows the following equation.

$$W(k+1) = W(k) + 2\mu e(k)X(k) \quad (17)$$

The step size parameter μ is a small positive constant. The selection of value of the step size highly influences the updating of the filter coefficients and has a major impact on the performance of the LMS algorithm. The smaller the selected value for , the larger the time it takes for the adaptive filter to converge to the optimal solution. However, selecting a large value for may cause the algorithm to be unstable and the output to diverge. For stable behaviour and convergence of the LMS algorithm, the step size must be a small positive value (μ<<1) and

$$0 < \mu < \frac{1}{2 \cdot N \cdot \boldsymbol{R}}$$

Where N is the number of taps of the filter and R is the input signal covariance matrix defined as

$$R = E(X(n)X^T(\mathrm{n}) \quad (18)$$

3.2.1 Selection of Adaptive Parameters

The choice of the step-size parameter and the order of the filter effectively determines the performance of LMS. From above equation the range of μis known but how can the value of μbe exactly chosen and how can the number of filter taps be chosen. when the filter taps are increased, this improves the convergent performance of LMS algorithm, but every tap (in structure of LMS adaptive filter) costs two more multipliers and two more adders ,as seen in figure. However, this will increase the area needed and decrease the maximum frequency of the design. So, balance is required between the convergent performance and the amount of hardware used effectively. Unfortunately, there is no clear mathematical analysis to derive the exact quantities. Only through experiments may a reasonable solution be obtained. In order to select appropriate step size and filter order, MATLAB simulation of LMS algorithm is carried out. Based on the simulation results (which will be discussed later), the adaptive parameters obtained will be applied to the hardware implementation process of LMS algorithm.

3.2.2 The LMS Algorithm Variations

Many variations for the LMS algorithm have been reported in the literature. Some of these variations were developed overcome the shortcoming of technology at their time of development such as long multiplication times. Other variations were developed to improve the tracking ability and the speed of convergence of the algorithm. These variations include:

1. The Normalized Least-Mean-Squares (NLMS) algorithm implements a normalized variation of the LMS algorithm. In this algorithm, the step size is time-varying parameter that increases or decreases as the mean-square error increases or decreases. In each iteration the step size is modified as follows

$$\mu = \frac{\mu}{X^T(n)X(n) + \in} \tag{19}$$

Where € is a small constant used to prevent the divergence of the algorithm. When $X^T(n)x(n)$ is very small. This allows the adaptive filter to better track changes in the system as well as produce small adjustment error. In many cases, the NLMS algorithm has better stability and accurate tracking capabilities than the LMS and converges more quickly with fewer iteration and fewer samples.

The sign-Data Least-Mean-Squares (SDLMS) modifies the weights applied to the filter coefficients at each iteration based on the sign of the input data x(n). In vector form, the SDLMS algorithm can be defined as

$$w(k+1) = w(k) + \mu\ e(k)\ sgn[x(k)], sgn[x(k)] = \begin{cases} 1 & x(k) > 0 \\ 0 & x(k) = 0 \\ -1 & x(k) < 0 \end{cases} \quad (20)$$

The SDLMS algorithm reduces the amount of computation needed for convergence. However, poorly selected initial conditions may result in unbounded error signal, thus causing the algorithm to be unstable. To ensure the SDLMS algorithm stability and thus convergence, the initial conditions must be set to nonzero values, and step size μ must be small (μ<<1). The practical bounds for μ are

$$0 < \mu < \frac{1}{\mathrm{N\{Input\ Signal\ Power\}}}$$

The sign data (or regressor) LMS was first developed to reduce the number of multiplications required by the LMS. The step size, μ, is carefully chosen to be a power of two and only bit shifting multiplies are required and the sign of the error only is used. The Sign-Error Least-Mean -Squares (SELMS) algorithm modifies the weights applied to the filter coefficients at each iteration based on the sign of the error, e(n). In vector form, the SELMS algorithm can be defined as

$$w(k+1) = w(k) + \mu\ e(n)\ sgn[x(k)], sgn[e(k)] = \begin{cases} 1 & e(k) > 0 \\ 0 & e(k) = 0 \\ -1 & e(k) < 0 \end{cases} \quad (21)$$

The sign error LMS was first developed to reduce the number of multiplications required by the LMS. The step size, μ, should be chosen to be a power of two so that only bit shifting multiplies are required and the sign of the data only is used. The Delayed-LMS (DLMS) algorithm a pipelined variation of the LMS algorithm. It uses registers in the filter and error feedback paths. The DLMS updates its coefficients according to the following equation:

$$\boldsymbol{W}(k+1) = \boldsymbol{W}(k) + 2\mu\, e(k-D)\boldsymbol{X}(k-D) \quad (22)$$

Where D represent the delays in the pipelined architecture. The DLMS-based adaptive filter is faster than the LMS-based filter. However, it needs more logic resources to implement its pipelined architecture.

3.3 SOFTWARE REQUIREMENTS

3.3.1 Xilinx 14.7

Xilinx ISE (Integrated Synthesis Environment) is a software tool produced by Xilinx for synthesis and analysis of HDL designs, enabling the developer to synthesize ("compile") their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer.

Xilinx ISE is a design environment for FPGA products from Xilinx, and is tightly-coupled to the architecture of such chips, and cannot be used with FPGA products from other vendors. The Xilinx ISE is primarily used for circuit synthesis and design, while ISIM or the ModelSim logic simulator is used for system-level testing.

System-level testing may be performed with ISIM or the ModelSim logic simulator, and such test programs must also be written in HDL languages. Test bench programs may include simulated input signal waveforms, or monitors which observe and verify the outputs of the device under test.

ModelSim or ISIM may be used to perform the following types of simulations:

- Logical verification, to ensure the module produces expected results
- Behavioural verification, to verify logical and timing issues
- Post-place & route simulation, to verify behaviour after placement of the module within the reconfigurable logic of the FPGA

- Xilinx's patented algorithms for synthesis allow designs to run up to 30% faster than competing programs, and allows greater logic density which reduces project time and costs.
- Also, due to the increasing complexity of FPGA fabric, including memory blocks and I/O blocks, more complex synthesis algorithms were developed that separate unrelated modules into slices, reducing post-placement errors.
- IP Cores are offered by Xilinx and other third-party vendors, to implement system-level functions such as digital signal processing (DSP), bus interfaces, networking protocols, image processing, embedded processors, and peripherals. Xilinx has been instrumental in shifting designs from ASIC-based implementation to FPGA-based implementation.

3.3.2 ModelSimSE 6.2c

ModelSim is a multi-language HDL simulation environment by Mentor Graphics, for simulation of hardware description languages such as VHDL, Verilog and SystemC, and includes a built-in C debugger. ModelSim can be used independently, or in conjunction with Altera Quartus or Xilinx ISE. Simulation is performed using the graphical user interface (GUI), or automatically using scripts.

ModelSim SE offers high-performance and advanced debugging capabilities, while ModelSim PE is the entry-level simulator for hobbyists and students. ModelSim SE is used in large multi-million gate designs.

ModelSim uses a unified kernel for simulation of all supported languages, and the method of debugging embedded C code is the same as VHDL or Verilog.

ModelSim enables simulation, verification and debugging for the following languages:

- VHDL
- Verilog
- Verilog 2001
- SystemVerilog

- PSL
- SystemC

3.3.3 Mat Lab

MATLAB is a programming language developed by Math Works. It started out as a matrix programming language where linear algebra programming was simple. It can be run both under interactive sessions and as a batch job.

It allows matrix manipulations; plotting of functions and data; implementation of algorithms; creation of user interfaces; interfacing with programs written in other languages, including C, C++, Java, and FORTRAN; analyze data; develop algorithms; and create models and applications. It has numerous built-in commands and math functions that help you in mathematical calculations, generating plots, and performing numerical methods

MATLAB is widely used as a computational tool in science and engineering encompassing the fields of physics, chemistry, math and all engineering streams. It is used in a range of applications including:

- signal processing and Communications
- image and video Processing
- control systems
- test and measurement
- computational finance
- computational biology

Following are the basic features of MATLAB:

- It is a high-level language for numerical computation, visualization and application development.
- It provides built-in graphics for visualizing data and tools for creating custom plots.
- MATLAB's programming interface gives development tools for improving code quality, maintainability, and maximizing performance.
- It provides tools for building applications with custom graphical interfaces.

3.4HARDWARE REQUIREMENTS

3.4.1 FPGA

A field-programmable gate array (FPGA) is an integrated circuit designed to be configured by a customer or a designer after manufacturing – hence "field-programmable". The FPGA configuration is generally specified using a hardware description language (HDL), similar to that used for an application-specific integrated circuit (ASIC).

FPGAs contain an array of programmable logic blocks, and a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together", like many logic gates that can be inter-wired in different configurations. Logic blocks can be configured to perform complex combinational functions, or merely simple logic gates like AND and XOR. In most FPGAs, logic blocks also include memory elements, which may be simple flip-flops or more complete blocks of memory.

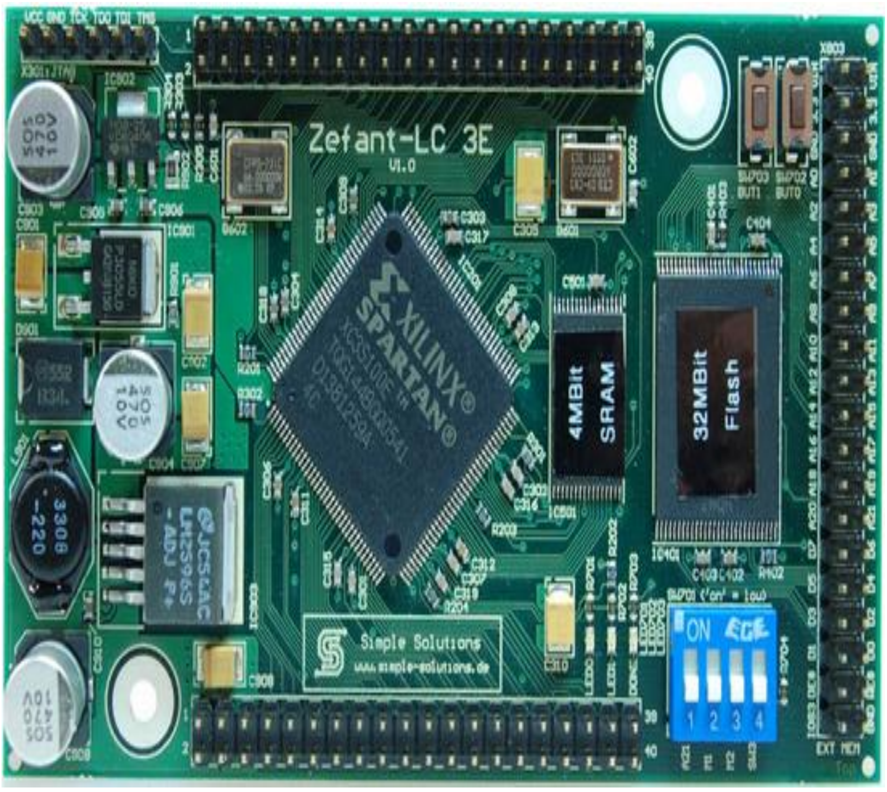


Fig.No.3.8 FPGA

3.4.2 MAX232

The MAX232 is an integrated circuit first created in 1987 by Maxim Integrated Products that converts signals from a TIA-232 (RS-232) serial port to signals suitable for use in TTL-compatible digital logic circuits. The MAX232 is a dual transmitter / dual receiver that typically is used to convert the RX, TX, CTS, RTS signals.

The drivers provide TIA-232 voltage level outputs (about ±7.5 volts) from a single 5-volt supply by on-chip charge pumps and external capacitors. This makes it useful for implementing TIA-232 in devices that otherwise do not need any other voltages.

Fig.No.3.9 MAX 232

The receivers reduce TIA-232 inputs, which may be as high as ±25 volts, to standard 5 volt TTL levels. These receivers have a typical threshold of 1.3 volts and a typical hysteresis of 0.5 volts. The MAX232 replaced an older pair of chips MC1488 and MC1489 that performed similar RS-232 translation. The MC1488 quad transmitter chip required 12 volt and -12 volt power,[1] and MC1489 quad receiver chip required 5 volt power.[2] The main disadvantages of this older solution was the +/- 12 volt power requirement, only supported 5 volt digital logic, and two chips instead of one.

3.4.3 Power Supply

The power supply is the most indispensable part of any project. IC regulators are versatile and relatively inexpensive and are available with features such as current/voltage boosting, internal short circuit current limiting, thermal shutdown and floating operation for high voltage applications. The regulated circuit is used to maintain constant output level. The integrated circuit regulator, sometimes called the three terminal regulators contains the circuitry for reference source error amplitude control device and overload protection all in a single IC chip. The regulator IC here used isL7805. It provides regulated 5V to the controller. Its maximum input voltage is 35V and minimum voltage is 8V. Output is constant 5V. The L7800 series of three-terminal positive regulators is available in TO-220, TO-220FP, TO-3 and D2PAK package And several fixed output voltages, making it useful in a wide range of applications. These regulators can provide local on-card regulation, eliminating the distribution problems associated with single point regulation. Each type employs internal current limiting, thermal shut-down and safe area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.

Also uses LM1117-1.2V, LM117-2.5V, LM117-3.3V.

Different packages of L7805

Features

- output current up to 1.5 A
- Thermal overload protection
- hot circuit protection
- Output transition SOA protection
- output voltages of 5; 5.2; 6; 8; 8.5; 9; 12; 15; 18; 24V
- Maximum input voltage =35V (for Vo=5 to 18V),

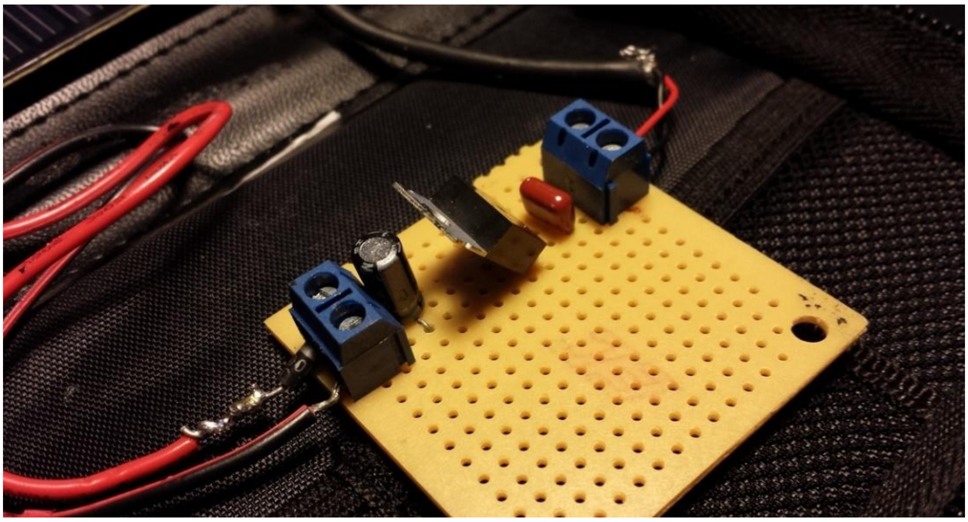


Fig.No.3.10 Power Supply

3.4.4 Push Button

The most familiar form of switch is a manually operated electromechanical device with one or more sets of electrical contacts, which are connected to external circuits. Each set of contacts can be in one of two states: either closed means the contacts are touching and electricity can flow between them, or open, means the contacts are separated and the switch is non-conducting. The mechanism actuating the transition between these two states (open or closed) can be either a toggle (flip switch for continuous on or off) or momentary (push-for on or push-for off type).A push button switch is a momentary or non-latching which causes a temporary change in the state of an electrical circuit only while the switch is physically actuated. An automatic mechanism (i.e. a spring) returns the switch to its default position immediately afterwards, restoring the initial circuit condition. There are two types:

A Push to Make switch allows electricity to flow between its two contacts when held in. When the button is released, the circuit is broken. This type of switch is also known as a Normally Open (NO) Switch. Examples: doorbell, computer case power switch, calculator buttons, individual keys on a keyboard)

Fig.No.3.11 Push Button

3.4.5 LCD Display

Most common LCD's connected to the microcontrollers are 16x2 and 20x2 displays. This means 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively. The standard is referred to as HD44780U, which refers to the controller chip which receives data from an external source and communicates directly with the LCD.

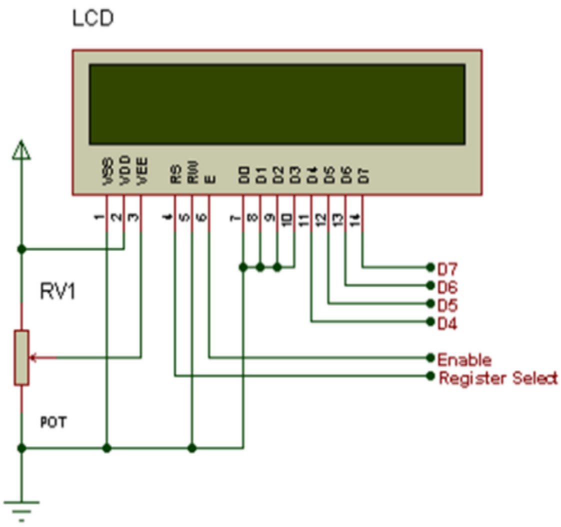


Fig.No.3.12 LCD Display

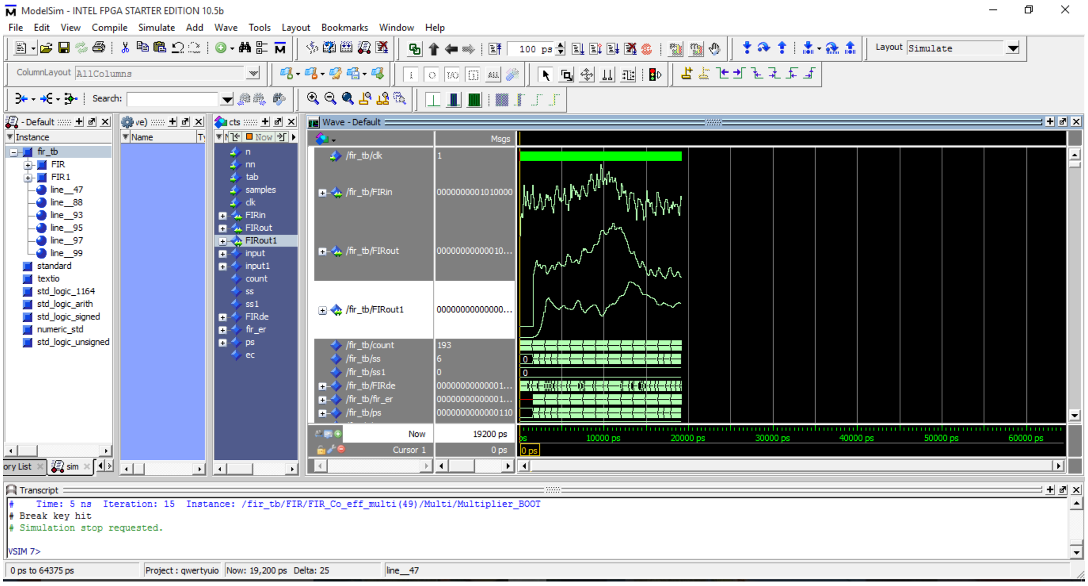
CHAPTER 4

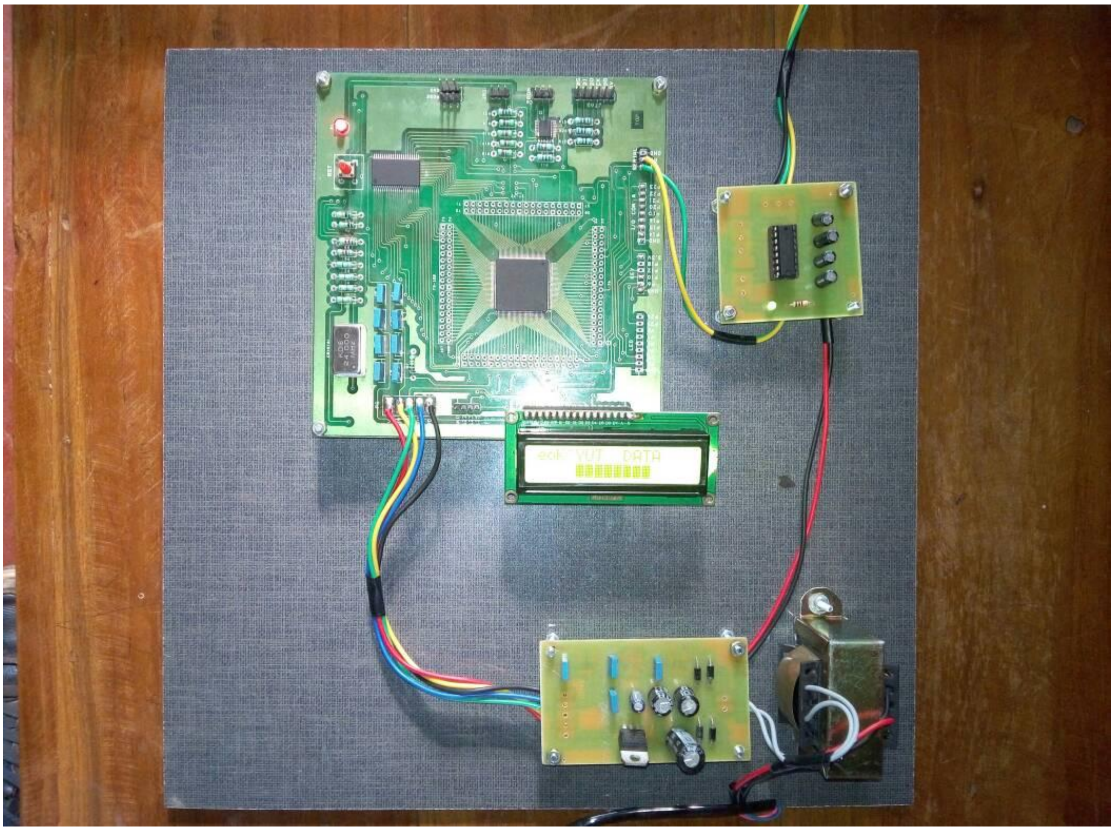
ADVANTAGES AND APPLICATIONS

- Cancelling 60-Hz Interference in Electrocardiography.
- Cancelling the Donor ECG in Heart-Transplant Electrocardiography.
- Cancelling the Maternal ECG in Fetal Electrocardiography.
- Cancelling Noise in Speech Signal.
- Cancelling Antenna Sidelobe Interference
- Adaptive Self-Tuning Filter.
- Cancelling Periodic Interference without an External Reference Source . Cancelling Periodic Interference without an External Reference Source.
- Adaptive Self-Tuning Filter.

CHAPTER 5

RESULT





The bit file generated using Hardware Co-Simulation feature of SysGen for execution on the target board is downloaded onto the board via JTAG interface. The FPGA response is again taken back to Simulink via JTAG, where it is displayed. The output of the adaptive filter and the error signal simulated using Matlab/Simulink and Xilinx block set are shown in and respectively. The output from the Virtex-5 is shown in . The hardware is in good agreement with the simulated responses. The hardware response shows the convergence of the error signal to a minimum value in accordance to the LMS algorithm.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

4.1 CONCLUSION

An active noise cancellation system was successfully simulated and implemented. The system met all requirements. The ultra-sound data was properly filtered, and the audio data was adequately filtered to recover the target signal. These requirements were accomplished in simulation.

4.2 FUTURE SCOPE

In this semester we proposed many topics, in those topics adaptive noise cancellation using LMS algorithm was selected and we simulated the algorithm using matlab. Later we will simulate the algorithm by VHDL or VERILOG. Hard ware implementation is done by FPGA. In this semester we are using only LMS, in next semester for better seek we are planning to use RLS.

REFERENCES

[1] L.J.Eriksson, M.C.Allie, and C.D.Bremigan," Active Noise Control using Adaptive digital signal processing," *in proc ICASSP*, New York,2004.

[2] C.Mosquera, J.A. Gomez "Adaptive filter for active noise control", *Sixth international conference on sound and vibration Copenhagen*, Denmark.

[3] Colin H.Hansen "Understanding Active Noise Cancellation "*IOS Press-2002.*

[4] *Sayed A.H. and lotfizad M. (2010) International Journal of Computer and Electrical Engineering, 2(2)*.

[5] Górriz J.M., Javier R., Cruces-Alvarez S., Carlos G.P., Elmar W. L., Deniz E. (2009) *IEEE Signal Processing Letters*, 16(1).

[6] Karni S. and Zeng G. (1989) *IEEE Trans. Circuits Syst*., 36(7), 1011-1012.