An Effective Implementation of Noise Cancellation for Audio Enhancement using Adaptive Filtering Algorithm

Hazel Alwin Philbert
Department of Electronics and Communication Engineering
Gogte Institute of Technology, Belgaum, Karnataka, India

Prof. S.P.Deshpande
Department of Electronics and Communication Engineering
Gogte Institute of Technology, Belgaum, Karnataka, India

Abstract- With innovation in technology there are many active noise cancellation systems. Researchers try to modify the performance of the system by different noise cancellation techniques. One of the techniques for noise cancellation is the adaptive filtering technique. An adaptive filter is one whose characteristics can be modified automatically to achieve an objective.

This paper deals with noise cancellation for speech signals using Least Mean Square (LMS) algorithm for the adaptation of the filter coefficients. The main aim of noise cancellation is to obtain a noise free signal by estimating the noise signal and subtracting it from the primary input. The simplicity of the algorithm provides better potential of incorporating existing performance enhancing techniques.

The system is implemented in Verilog design and the simulation results obtained by the Xilinx synthesis tool are noted down and analyzed. The simulation results show analysis on the basis of the Signal to Noise Ratio (SNR), the Mean Square error (MSE) and the learning curve. The experimental results show an improvement in the performance of the proposed method.

Keywords – Adaptive filter, LMS Algorithm, MSE, SNR, Learning Curve

I. INTRODUCTION

In today's modern world we are surrounded by different kinds of signals which are available in various forms. Some signals are necessary and are pleasant, while many are unnecessary and unwanted. These useful and unwanted signals carry information.

Since signal processing deals with enhancing, extracting, storing and transmission of useful information, Adaptive Filtering plays an important role in signal processing. Adaptive Filters provides a solution over conventional filtering techniques, when there is no prior knowledge of the received signal. Adaptive filters are filters that adjust its filter coefficients so that the quality of the filter output is enhanced. Adaptive filters are used in communication systems for a number of applications such as noise cancellation, echo cancellation, equalization and speech compression.

In this paper we deal with adaptive filters for noise cancellation of real time audio processing system. Adaptive noise cancellation (ANC) is a technique of estimating noise or interference in a corrupted signal by passing it through an adaptive filter and improving the Signal to Noise ratio (SNR).The ANC system should meet the following requirements 1) Minimum Mean Square Error (MSE) 2) Computational Complexity 3) Stability and Robustness.

The most practical and widely used adaptive FIR feed forward system is used, for removal of noise without affecting the received signal. It consists of two inputs, one is the wanted signal s(n) corrupted by noise n(n) called the primary signal d(n) and the other is the reference signal x(n),measured at the noise source. When the reference signal is passed through the adaptive filter it produces a system output signal y(n) with an inverted sign. The error signal e(n) is the superimposition of this signal with the primary signal so that a noise free output can be obtained.

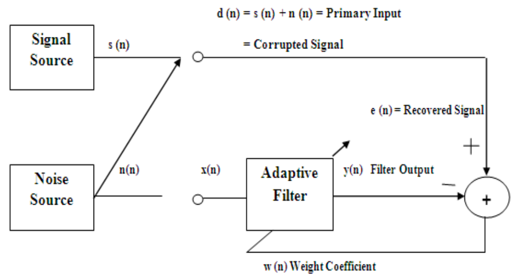

Figure1 Feed-forward Adaptive Noise Canceller

s(n) = signal source
n(n) = noise signal
x(n) = noise reference input
d(n) = primary signal
y(n) = output of adaptive filter
e(n) = system output signal

Among all the other algorithms, the Least Mean Square (LMS) algorithm is most suitable for the adaption of the filter coefficients. Thus, the objective of operating under changing conditions and readjust itself continuously to minimize the error is fulfilled by the LMS algorithm. This process is called as an adaptive process, which means no prior knowledge of the characteristics of the signal or noise are required. The LMS algorithm does not make use of an inverse matrix nor does it require the measurement of pertinent correlation functions.[4] Thus the special feature of the LMS algorithm is its simplicity and reliability. This special feature has made the LMS algorithm a benchmark in all the adaptive algorithms.
The rest of the paper is organized as follows .The section II consists of the proposed design and model. Section III presents the simulation results obtained and in section IV the concluding remarks are stated.

II.PROPOSED ASPECTS AND MODEL

The adaptive algorithm must be chosen keeping in mind the following factors of rate of convergence, minimum MSE, stability and computational complexity. Therefore the ANC system proposed using the adaptive algorithm for noise cancellation should converge at a faster rate with a small MSE indicating that the system has accurately modeled and adapted to a solution for the system. Out of the many adaptive filtering algorithms like the LMS, RLS (Recursive Least Square) algorithms we chose the LMS algorithm to obtain the desired results because of its simplicity and low computational complexity.

A. Least Mean Square Algorithm

The Least Mean Square (LMS) algorithm, introduced by Widrow and Hoff in 1960 is an adaptive algorithm which uses a gradient based method of steepest decent. It estimates gradient vector from the available data and incorporates an iterative procedure that makes successful corrections to the weight vector, which gradually leads to minimum mean square error.
The LMS algorithm consists of the following processes:
1] The filtering process, which computes the output of a linear filter produced by a set of input signals and generating an estimation error by comparing the output with the desired response.
2] An adaptive process, which automatically adjusts the co=efficients of the filter with the estimated error. [4]
A step size parameter μ is used to control the convergence speed and the stability of the algorithm. If μ is chosen to be very large then the algorithm converges very fast where as if the μ value is small the algorithm converges slowly but with a better stability.
The upper bound for μ so that the algorithm is convergent in mean square, if μ satisfies the condition

$$0 < \mu < 1/(M \times P) \tag{1}$$

B. Mathematical Treatment

Consider a filter with input x(n), i.e. a vector with N most recent input samples at a sampling point n given by

$$x(n) = [x(n), x(n-1), \ldots, x(n-N-1)] \tag{2}$$

and w(n) , i.e. vector of filter coefficients as

$$w(n) = [w_0(n), w_1(n), \ldots\ldots, w_{N-1}(n)] \tag{3}$$

At some discrete time n the filter produces the output y(n) given by

$$y(n) = \sum_{i=0}^{N-1} wi\, x(n-i) = w'x(n) = x'(n)w \tag{4}$$

Subsequently the error signal can be written as

$$e(n) = d(n) - y(n) = d(n) - w'x(n) = d(n) - x'(n)w \tag{5}$$

Where d(n) is the primary signal given by

$$d(n) = s(n) + n(n) \tag{6}$$

The cost function is then given by

$$\xi = E[e(n)^2] = E[d^2(n)] - E[d(n)\,x'(n)]w - w'E[d(n)x(n)] + w'E[d(n)x'(n)]\,w$$

$$= E[d^2(n)] - 2p'w + w'Rw \tag{7}$$

The gradient vector can also be expressed as

$$\nabla\xi = 0 \tag{8}$$

where, ∇ is the gradient operator

The partial derivatives of ξ with respect to the filter tap weights can be solved such that

$$\nabla\xi = 2Rw(n) - 2p$$

where, R is the autocorrelation matrix of x(n) and p is the cross correlation matrix of d(n) and x(n).

To find an optimal solution $w_{0,}$ (initial guess) that satisfies the condition

$$\xi(w_0) \le \xi(w) \qquad \text{for all } w \tag{9}$$

this is a mathematical statement of unconstrained optimization.

Starting with w(0),generate a sequence of weight vector w(1),w(2),….,such that the cost function ξ (w) is reduced at each iteration of the algorithm therefore

$$\xi(w(n+1)) < \xi(w(n)) \tag{10}$$

where, w(n) is the old value of the weight vector and w(n+1) is the updated value of the weight vector.

Thus a new recursion relation can be obtained in order to update the weight vector and is given by

$$w(n+1) = w(n) - \mu\nabla e^2(n) \tag{11}$$

Solving for $\nabla e^2(n)$ and substituting in the above equation we obtain the tap weight adaption formula for the LMS algorithm as

$$w(n+1) = w(n) - \mu\, x(n)e(n) \tag{12}$$

Thus during every iteration, the tap weight adaptation updates the tap weight vector w in the direction of minimizing the cost function so that it has converged to an optimal solution.

C. Verilog Implementation of the System

The Verilog implementation of the system is shown in the figure 2.The system is implemented using floating point arithmetic.

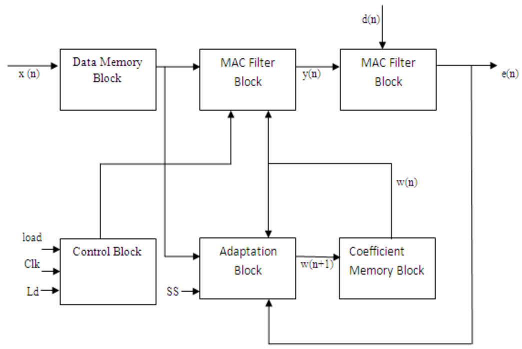

Figure 2 Verilog implementation of the adaptive filter

The design splits into six blocks as follows:
1] The Data Memory block:
Here a single port RAM is used for storage of the audio samples. The noise sample and the source sample (original signal) are stored in the data RAM. When the clock input is applied the data gets transferred to the MAC filter block and is available for adaptation. A load signal is used to load the content of the data RAM.

2] MAC filter block:

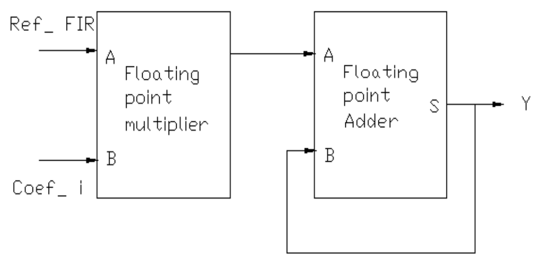

Figure3. FIR filter block

A two stage pipe line is used for the filtering cycle in this block. The data (input samples) read from the RAM block are multiplied with their corresponding filter coefficients taken from the dual-ported coefficient RAM block and is stored in the accumulator.
The transposed direct form of the FIR filter design is used to keep maximum data path length short. The filter which is a part of a sequential MAC unit performs M accumulations of products at every sample period so that a resource sharing can be utilized. No parallel structure with M multipliers and M-1 adders are necessary, because they audio sample period f_s provide a large amount of available clock cycles per audio cycle. [1]

3] The System Output Block:
The system output block performs the subtraction of the primary signal d(n) and the saturation block output y(n) to produce an error signal e(n). The signal e(n) is the required system output and is the estimate of the wanted signal in the system output.

4] Adaptation Block Algorithm:

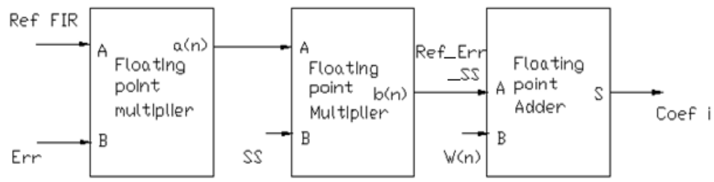

Figure 4 Coefficient Adaptation Unit

This block is designed as a three stage pipelined structure for updating the co-efficient value, for each iteration. It uses the tap weight adaptation formula of the LMS algorithm, so as to minimize the cost function and converge to an optimum solution.
The co-efficient is calculated by first taking a product of the input sample x(n) with the error signal e(n) and is termed as a(n).This result is then multiplied with the step size parameter (μ). An adder block is used to add the multiplied result with the previous filter coefficient to get the updated value of the weight vector.

5] Coefficient Memory Block:
This block stores the current filter coefficients .The dual port RAM is used for the parallel processing of the FIR filter block and the coefficient update block.

6] Control Block:
The control block functionality controls the operation of the adaptive filter. It consists of three control signals namely clock, load and ld. During the positive edge transition of the clock signal a new output is available. The load input enables the operations of the adaptive filter. When the ld signal is high the filter is updated with a new value of the weight vector.

III. EXPERIMENT AND RESULTS

Xilinx 14.2 development environment is used for implementing the proposed Verilog design of the LMS algorithm. The design is written in Verilog and simulated using ISE simulator. A data RAM is created for testing the system. A combination of the original signal and noise signal is used as the primary input to the filter. A reference signal is the noise signal which is correlated version of noise added with original signal. An error signal is produced as the system output, which is the recovered signal.
Figure 5 shows the recovered signal output denoted by e(n). Floating point arithmetic is used for arithmetic computations. Figure 6 shows the data Ram output. The Figure 7 shows the RTL synthesis for the proposed design. The synthesis report shows a total gate delay of 29.639 ns.
A series of tests were conducted to evaluate the performance of our proposed ANC system. The simulation results obtained show the variation in the number of iterations versus the error and the SNR for two different cases.

Using the proposed method for 400 iterations the following results were obtained.

Table 1: Variation of the step size factor for 400 iterations

Step Size	Mean Square Error	Signal to noise ratio (db)
0.02	0.1526	4.9999
0.05	0.1508	5.7194
0.075	0.1426	6.1376

Using the proposed method for 800 iterations the following results were obtained.

Table 2: Variation of the step size factor for 800 iterations

Step Size	Mean Square Error	Signal to noise ratio (db)
0.02	0.1465	5.6779
0.05	0.1341	6.1841
0.075	0.1342	6.4881

Table 1 and table 2 summarize the dependencies of the step size μ for different iterations. In figure 9 with a small step size or a convergence factor of 0.05 the convergence rate is quite slow. It has converged at 175 samples. Whereas figure 11 shows the effect of the step size on the convergence when μ= 0.075.A moderate increment in the step size leads to faster convergence rate. The algorithm has adapted itself at a convergence rate of 150 samples. The timing summary of the implemented Verilog design is given below

Timing Summary:

Speed Grade: -4

Minimum period: 29.639ns

Minimum input arrival time before clock: 4.546ns

Maximum output required time after clock: 4.310ns

Maximum combinational path delay: No path found.

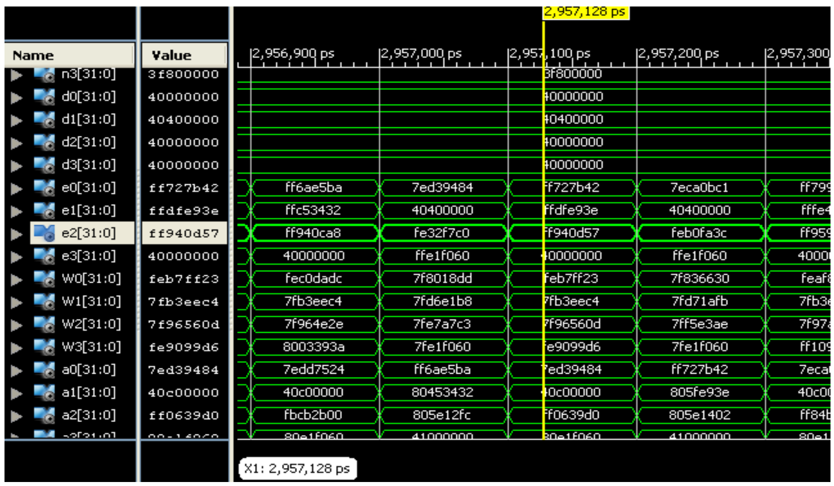

Figure 5: Output of the error signal e(n) and the updated weight vector W(n)

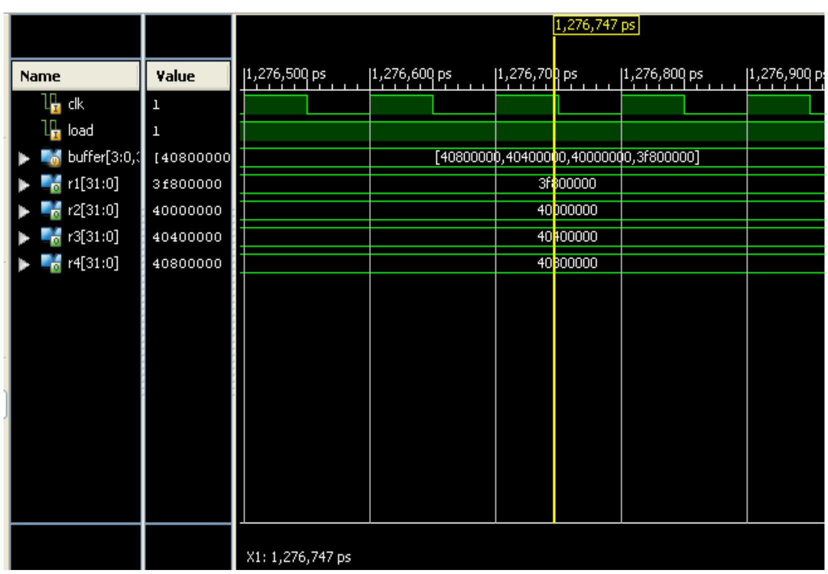

Figure 6: Output for the transfer of data from data RAM

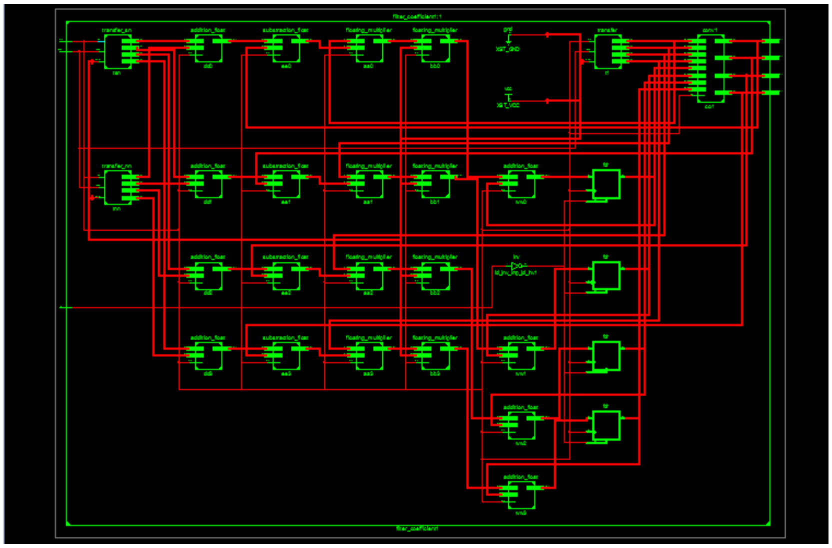

Figure 7: RTL Schematic for the proposed Verilog Design

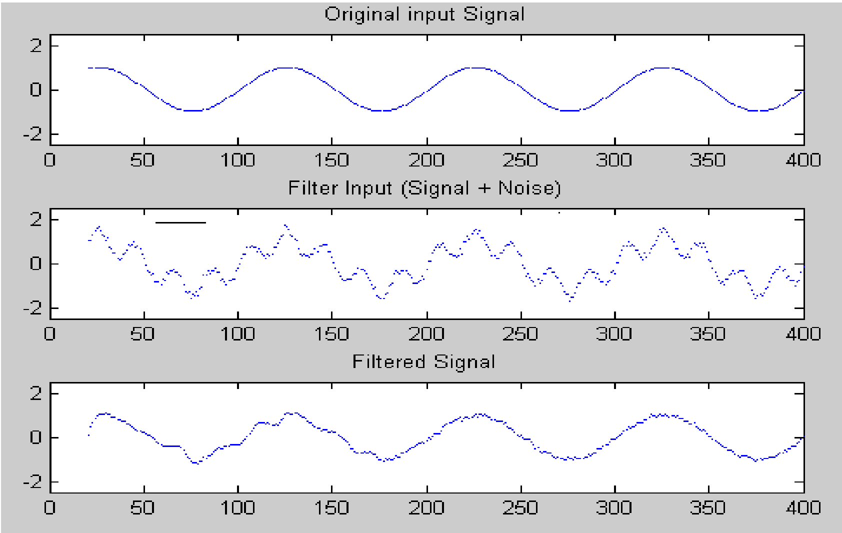

Figure 8: Original and filtered signal for 400 iterations and μ=0.05

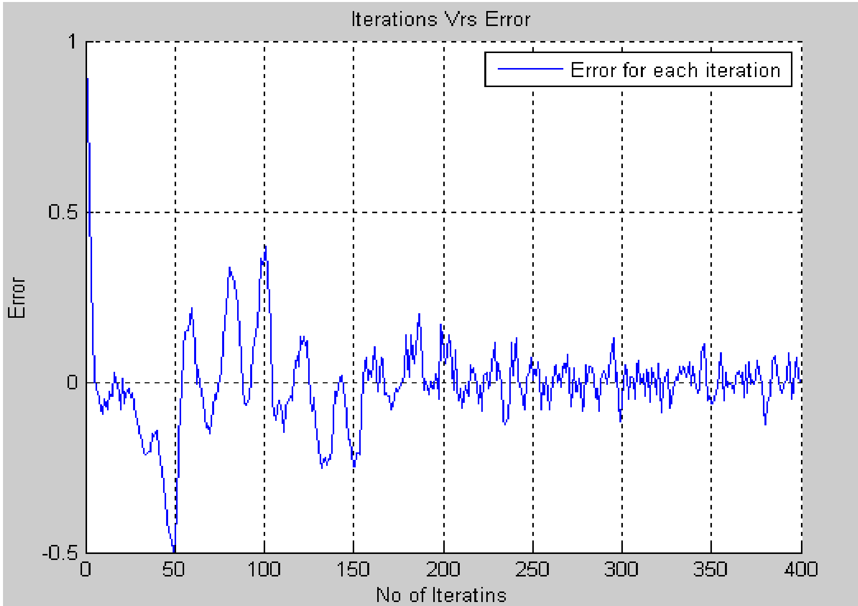

Figure 9: Learning Curve for 400 iterations and for μ=0.05

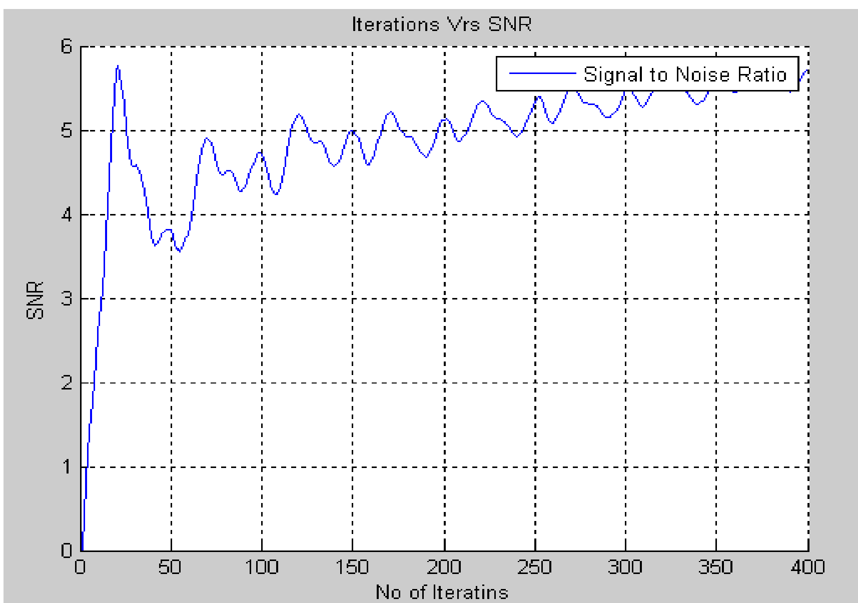

Figure 10: SNR for 400 iterations and for μ=0.05

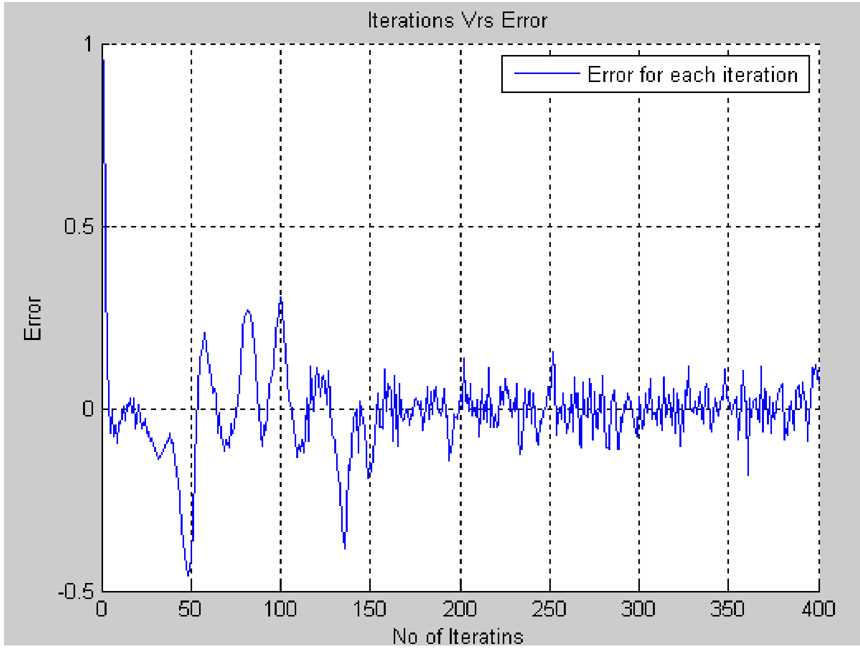

Figure 11: Learning Curve for 400 iterations and for μ=0.075

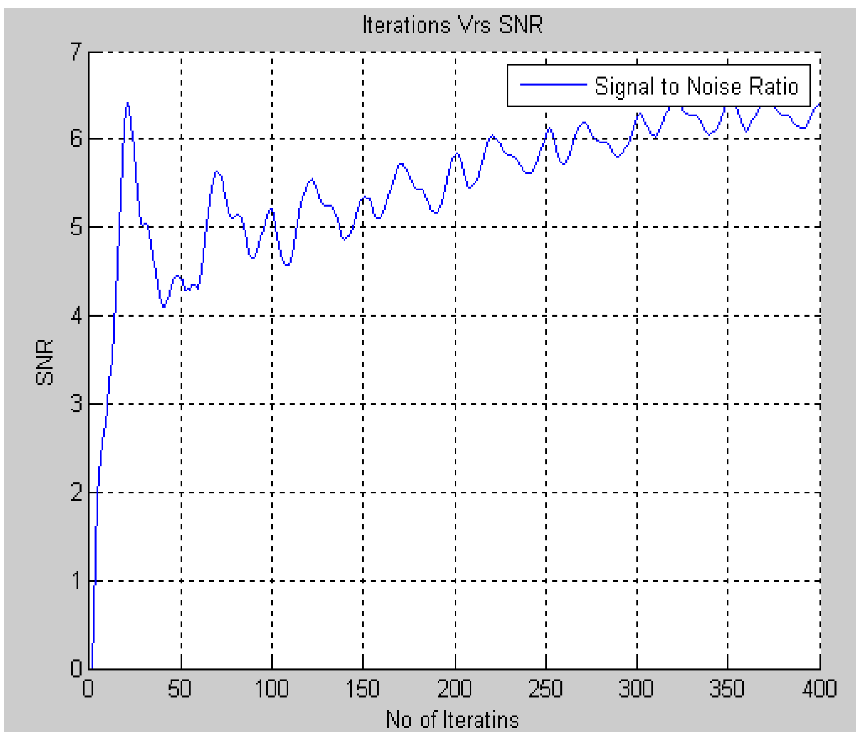

Figure 12: SNR for 400 iterations and for μ=0.075

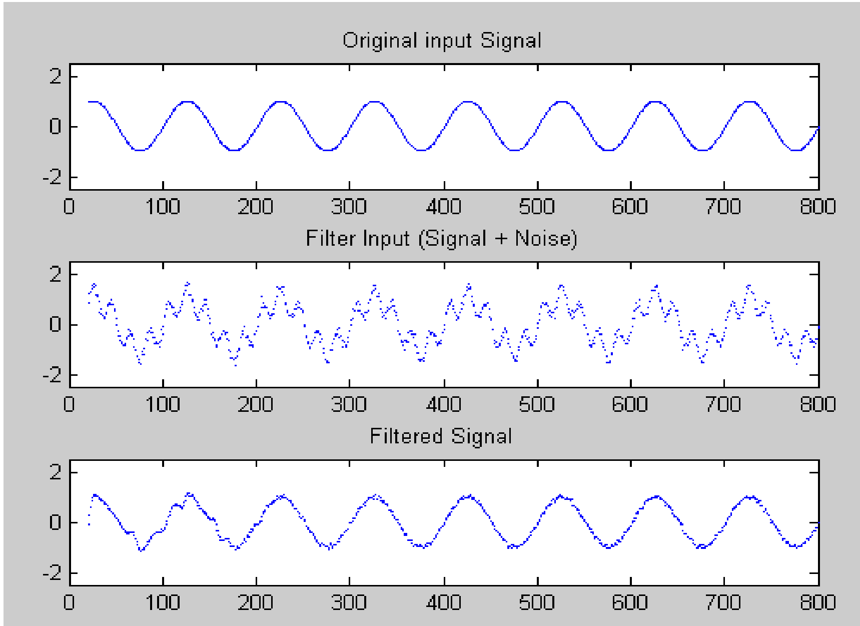

Figure 13: Original and filtered signal for 800 iterations and μ=0.05

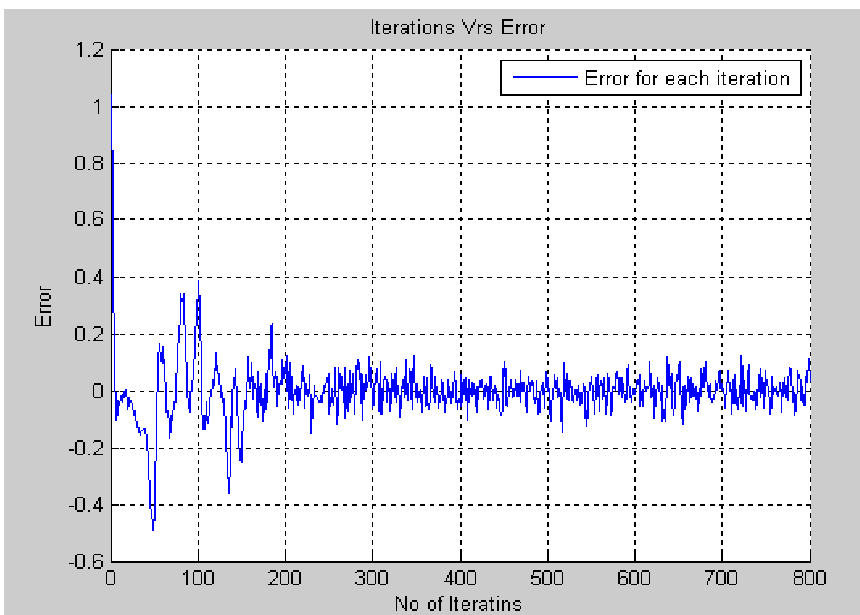

Figure 14: Learning Curve for 800 iterations and for μ=0.05

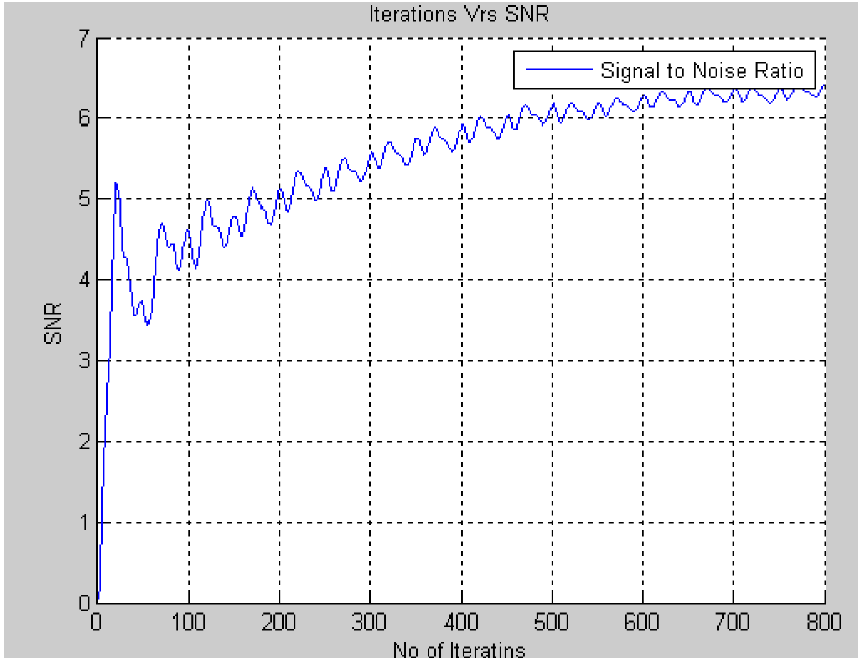

Figure 15: SNR for 800 iterations and for μ=0.05

IV.CONCLUSION

An adaptive noise cancellation using the LMS algorithm has been successfully implemented and tested using Xilinx 14.2 simulator. The algorithm shows an improvement in the performance when tested for different signals. In this algorithm the number of iterations is a parameter that is varied giving rise to attractive changes in the convergence rate. When the step size is small the convergence rate is slow and the error is large and when the step size is large the convergence rate is faster and the error is also reduced. For future work, we can implement the algorithm with an extension to accommodate more reference sensors to record noise (colored noise).

REFERENCES

[1] Mrs. A. B. Diggikar and Mrs. S. S. Ardhapurkar "Design and Implementation of Adaptive filtering algorithm for Noise Cancellation in speech signal on FPGA" 2012 International Conference on Computing, Electronics and Electrical Technologies [ICCEET].

[2] S. S. Godbole, P. M. Palsodkar and V.P. Raut "FPGA Implementation of Adaptive LMS Filter" Proceedings of SPIT-IEEE Colloquium and International Conference, Mumbai, India Vol. 2,229

[3] Mohamed Al-Ashrafy, Ashraf Salem and Wagdy Ani "An effective Implementation of floating point Multiplier" 978-1-4577-0069-9/11/2011 IEEE

[4] Simon Haykin "Adaptive Filter Theory" Pearson Education, Fourth edition.

[5] Bernard Widrow and Samuel D.Stearns "Adaptive Signal Processing" Pearson Education, 2001.

[6] V.Jagan Naveen, T.Prabakar, J.Venkata Suman and P.Devi Pradeep "Noise suppression in speech signals using adaptive algorithms" International Journal of Signal Processing, Image Processing and Pattern Recognition Vol. 3, No. 3, September, 2010

[7] J.Jebastine and Dr. B. Sheela Rani "Design and Implementation of noise free audio speech signal using fast block least mean square Algorithm" Signal & Image Processing : An International Journal (SIPIJ) Vol.3, No.3, June 2012.

[8] Charu Sharma and Monika Bhardwaj "Applications and Simulation of Adaptive Filter in noise cancellation" Journal of Information Systems and Communication. ISSN: 0976- 8742 & E-ISSN: 0976-8750, Volume 3, Issue 1, 2012, pp.-343-346

[9] Rudra Pratap "Getting started with MATLAB" Oxford University Press 2010

[10] Nazeih M.Botros "HDL Programming VHDL and Verilog" Thomson Delmar Learning, Edition 2008.

[11] K.A. Vinger, J. Torresen, "Implementing evolution of FIR-filters efficiently in an FPGA." Proceeding,. NASA/DoD Conference on Evolvable Hardware, 9-11 July 2003. Pages: 26 – 29

[12] L. l. Eriksson, M. C. Allie, and C. D. Bremigan, "Active Noise Control using Adaptive digital Signal Processing " in Proc. ICASSP , New York, 2004 pp. 2594-2597.