

Military Technical College
Kobry El-Kobbah,
Cairo, Egypt



6th International Conference
on Electrical Engineering
ICEENG 2008

Decomposition of long adaptive filter approach to acoustic echo cancellation and its implementation using FPGA

By

Amin Mohamed Nassar *

Ashraf Mohamed Ali**

Abstract:

The modeling of the acoustic echo path was presented using multiple of small adaptive filters rather than using one long adaptive filter. A new approach is proposed using the concept of decomposing the long adaptive filter into low order multiple sub-filters in which the error signals are independent on each other. The independency of the error signals exhibits the parallelism technique. This achieves our goal in increasing speed of the convergence rate. Simulation results show that the proposed decomposed least-mean-square (LMS) adaptive algorithm significantly improved the convergence rate with respect to that of the original long adaptive filter. The proposed algorithm is also compared with multiple sub-filters approach used for acoustic echo cancellation as the technique of decomposition of error. This technique is based on using multiple sub-adaptive filters in which the error signals are dependent on each other. In this way the parallelism technique is not achieved and as the result the convergence rate increases. This is different from our proposed technique which is based on independency of the error signals to assure that our algorithm has faster convergence rate. The steady state error of our proposed technique is still high as the technique of decomposition of error. This steady state error is small with respect to using one long adaptive filter and this will be obvious in our simulation results. The hardware implementation of this proposed technique was also introduced using field programmable gate arrays (FPGAs). Filtering data in real-time requires dedicated hardware to meet demanding time requirements. If the statistics of the signal are not known, then adaptive filtering algorithms can be implemented to estimate the signals statistics iteratively. The modeling of the acoustic echo path was represented by using three sub-adaptive filters of order =10 with fixed step size =0.05/3 for each adaptive filter. We use sinusoidal input signal with additive white gaussian noise (AWGN) for different signal-to-noise ratio (SNR) to examine our approach.

Keywords:

Decomposition, echo cancellation, FPGA, recourses, adaptive filter, steady state error, convergence rate and parallelism technique.

-
- * Electronics & Communication Dept., Faculty of Engineering, Cairo University, Giza, Egypt.
 - ** Modern Academy In Maadi, Communication Dept.

1. Introduction:

The increase in data band-width for telecommunications today has created a need for high quality audio teleconferencing. Echo-cancellers are a common feature to teleconferencing systems which use “hand-free” operating systems, where by the users at each end of the conference can freely interact with each other. The purpose of an acoustic echo-canceller for these applications is to reduce the amount of sound which a far-end teleconference transmits from returning to them. A common approach for estimating the impulse response of the acoustic echo path is the LMS algorithm. Acoustic echo originates due to the coupling of the loudspeaker and microphone in hands-free telephony and teleconferencing [1]. On systems that perform real-time processing of data, performance is often limited by the processing capability of the system [2]. Therefore, evaluation of different architectures to determine the most efficient architecture is an important task. Digital Signal Processing (DSP) has revolutionized the manner in which we manipulate data. The DSP approach clearly has many advantages over traditional methods, and furthermore, the devices used are inherently reconfigurable, leading to many possibilities. Modern computational power has given us the ability to process tremendous amounts of data in real-time. DSP is found in a wide variety of applications, such as: filtering, speech recognition, image enhancement, data compression, neural networks; as well as functions that are unpractical for analog implementation, such as linear-phase filters [3]. Signals from the real world are naturally analog in form, and therefore must first be discretely sampled for a digital computer to understand and manipulate. The signals are discretely sampled and quantized, and the data is represented in binary format so that the noise margin is overcome. This makes DSP algorithms insensitive to thermal noise. Further, DSP algorithms are predictable and repeatable to the exact bits given the same inputs. This has the advantage of easy simulation and short design time. Additionally, if a prototype is shown to function correctly, then subsequent devices will also. There are many advantages to hardware that can be reconfigured with different programming files. Dedicated hardware can provide the highest processing performance, but is inflexible for changes. Reconfigurable hardware devices offer both the flexibility of computer software, and the ability to construct custom high performance computing circuits [2]. The hardware can swap out configurations based on the task at hand, effectively multiplying the amount of physical hardware available. In space applications, it may be necessary to install new functionality into a system, which may have been unforeseen. For example, satellite applications need to be able to adjust to changing operation requirements [4]. With a reconfigurable chip, functionality that was not predicted at the outset can be uploaded to the satellite when needed. A simplified model for acoustic echo path is developed based on the idea that the propagation delay is caused due to the speed of the sound wave, while reflections experience attenuation of high frequency

components and some energy loss. The response of the acoustic echo path is broken into frames according to the reflections received at the microphone using process of segmentation approach for modeling of non-stationary processes [5]. Many signal processing applications call for adaptive filters with very long impulse responses. In acoustic echo cancellation, thousands of finite impulse response (FIR) filter coefficients may be required to sufficiently model the echo path. Moreover, the input data are often very strongly correlated which causes slow convergence of most adaptation algorithms, such as the well-known normalized least-mean-square algorithm. The requirements are particularly demanding for high-quality and/or multi-channel audio reproductions so that more sophisticated algorithms taking into account the input signal correlations have to be used [6]. Usually acoustic echo cancellers are realized by adaptive FIR filters, requiring thousands of coefficients to accurately model the echo return path. This leads to excessive burden of computation and slower convergence rate. One of the ways to mitigate this slowly convergent and computationally intensive long adaptive filter problem is to use decomposition. This idea is based on distributing the load of adjusting a long adaptive filter to low order multiple sub-filters updated individually by a separate adaptive algorithm. It is generally found that adaptive LMS algorithm with lower order has faster convergence [7]. In most of the cases, the eigen-value spread of the auto correlation decreases as the order of the filter decreases except for white input [8]. We present the modeling of the acoustic echo path based on segmentation approach. A new algorithm is proposed using the concept of decomposing the long adaptive filter into low order multiple sub-filters with the parallelism technique. Simulation results show that the proposed decomposed LMS adaptive algorithm significantly improved the convergence rate.

2. Previous Approaches To Acoustic Echo Cancellation

In this Session, we try to discuss different approaches used for acoustic echo cancellation and discuss the main problems in these approaches.

2.1 Echo Cancellation Using One Long Adaptive Filter

This approach, as shown in figure (1), is based on using the filtering algorithms which try to estimate the impulse response of the acoustic echo path, $h(n)$, and filter the incoming signal from the far-end, $x(n)$ [9,10]. The near-end input, $d(n)$, e.g., from a microphone, will contain both the far-end sound and the new near-end sound. The far-end sound is convolved with the estimated, $h(n)$, and subtracted from $y(n)$ before being sent to the far-end.

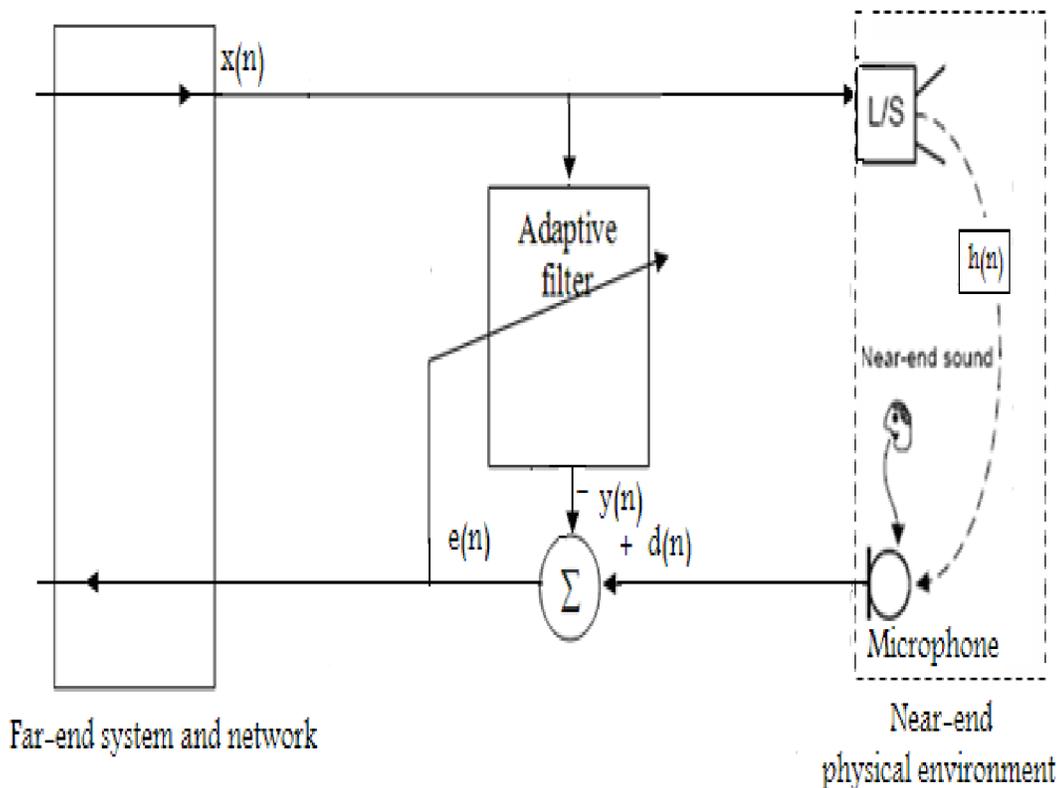


Figure (1): Echo cancellation using one long adaptive filter

The main problems that have to be addressed are system identification of the loudspeaker-room-microphone path in order to cancel the acoustic echo, $d(n)$, and that this technique requires thousands of coefficients to accurately model the echo return path. These lead to excessive burden of computation and slower convergence rate.

2.2 Decomposition of Error Technique

This approach is based on using multiple sub-adaptive filters in which error signals are dependant on each other. The idea of decomposing the input signal vector and the weight vector into sub-vectors was first presented in [11]. Here the decomposition is to partition the long single adaptive filter into smaller multiple sub-filters. Each sub-filter is updated by an individual adaptive algorithm. Adaptive algorithms are constructed depending upon how the error signal is generated. The error signal can be obtained at each stage of the sub filter for its updation. These arrangements are shown in figure (2).

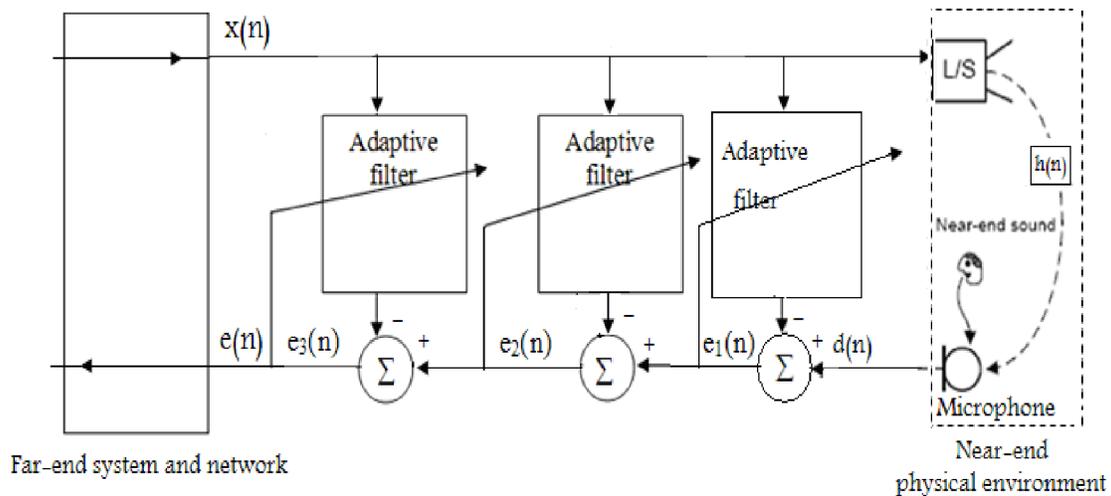


Figure (2): Decomposition of error technique

The adaptation factor is separately chosen for each sub-filter. In the different error mode although algorithm appears to be fast at the beginning but the steady state error is high [12]. The main problems of this approach are its slower convergence rate and it has large steady state error.

3. The Proposed Approach

The drawbacks of the previous approaches are their slower convergence rate and their large steady state error. Our proposed approach tries to overcome these drawbacks. The main idea of our algorithm is based on using multiple sub-adaptive filters in which error signals are independent on each other. In our proposed technique the error signals $e_1(n)$, $e_2(n)$ and $e_3(n)$, as shown in figure (3), can be obtained by comparing the output of FIR filter that represents the impulse response of the echo path in the near-end physical environment and the output of multiple low order sub-adaptive filters. Then we take the average of these error signals, using the average block, before being sent to the far end system. By this way we make error signals independent on each other and this exhibits the parallelism technique that achieves faster convergence rate. In another word the independency of the error signals, in our proposed approach, introduces the parallelism technique that tries to make the convergence rate faster. In this way we solve the problem of slower convergence rate but still the steady state error the same as that of the decomposition of error approach. This steady state error is small with respect to using one long adaptive filter.

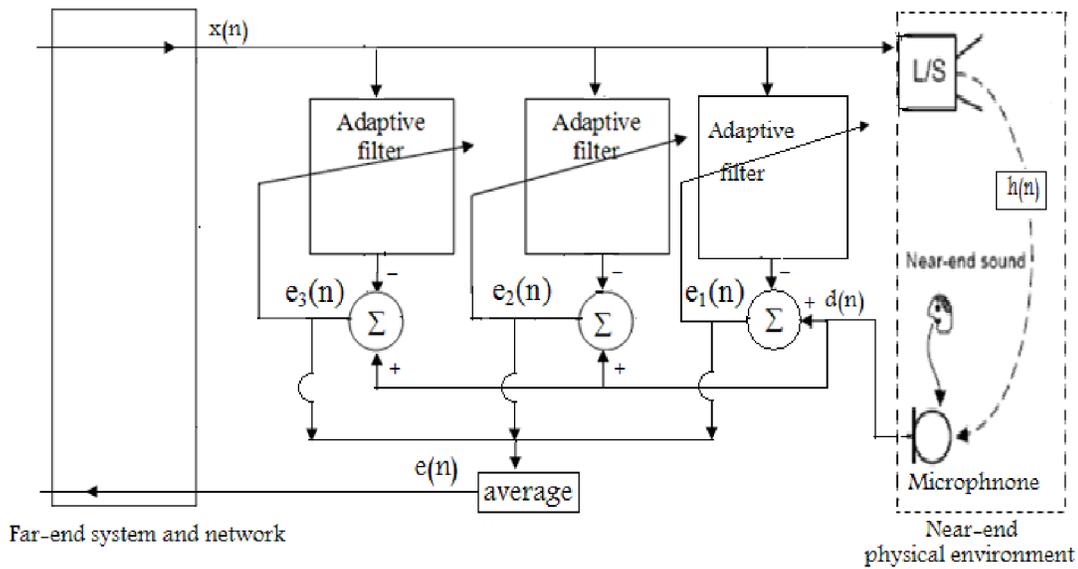


Figure (3): The proposed approach

From figure (3) we note that the error signals $e_1(n)$, $e_2(n)$ and $e_3(n)$ are independent on each other as we discussed before. The average block is used to get the average of these three error signals before being sent to the far end system. Each sub-filter can be updated by an individual adaptive algorithm. The adaptation factor is separately chosen for each sub-filter.

We should have been observed that FIR filter must be system identification of the loudspeaker-room-microphone path. Acoustic echo can result from a combination of direct acoustic coupling and multi-path effect where the sound wave is reflected from various surfaces and then picked up by the microphone. The reflected sound signal experiences attenuation, propagation delay and energy loss. The model of the acoustic echo path cannot be static unless there is no change or movement of the person and objects in the environment. Keeping in view the characteristics of the reflected sound signal, the model for multiple reflections together with the direct path from Loudspeaker to microphone can be obtained. These attenuation constants depend upon the size of the room and surface from which reflections occur.

4. Convergence Behavior of The Proposed Approach

Here we will discuss only the adaptive filter portion of the Acoustic Echo Canceller rather than discussing the other components like double talk detector and residual echo suppressor. We assume that the person in the near end physical environment is silent.

The input time series, $x(n)$, is assumed sinusoidal function with AWGN for different SNR. The microphone output is described as

$$d(n) = \sum_{j=0}^M H_j^T X_j(n) + \varepsilon(n) \quad (1)$$

Where $H_j = [h_{j,1}, h_{j,2}, \dots, h_{j,L_j}]^T$ is the Wiener solution of the j^{th} sub-filter, $X_j(n) = [x(n - (k_1 + k_2 + \dots + k_j)), x(n - (k_1 + k_2 + \dots + k_j + 1)), \dots, x(n - (k_1 + k_2 + \dots + k_j + L_j - 1))]^T$ is input sequence, M is the number of iterations taken into account and L_j is the length of low path FIR filter corresponding to each reflection. Also $H_0 = 1.0$, $X_0(n) = x(n)$ and $\varepsilon(n)$ is the ambient noise assumed independent of sequence $X_j(n)$ with a variance $\varepsilon_{\min} = E[\varepsilon^2(n)]$. $E[*]$ represents the expectation operation. The order of each adaptive sub-filter is considered to be same as that of the individual reflection path coefficient length in exact modeling.

The output of the adaptive sub-filter is given by

$$y(n) = \sum_{j=0}^M W_j^T(n) X_j(n) \quad (2)$$

Where $W_j(n) = [W_{j,1}(n), W_{j,2}(n), \dots, W_{j,L_j}(n)]^T$

The error signals are defined as

$$e_i(n) = d(n) - W_i^T(n) X_i(n); \quad i = 1, 2, 3, \dots, M \quad (3)$$

The LMS adaptation of each sub-filter, $W_i(n)$, is given as

$$W_i(n+1) = W_i(n) + \mu_i X_i(n) e_i(n); \quad i = 1, 2, 3, \dots, M \quad (4)$$

Where μ_i represents the step size of each sub-adaptive filter

We define the weight error vector

$$V_i = W_i(n) - H_i \quad (5)$$

Then the mean square error (MSE) under well known assumption is given as

$$\varepsilon(n) = \varepsilon_{\min} + \sum_{i=0, i \neq j}^M \sum_{j=0}^M E[V_i(n)] R_{i,j} E[V_j(n)] + \sum_{i=0}^M \text{tr}\{R_{i,i} E[V_i(n)]\} \quad (6)$$

Where $R_{i,j} = E[V_i(n) V_j^T(n)]$, $R_{i,i} = E[V_i(n) V_i^T(n)]$,

$E[*]$ represents the expectation operation and $\text{tr}\{*\}$ represents the trace operation.

We must note that for the MSE to converge, it is necessary that $E[V_i(n)]$ and $\text{tr}\{R_{i,i} E[V_i(n) V_i^T(n)]\}$ converge.

$$E[V_i(n+1)] = (I - \mu_i R_{i,i})E[V_i(n)] - \mu_i \sum_{j=0, j \neq i}^M R_{i,j} E[V_j(n)] \quad (7)$$

$$\text{Where } 0 < \mu_i < 2 / \text{tr}(R_{i,i}) \quad \forall i = 1, 2, 3, \dots, M$$

This gives the essential condition for convergence but still tighter bound for the step size can be obtained to confirm stability and convergence of sub-filters. Therefore we obtain the correlation matrix of weight error vector by post multiplying (7) by its transpose and taking expectation. Applying the Gaussian fourth moment factoring theorem and defining the rotation vector as

$$\tilde{V}_i(n) = Q_i^T V_i(n) \quad (8)$$

Where Q_i is the eigen vector matrix of $R_{i,i}$, The MSE convergence and stability are ensured if step size μ_i is shown such that

$$0 < \mu_i < 2 / (3\text{tr}\{R_{i,i}\}) \quad \forall i = 1, 2, 3, \dots, M \quad (9)$$

5. Loadable Coefficient Filter Taps

The heart of any digital filter is the filter tap. This is where the multiplications take place and is therefore the main bottleneck in implementation. Many different schemes for fast multiplication in FPGAs have been devised, such as distributed arithmetic, serial-parallel multiplication, and Wallace trees [13], to name a few. Some, such as the distributed arithmetic technique, are optimized for situations where one of the multiplicands is to remain a constant value, and are referred to as constant coefficient multipliers (KCM)[14]. Though this is true for standard digital filters, it is not the case for an adaptive filter whose coefficients are updated with each discrete time sample. Consequently, an efficient digital adaptive filter demands taps with a fast variable coefficient multiplier (VCM). A VCM can however obtain some of the benefits of a KCM by essentially being designed as a KCM that can reconfigure itself. In this case it is known as a dynamic constant coefficient multiplier (DKCM) and is a middle-way between KCMs and VCMs [14]. A DKCM offers the speed of a KCM and the reconfiguration of a DCM although utilizes more logic than either. This is a necessary price to pay however, for an adaptive filter.

6. A Multiplier

An approach to multiplication that uses the Full_Adder, 8-bit ripple-carry adder, Positive-edge-triggered D flip-flop with asynchronous clear, an 8-bit register, an 8-bit multiplexer, a zero detector, a variable-width shift register and a Moore state machine

is as shown in Figure (4). This figure shows a schematic that describes the interconnection of all the components for the multiplier. Notice that the schematic comprises two halves: an 8-bit-wide datapath section (consisting of the registers, adder, multiplexer, and zero detector) and a control section (the finite-state machine). The arrows in the schematic denote the inputs and outputs of each component. VHDL has strict rules about the direction of connections.

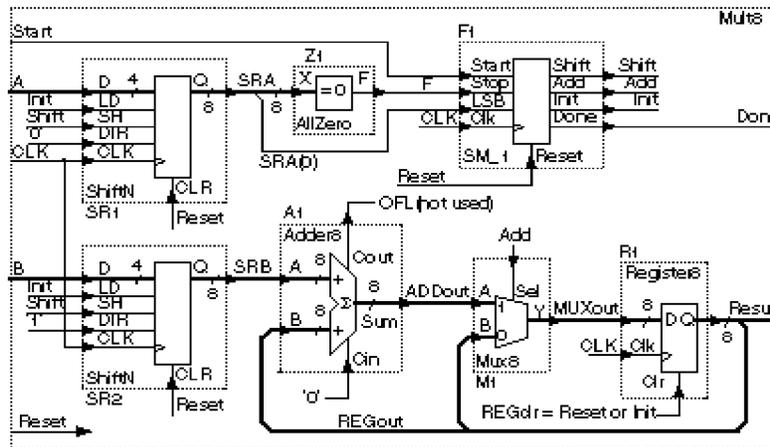


Figure (4): The multiplier used for our proposed approach

7. Hardware Verification

The proposed design was thoroughly tested on the FPGA. The VHDL design using the FPGA fabric only was tested as well as the hybrid designs using the FPGA fabric for filtering and utilizing the PowerPC for the training algorithm. To test the validity of the hardware results, an Avnet Virtex-II Pro Development kit with a Xilinx XC2VP20 FPGA was used. This board can plug directly into the PCI slot of a host computer for fast data transfer over the PCI bus. The included PCI Utility enabled this as well as quick FPGA reconfiguration over the PCI.

8. Simulation Results

8.1 Matlab Simulation

We examined our approach by using sinusoidal input signal, $x(n)$, with AWGN for different SNR. Echo path impulse response, $h(n)$, in the near end physical environment was measured at a distance of 1.0 ft using computer loudspeaker and unidynamic microphone. The sampling frequency of the simulation is 16,000 Hz. We assume that low path FIR filter which represents the impulse response of the echo path

in the near end physical environment has order =10 and the three adaptive filters have order =10. The sinusoidal input signal $x(n)$, The output of FIR filter, $d(n)$, and the average output of the three adaptive filters, $y(n)$, for different SNR are shown in figure (5) through (8).

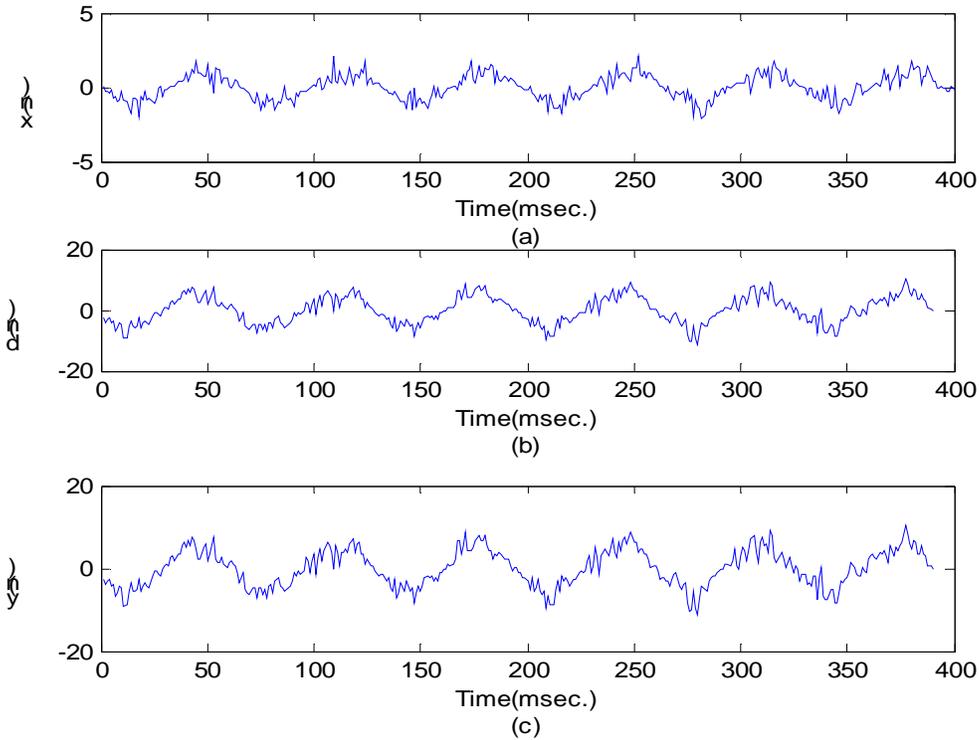
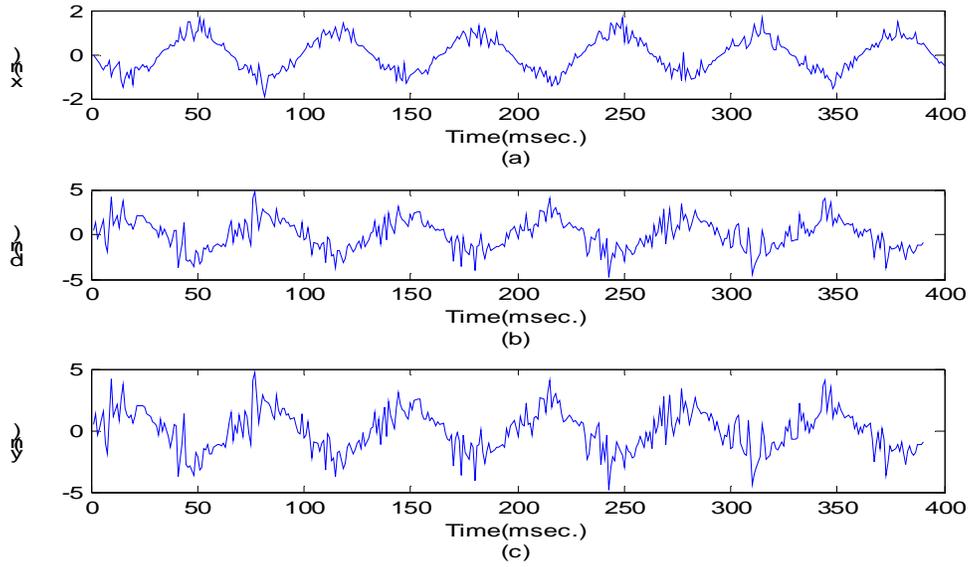
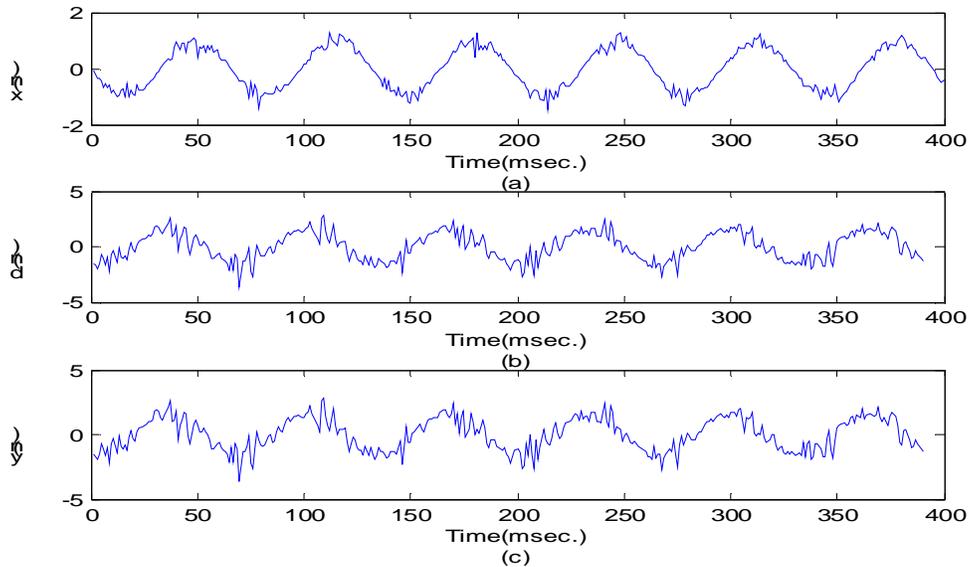


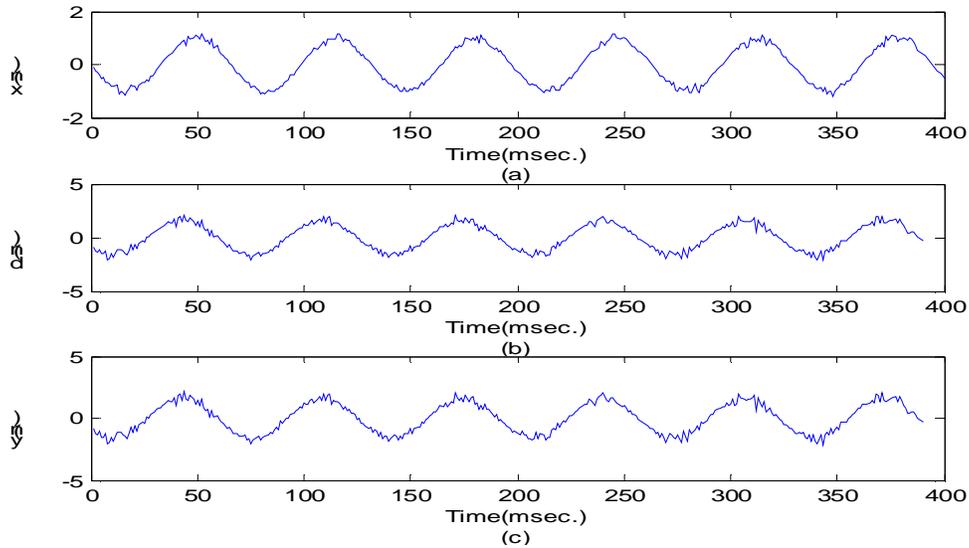
Figure (5): a) The input signal ,b) The output of FIR filter, c) The average output of the three adaptive filters
For SNR=5dB



Figure(6): a) The input signal ,b) The output of FIR filter, c) The average output of the three adaptive filters
For SNR=10dB

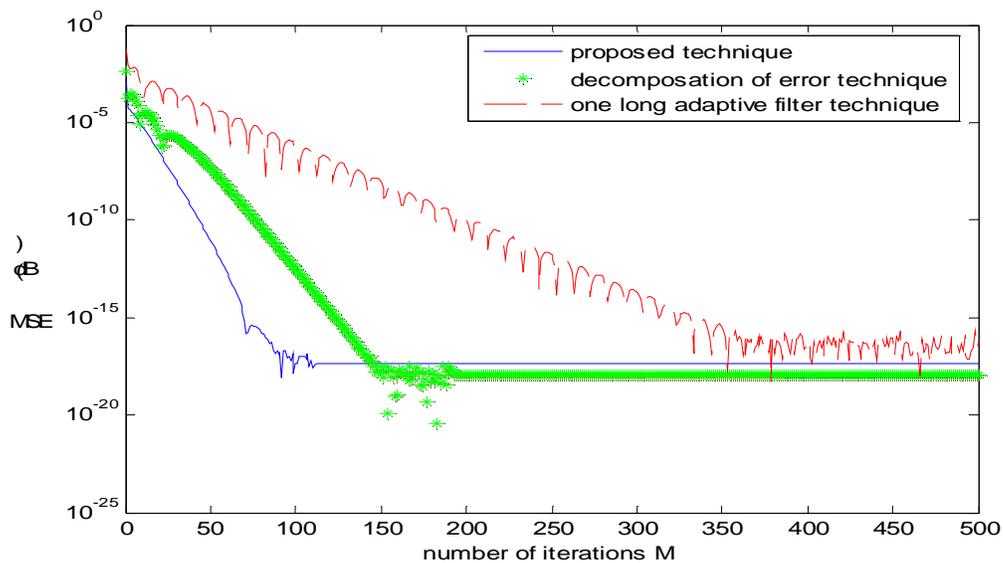


Figure(7): a) The input signal ,b) The output of FIR filter, c) The average output of the three adaptive filters
For SNR=15dB

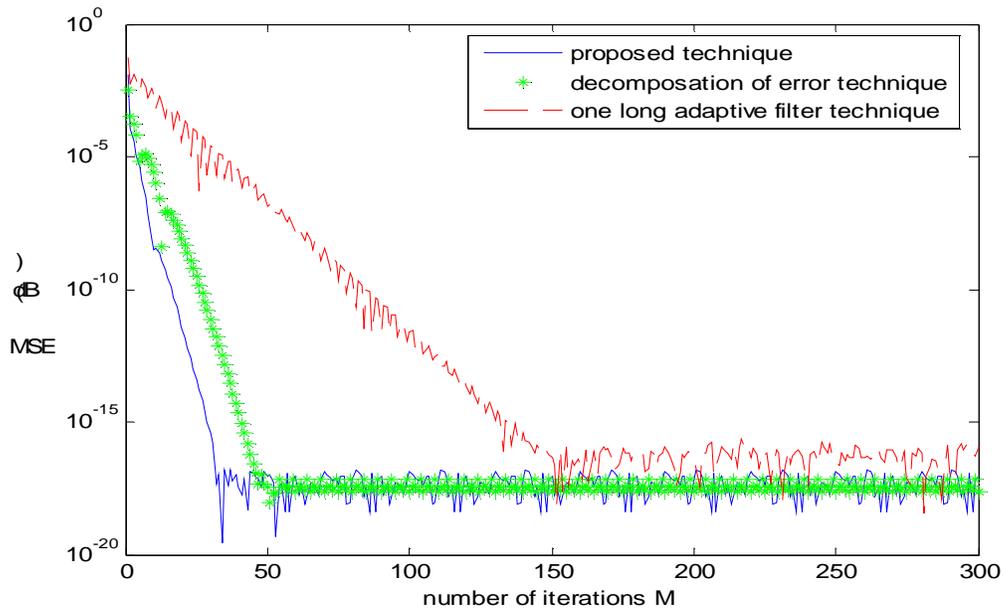


Figure(8): a) The input signal ,b) The output of FIR filter, c) The average output of the three adaptive filters
For SNR=20dB

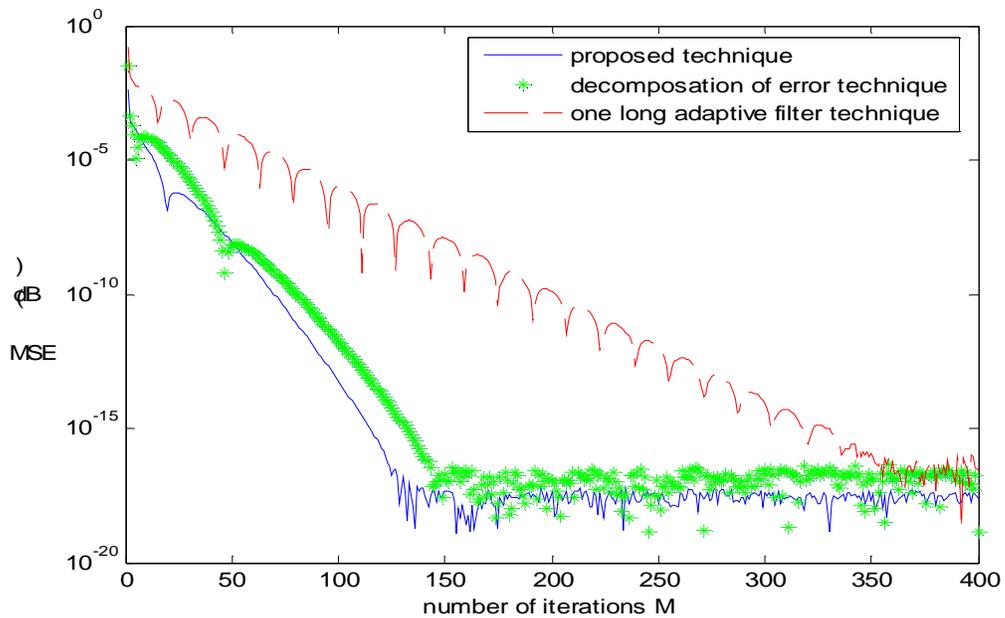
The MSE, defined in equation (6), versus the number of iterations M in the proposed algorithm for different SNR is compared with the previous techniques as shown in figure (9) through (12).



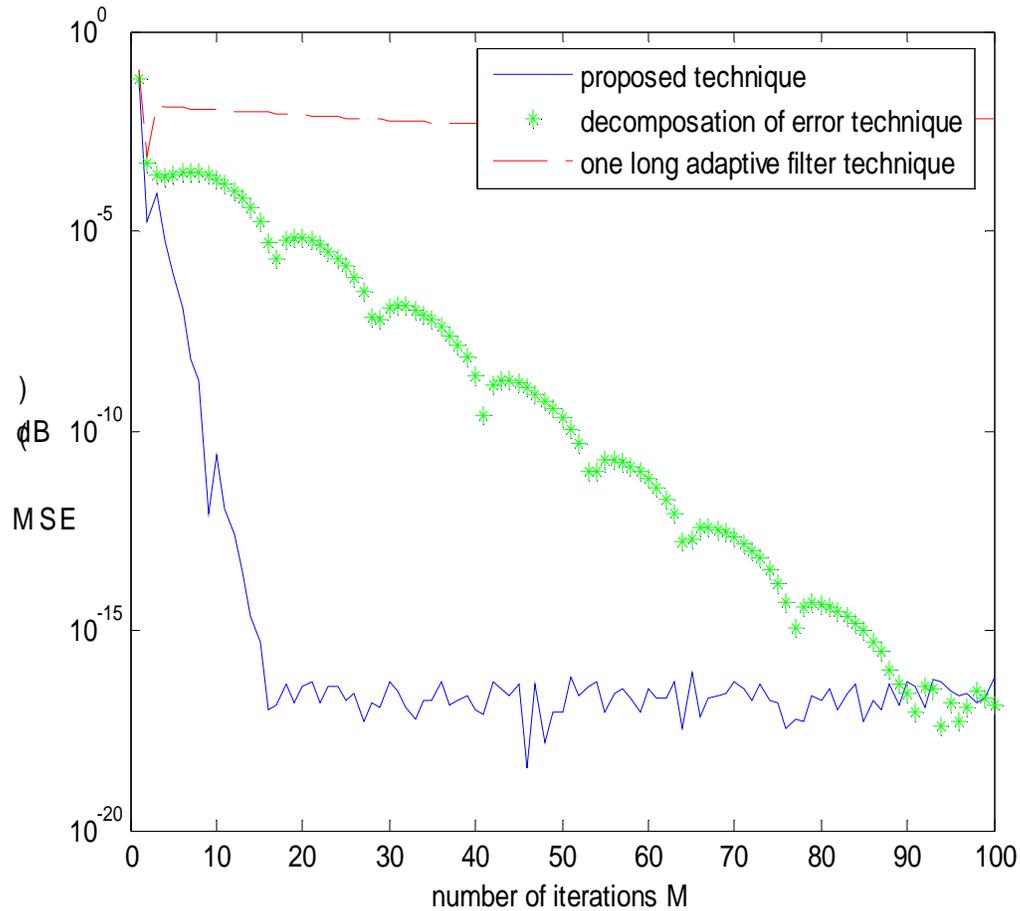
Figure(9): MSE versus number of iterations M for different approaches at SNR=5dB



Figure(10): MSE versus number of iterations M for different approaches at SNR=10dB



Figure(11): MSE versus number of iterations M for different approaches at SNR=15dB



Figure(12): MSE versus number of iterations M for different approaches at $SNR=20dB$

From Figure (9) through (12), we conclude that in our proposed technique, we need a small number of iterations to reach steady state error. We also conclude that in the technique of decomposition of error and the technique of using one long adaptive filter we need large number of iterations to reach steady state error. In this way it is obvious that our proposed technique achieves faster convergence rate. This is expected as our proposed technique is based on using multiple sub-adaptive filters with the independency of error signals to achieve the parallelism technique. This is, as we discussed before, the key idea for the improvement of our proposed technique compared with the previous techniques. It is obvious from our simulations that the steady state error of our proposed approach is the same as using technique of decomposition of error. Also it is obvious that this steady state error is small with respect to using one long adaptive filter. Table (1) demonstrates the comparison of our proposed approach with the previous approaches.

Table (1): Comparison of different approaches

Technique parameter	one long adaptive filter technique	decomposition of error technique	proposed technique
Number of adaptive filters used	One	Three	Three
Step size of each adaptive filter	0.05	0.05/3	0.05/3
Order of each adaptive filter	30	10	10
Convergence rate at SNR=5dB	350iterations	150 iterations	100 iterations
Convergence rate at SNR=10dB	150 iterations	50 iterations	40 iterations
Convergence rate at SNR=15dB	350 iterations	150 iterations	140 iterations
Convergence rate at SNR=20dB	100 iterations	90 iterations	18 iterations
Steady state error at previous SNRs	0.001	0.000001	0.000001

8.2 FPGA Simulation and Synthesis:

The algorithms for adaptive filtering were coded in Matlab and experimented to determine optimal parameters such as the learning rate for the LMS algorithm. Next, the algorithms were converted to a fixed-point representation, and finally, coded for the Virtex-II Pro. The above algorithms were converted so that all internal calculations would be done with a fixed-point number representation. This is necessary, as the embedded PowerPC has no floating-point unit (FPU), and FPGA's don't natively support floating-point either. Although a FPU could be designed in an FPGA, they are resource intensive, and therefore can feasibly only support sequential operations. Doing so however would fail to take full advantage of the FPGA's major strength, which is high parallelization. Figures (13) through (16) show the timing diagram of the proposed approach at different SNR. We perform rounding operation on the signals before converting it into binary format.

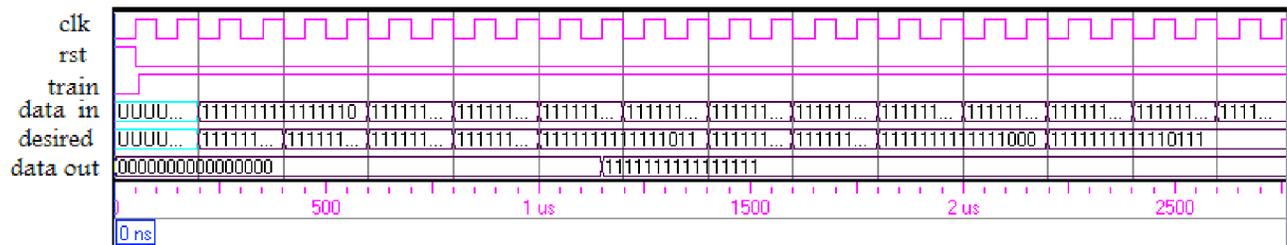


Figure (13) : Timing diagram of the proposed approach at SNR=5dB

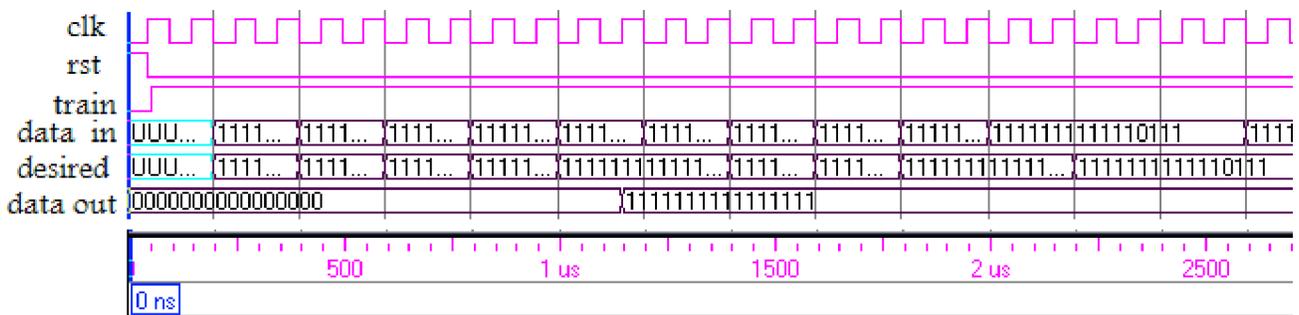


Figure (14) : Timing diagram of the proposed approach at SNR=10dB

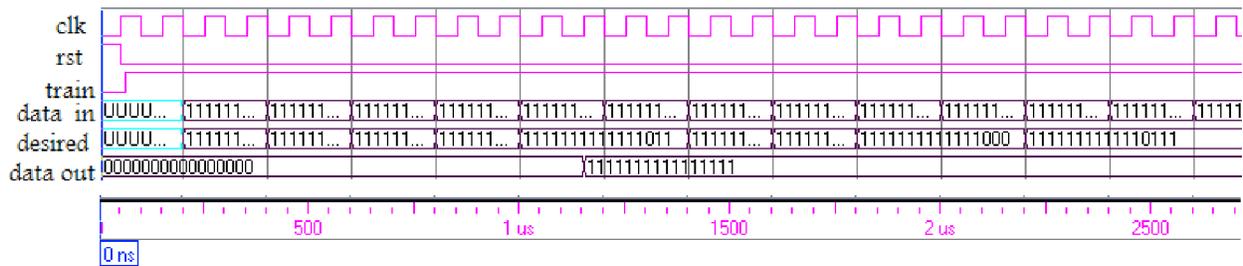


Figure (15) : Timing diagram of the proposed approach at SNR=15dB

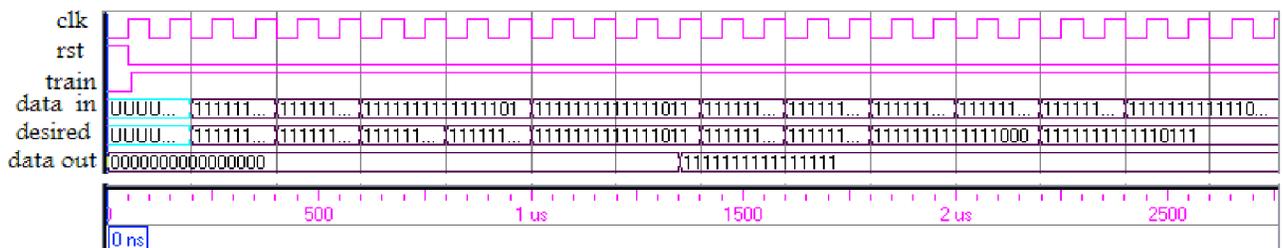


Figure (16) : Timing diagram of the proposed approach at SNR=20dB

From figure (13) through (16) we must note that, The desired is the output of FIR filter after conversion to binary format and data out is the average output of adaptive filter after conversion to binary format. We must also note that these binary numbers after 1usec begin to converge from each other this is due to the rounding operation which we preformed on the signals before the conversion into binary format. we try to introduce the device utilization for 2VP2fg256 for our proposed approach through table (2) to show how our proposed approach has an efficient realization

Table (2) : The Device Utilization for 2VP2fg256 for the proposed approach

Resources	Used	Avail	Utilization
IOs	66	240	27.50%
Global Buffers	1	32	3.13%
Function Generators	323	12288	2.63%
CLB Slices	294	6144	4.79%
Dffs or Latches	587	13008	4.51%

9. Conclusions:

An acoustic echo cancellation based on using multiple sub-adaptive filters in our proposed technique shows the improvement in performance. This technique achieves faster convergence rate than using the method of decomposition of error and that of using only one long adaptive filter. The key idea for this improvement is that our proposed algorithm is based on independency of the error signals which exhibits the parallelism technique. In this way, we achieve faster convergence rate.

The Least Mean-Square algorithm was found to be an efficient training algorithm for FPGA based adaptive filters. The used resources in the proposed approach show the efficient of this algorithm for FPGA realization. The issue of whether to train in hardware or software is based on power specifications, and is dependent on the complete system being designed. While the extra power consumed would make the PowerPC seem unattractive, as part of a larger embedded system this could be practical. If many processes can share the PowerPC then the extra power would be mitigated by the creation of extra hardware that it has avoided. With no microprocessor, a finite state machine for timing, as well as a memory interface is needed, and these will consume more power, although still less than the PowerPC. Lastly, the microprocessor can be used to easily swap out software training algorithms for application testing and evaluation. Embedded microprocessors within FPGA’s are opening up many new possibilities for hardware engineers, although it requires a new design process. The future of embedded Systems-on-Chip design will involve more precisely determining the optimal hardware and software tradeoffs for the functionality needed.

References:

- [1] J. Benesty, T. Gansler, D.R. Morgan, M.M.Sondhi, and S.L. Gay, *Advances in Network and Acoustic Echo Cancellation*, Springer-Verlag, 2001.
- [2] K.A. Vinger, J. Torresen, *Implementing evolution of FIR-filters efficiently in an FPGA*. Proceeding, NASA/DoD Conference on Evolvable Hardware, 9-11 pp. 26 – 29, July 2003.
- [3] E. C. Ifeachor and B. W. Jarvis, *Digital Signal Processing, A Practical Approach*, Prentice Hall, 2002.
- [4] S.J. Visser, A.S. Dawood, J.A. Williams, *FPGA based real-time adaptive filtering for space applications*, Proceedings, IEEE International Conference on Field-Programmable Technology, 16-18 pp. 322–326, Dec. 2002.
- [5] M. Niedzwiecki, *Identification of Time -Varying Processes*, John Wiley and Sons, Ltd.2000.
- [6] Herbert Buchner, Member, IEEE, Jacob Benesty, Senior Member, IEEE, Tomas Gänsler, Member, IEEE, and Walter Kellermann, Member, IEEE, *Robust Extended Multidelay Filter and Double-Talk Detector for Acoustic Echo Cancellation*, IEEE transactions on audio, speech, and language processing, vol. 14, pp.1633-1634, 2006.
- [7] Z. Pritzker, and A. Fuer, *Variable Length Stochastic Gradient Algorithm*, IEEE Trans.Signal Processing, vol. 39, pp.997-1001, 1991.
- [8] J. Homer, R.R. Bitmead, and I. Mareels, *Quantifying the Effects of Dimension on the Convergence Rate of the LMS Adaptive FIR Estimator*, IEEE Trans. Signal Processing, vol.46, pp.2611-2615,1998.
- [9] S. Haykin, *Adaptive Filter Theory*, PrenticeHall, Englewood Cliffs, N. J., 4th edition, 2001.
- [10] S. L. Gay and J. Benesty (eds.), *Acoustic Signal Processing for Telecommunication*, Kluwer Academic Publishers, Boston, 2000.
- [11] A.K. Chaturvedi, and G. Sharma, *A New Family of Concurrent Algorithms for Adaptive Volterra and Linear Filters*, IEEE Trans. Signal Processing, vol.47, pp.2547-2551, 1999.
- [12] R.N. Sharma, A.K. Chaturvedi and G. Sharma, *Acoustic Echo Cancellation Using Multiple Sub-Filters*, IEEE Trans. Signal Processing, vol.36, pp.2111-2114,2005.
- [13] U. Meyer-Baes, *Digital Signal Processing with Field Programmable Gate Arrays*, Springer, 2001.
- [14] K. Wiatr, *Implementation of multipliers in FPGA structures*, 2001 International Symposium on Quality Electronic Design, 26-28 Pages: 415 – 420, March 2001