

# DESIGN AND IMPLEMENTATION OF ADAPTIVE FILTERS FOR REAL TIME APPLICATIONS

Sarada.Ch

M.Tech (S.S.P), ECE department  
Lakireddy Balireddy College Of Engineering  
Mylavaram-AndhraPradesh-India

Amarbabu.Y

Associate Professor, ECE department  
Lakireddy Balireddy College Of Engineering  
Mylavaram-AndhraPradesh-India

## Abstract:

In real time signal processing applications, filtering data requires dedicated hardware to meet challenging time requirements. If the statistics of signal are not known then adaptive filtering algorithms like LMS and RLS can be implemented to estimate the signal statistics iteratively. Adaptive filters are having large range of applications such as noise cancellation, system identification, adaptive linear prediction and beamforming etc. Compared with other digital filters adaptive filter is one whose characteristics can be adapted automatically to get the desired signal.

This article deals with Adaptive noise cancellation application using LMS algorithm and RLS algorithm. An evaluation is made between these two algorithms using MATLAB programming. This application is implemented using VHDL design and the simulation results are obtained by the Xilinx synthesis tool.

**Keywords:** Adaptive filter, LMS algorithm, RLS algorithm, VHDL

## 1.INTRODUCTION

For the past many years, adaptive filters design has been an active area of scholarly research and innovative implementations. An adaptive filter is one kind of filter that self-adjusts its coefficients according to an optimizing algorithm. Adaptive filters are essential components in a wide range of signal processing, control, and communications including: 1) signal detection; 2) echo cancellation 3) noise cancellation and/or suppression 3) channel equalization; 4) system identification and inverse modeling of unknown systems; 5) forward and backward predictions and adaptive tracking; and 6) spectral analysis. In many of these applications, the hardware implementations are necessary whenever real-time execution is needed[5].

With the continuous development of the adaptive algorithm application in the digital signal processing field, there are many issues to attract everyone's attention, including a large amount of computation and the difficult to achieve high-speed and real-time. For a long time, adaptive filtering algorithms are based on the DSP chip, and achieved by the compilation or high-level language programming procedures. This can be a good way to meet the requirements in less demanding situations of real-time, but in the higher real-time requirements occasion and the harsh electromagnetic environment, it has been unable to meet the processing speed and robustness and so on. Field Programmable Gate Array (FPGA) can provide a

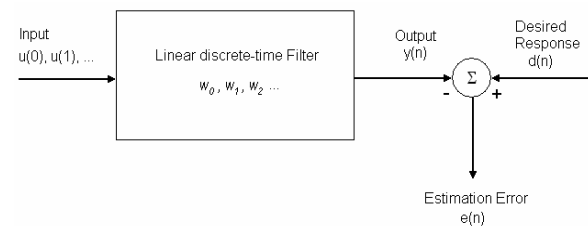
new method for the hardware implementation of adaptive algorithm through its high flexibility and integration [1].

In this paper in order to explain the performance of LMS algorithm and RLS algorithm, we designed an adaptive noise cancellation application in MATLAB programming and it is implemented in VHDL using ISIM simulator and Xilinx synthesis tool.

The contents of this paper is explained as section 2 describes about adaptive filtering and its algorithms, section 3 describes adaptive noise cancellation application, section 4 describes software simulations and section 5 describes the hardware results.

## 2. ADAPTIVE FILTERING PROBLEM AND ITS ALGORITHMS:

The goal of any filter is to extract useful information from noisy data. Whereas a normal fixed filter is designed in advance with knowledge of the statistics of both the signal and the unwanted noise, the adaptive filter automatically adjusts to a changing environment through the use of recursive Algorithms. This is useful when either the statistics of the signals are not known before hand of change with time.



**Figure1:** Block diagram for the adaptive filter problem.

The discrete adaptive filter (see figure 1) accepts an input  $u(n)$  and produces an output  $y(n)$  by a convolution with the filters weights,  $w(k)$ . The desired signal,  $d(n)$ , is compared with the output to obtain an estimation error  $e(n)$ . This error signal is used to adjust the filter's weights automatically for the next time instant. Several adaptive algorithms exist for the weight adjustment, such as the *Least - Mean - Square* (LMS) and the *Recursive Least - Squares* (RLS) algorithms.

The choice of training algorithm is dependent upon needed convergence time and the computational complexity available, as statistics of the operating environment [3].

This section briefly describes two of the most recognized adaptive filter design algorithm namely the LMS and the RLS.

**a) LMS algorithm:**

The LMS algorithm is based on the principle of Minimum Mean square error and the steepest descent algorithms. 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. The following mathematical model describes the LMS algorithm [6].

$$y(n) = w^T(n).x(n)$$

$$e(n) = d(n) - y(n)$$

$$w(n+1) = w(n) + \mu e(n)x(n)$$

where  $y(n)$  is the filter output,  $w^T(n)$  is the filter weights in transposed form,  $x(n)$  is the filter inputs,  $d(n)$  is the desired filter output,  $e(n)$  is the error signal which is used for training the filter weights and  $\mu$  is the stepsize factor which is used for controlling the stability and the rate of convergence. the step size must be a small positive value ( $\mu \ll 1$ ) and

$$0 < \mu < 1/2.N.R$$

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

$$R = E[x(n).x^T(n)]$$

**b) RLS algorithm:**

The recursive-least-squares (RLS) algorithm is based on the well-known least squares method. The RLS algorithm recursively solves the least squares problem.

In the following equations, the constants  $\lambda$  and  $\delta$  are parameters set by the user that represent the *forgetting factor* and *regularization parameter* respectively. The forgetting factor ( $\lambda$ ) is a positive constant less than unity, that is roughly a measure of the memory of the algorithm; The vector  $w$  represents the adaptive filter's weight vector and the  $M$ -by- $M$  matrix  $P$  is referred to as the inverse correlation matrix. The vector  $\pi$  is employed as an intermediary step to computing the gain vector  $k$ . This gain vector is multiplied by a priori estimation error  $\xi(n)$  and added to the weight vector to update the weights. Once the weights are updated the inverse correlation matrix is recalculated, and also the training resumes with the new input values. A summary of the RLS algorithm follows:

Initialize the weight vector and the inverse correlation matrix  $P$ .

$$\hat{w}^H(0) = \bar{\delta}$$

$$p(0) = \delta^{-1} I$$

$$\delta = \begin{cases} \text{Large positive constant for low SNR} \\ \text{small positive constant for high SNR} \end{cases}$$

For each instance of time  $n = 1, 2, 3 \dots$ , compute:

$$\Pi(n) = p(n-1) u(n)$$

$$k(n) = \frac{\Pi(n)}{\lambda + u^H(n)\Pi(n)}$$

$$\xi(n) = d(n) - \hat{w}^H(n-1)u(n)$$

$$\hat{w}(n) = \hat{w}(n-1) + k(n) \xi^*(n)$$

and

$$p(n) = \lambda^{-1} p(n-1) - \lambda^{-1} k(n) u^H(n) p(n-1)$$

An adaptive filter trained with the RLS algorithm can converge up to an order of magnitude faster than the LMS filter at the expense of increased computational complexity.

**3) ADAPTIVE NOISE CANCELLATION SYSTEM**

In adaptive noise cancellation, the adaptive filter is usually designed as a transversal FIR filter structure. The transversal filter consists of three basic elements; unit delay elements, multipliers and adders as shown in Figure 2. The unit delay elements are designed using the unit delay operator  $z^{-1}$ . The output of the unit-delay operator  $z^{-1}$  is a delayed copy of its input. The number of delay elements represents the filter order.

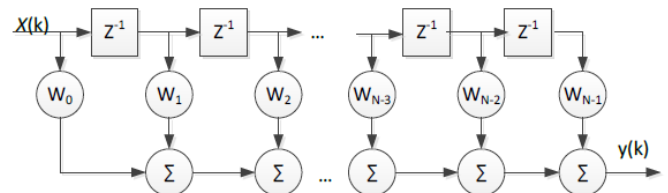


Figure 2. The Transversal FIR Filter Structure

The adaptive noise cancellation filter employing the LMS algorithm can be implemented as shown in Figure 3.

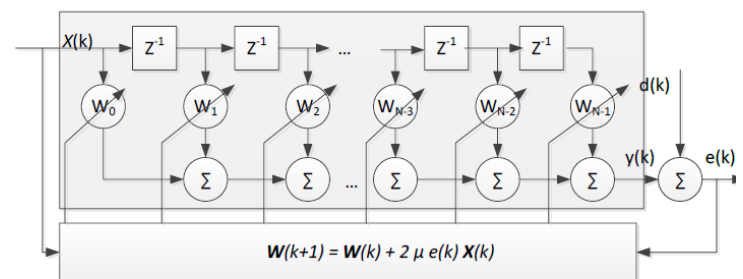


Figure 3. Adaptive noise cancellation filter

**4) SOFTWARE SIMULATION RESULTS**

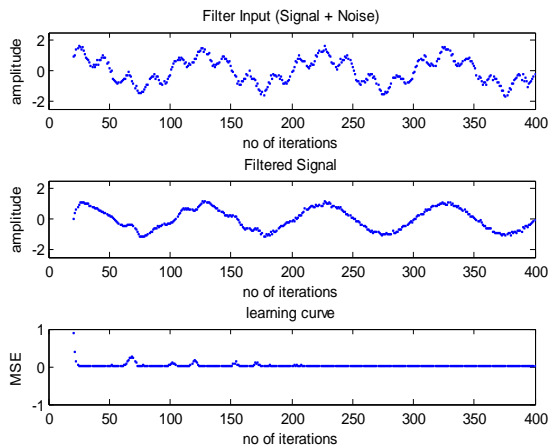
Adaptive noise cancellation is designed by using both LMS algorithm and RLS algorithm. In this we have taken input as sinusoidal signal plus 3 types of noises. those are uniform noise, Gaussian noise and additive white Gaussian noise.

**Specifications of LMS and RLS algorithms in MATLAB**

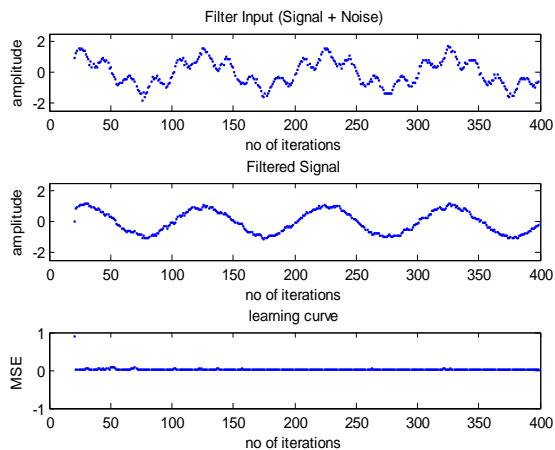
- No of filter taps=20
- LMS step size( $\mu$ ) =0.05
- RLS forgetting factor( $\lambda$ ) =0.99
- RLS regularization factor( $\delta$ ) =1
- Initial value of filter coefficients =0

*a) Signal corrupted with additive white Gaussian noise*

**MATLAB LMS Result**

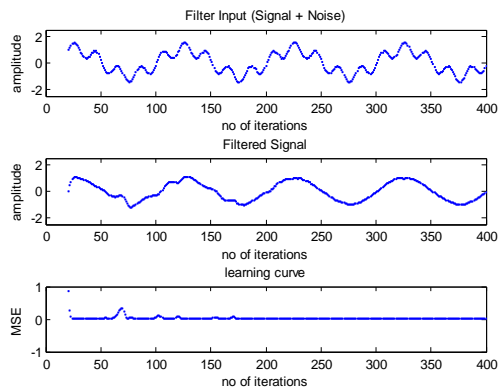


**MATLAB RLS Result**

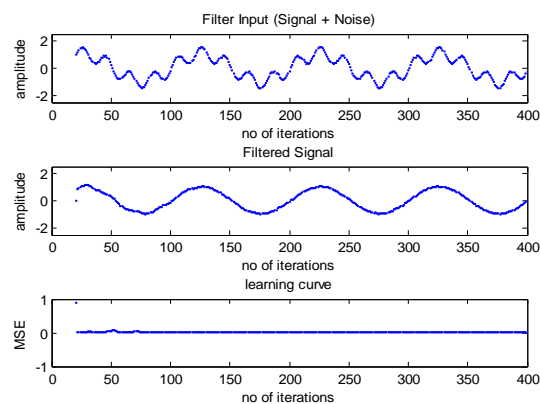


*b) Signal corrupted with uniform noise*

**MATLAB LMS result**

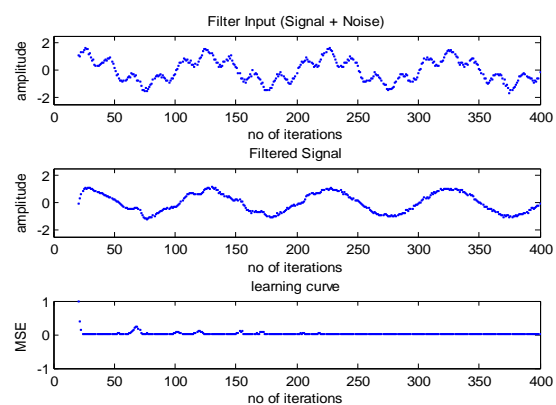


**MATLAB RLS result**



*c) Signal corrupted with gaussian noise*

**MATLAB LMS result**



**MATLAB RLS result**

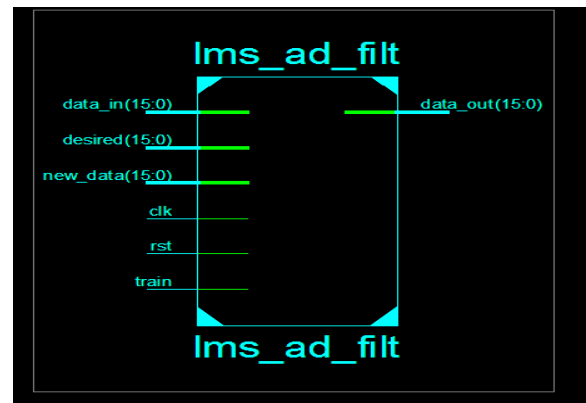
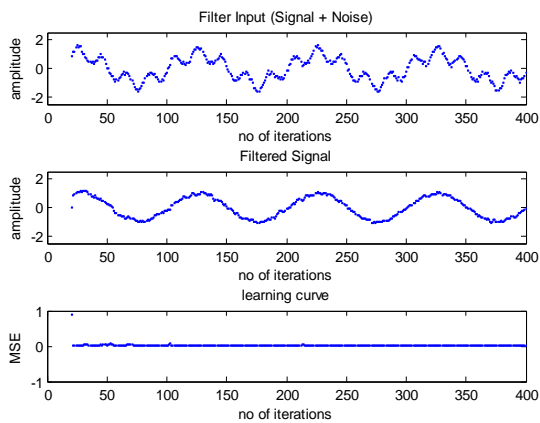
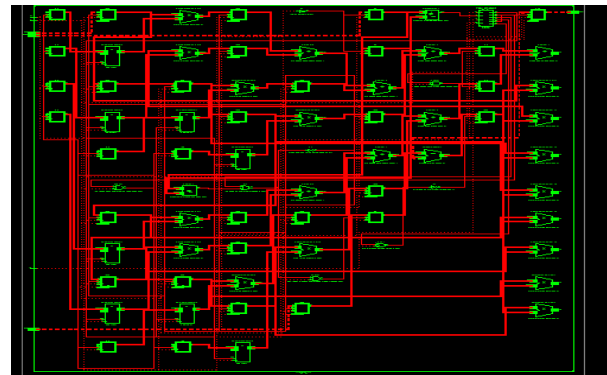


Fig5: schematic diagram of LMS adaptive filter



Algorithm	No of taps	Total no of iterations	No of iterations to converge	MSE	Computations
LMS	20	400	175	0.024	2N+1
RLS	20	400	60	0.014	4N <sup>2</sup>

Fig 6:RTL schematic diagram of LMS adaptive filter.

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	387	4656	8%
Number of Slice Flip Flops	587	9312	6%
Number of 4 input LUTs	317	9312	3%
Number of bonded IOBs	67	232	28%
Number of MULT18K18S10Ds	16	20	80%
Number of GCLKs	1	24	4%

Table 2 : LMS adaptive filter synthesis results

in these results input is the sinusoidal signal plus noise. learning curve is drawn between mean square error vs. no of iterations. In matlab LMS result the output is converged after 175 iterations where as in matlab RLS result the output is converged after 60 iterations only. LMS algorithm takes less no of computations compared with RLS algorithm. comparison between LMS and RLS is shown in the following table1.

Table1: comparison between LMS and RLS algorithms

**5)HARDWARE IMPLEMENTATION RESULTS**

Adaptive noise cancellation is implemented in VHDL program by using Xilinx synthesis tool. The simulation results are shown in the figure.

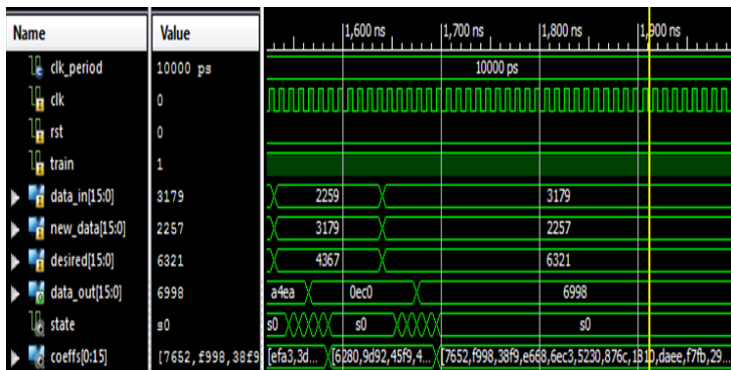


Fig 4: convergence of data o/p with i/p data

The schematic diagram of LMS adaptive filter is shown the figure. Here it has 6 inputs and 1 data output. Data\_in is 16 bit data,desired data is 16 bit and the data\_out is of 16 bit size.

## 6) CONCLUSIONS:

This paper uses MATLAB software to design LMS and RLS adaptive filter algorithms and comparison between these two is done. This paper uses a high level language called VHDL to design LMS algorithm for adaptive noise canceller. A desired signal corrupted by additive noise can often be recovered by adaptive noise canceller using the Least Mean Square (LMS) algorithm. In this paper strategies & implementation of Least Mean Square algorithm for adaptive noise canceller is described. The Least Mean Square algorithm was found to be the most efficient training algorithm for FPGA based adaptive filters. Experimental results show that the platform achieves the accuracy and the real-time of the adaptive filter algorithm.

## REFERENCES:

[1]” Design and Implementation of LMS Adaptive Filter Algorithm Based on FPGA” 2013 2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation (IMSNA)

[2] Tian Lan<sup>1</sup>, Jinlin Zhang<sup>2</sup>” FPGA Implementation of an Adaptive Noise Canceller” 2008 International Symposiums on Information Processing

[3] Joseph G. Petrone,*Florida State University.*” Adaptive Filter Architectures For FPGA Implementation” Electronic Theses, Treatises and Dissertations.

[4] “Adaptive Filtering”by: Thomas Drumright Spring 1998.

[5]” Software/Hardware Implementation of an Adaptive Noise Cancellation System” Dr. Wagdy H Mahmoud, University of the District of Columbia  
Dr. Nian Zhang, University of the District of Columbia.

[6] “ Performance Evaluation of LMS and DLMS Digital Adaptive FIR Filters by Realization on FPGA” Hesam Ariyadoost<sup>\*1</sup>, Yousef S. Kaviani<sup>\*2</sup>, Karim Ansari-As<sup>\*3</sup>  
*Faculty of Engineering, Shahid Chamran University, Ahvaz, Iran.*

[7] “ Implementation of Adaptive Noise Canceller using LMS Algorithm” *National Conference on Innovative Paradigms in Engineering & Technology (NCIPET-2013) Proceedings published by International Journal of Computer Applications® (IJCA)*

[8]Mamta M. Mahajan et al ,”Design of Least Mean Square algorithm for Adaptive Noise canceller,”Issue2,VOL.5,May-2011.

[9].S.A.Hadei,” A family of Adaptive Filter Algorithms in noise Cancellation for speech Enhancement”, International

Journal of computer and Electrical Engineering,  
VOL.2,No.2,April 2010,pp 307-315.

[10]”Implementing Evolution of FIR-Filters Efficiently in an FPGA”Knut Arne Vinger and Jim Torresen.

[11] Simon Haykin, “Adaptive filter theory”, Third edition, Prentice Hall, 2002.

[12] Zhengwei Hu, Zhiyuan Xie, “Adaptive LMS Algorithm Based on FPGA”, Journal of North Electric Power University, Vol.30, No.4, pp.74-77, July 2003.

[13] A. Elhossini, S. Areibi and R. Dony, “An FPGA implementation of the LMS adaptive Filter for audio processing,” In Proceedings of IEEE International Conference on Reconfigurable Computing and FPGAs 2006 (RECONF’06), pp. 1–8, September 2006.

[14] A simple adaptive beamforming algorithm with Interference suppression, K.Meena alias Jeyanthi 1, Dr.A.P.Kabilan<sup>2</sup> June 12, 2009.

[15] Yaghoub & Mollaei(2009),”Hardware Implementation of Adaptive filters: Publishing:  
Proceeding of 2009 student conference on Research and Development (SCORED 2009),16-18.

## Authors

1. **Ch.Sarada** has received her B.Tech from JNTU Kakinada with Electronics & Communications Engineering Specialization and at present she is pursuing M.Tech from JNTU Kakinada University, AP,India in the area of Systems and Signal Processing .She has actively involving in college level activities and technical symposiums.

2.**Mr.Y.Amar Babu** has received his Master’s Degree from Bharathiyar University in the Discipline of VLSI.He is pursuing his Ph.D in the VLSI and Communications area.He has been actively involved in teaching and research for the past 15Years and he is working as Assoc Professor, in the Dept of ECE,Lakireddy Balireddy College Of Engineering, Mylavaram.AP,India.