IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 6, Issue 3, Ver. III (May. - Jun. 2016), PP 60-65
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Implementation of LMS Adaptive Filter using Vedic Multiplier

Neha Saxena
Department of Electronics and Communication Engineering , Shri Vaishnav Institute of Technology & Science (SVITS), Indore (M.P) India.

Abstract: *In this paper, we propose LMS (least mean squares) algorithm for the designing an of adaptive traversal filters using two different techniques, a carry save multiplier based architecture and a high speed vedic mathematics based multiplier architecture. Finally comparing the results to evaluate the best design in terms of processing speed. Both of the adaptive filter architectures have been coded in verilog HDL language and synthesized, simulated in Xilinx ISE environment. The main application areas includes unknown system identification for communication processes, Channel Equalization, Noise Cancellation and Echo cancellation like DSP operations.*

Keywords: *Adaptive Filter, LMS Algorithm, Vedic multiplier, Transversal FIR filter, Carry Slave multiplier*.

I. Introduction

Adaptive filters are composed of three basic modules such as filtering structure, performance criterion and adaptive algorithm. The filtering structure determines the output of the filter from given input samples. FIR is preferred over the IIR for filtering due to stability measures. Then, the performance criterion is chosen according to the application and it is used to derive the adaptive algorithm. The three generally used performance criteria are mean squared error, least squares and weighted least squares. Finally, the adaptive algorithm is used to update the filter coefficient based on the performance criterion to improve the performance. An adaptive filter automatically sets its weights with the changing environment by using various adaptive algorithms like LMS and RLS etc. and it is used to evaluate unknown parameters for the system and adaptive estimation of parameters or signals.

II. Adaptive Filter

An adaptive filter takes a signal input of x(n) and gives the output signal as y(n) in the operation of convolution with the weight w(k).

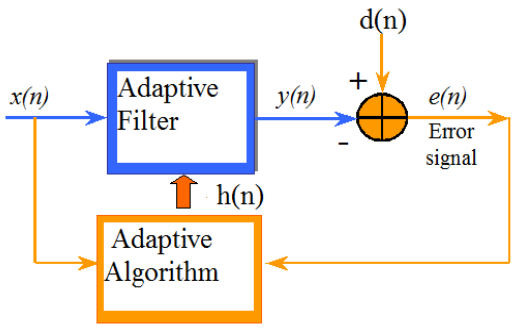

Fig. 2.1 Block diagram of Adaptive filter

At the same time a desired signal/wished signal d(n), is used to calculate the error denoted by *e*(n). The resulted e(n) error signal is used to automatically adjust the filter weight with the use of an adaptive algorithm. Following are some example algorithms used for adaptive change of weight. These are LMS-Least Mean Square and RLS-Recursive Least Square.

An FIR Transversal filter is chosen as adaptive filter because of its stability. The use of the transversal structure allows relatively straight forward construction of the filter. The output of the filter is compared with the desired response to find the error signal. The error signal is used by the adaptation algorithm to update the adaptive filter coefficient vector according to mean square error performance criterion. In general, the whole adaptation process aims at minimizing some metric of the error signal, forcing the adaptive filter output signal to approximate the reference signal in a statistical sense. There are several adaptation algorithms with different

performance criterion. Due to its low complexity and proven robustness, Least Mean Square (LMS) algorithm is used widely.

III. Transversal Filter

An N-Tap transversal was assumed as the basis for this adaptive filter. The value of N is determined by practical considerations. An FIR filter was chosen because of its stability. The use of the transversal structure allows relatively straight forward construction of the filter.

As the input, filter coefficients and output of the filter are all assumed to be complex valued then the natural choice for the property measurement is the modulus, or instantaneous amplitude. If y(k) is the complex valued filter output, then |y(k)| denotes the amplitude.

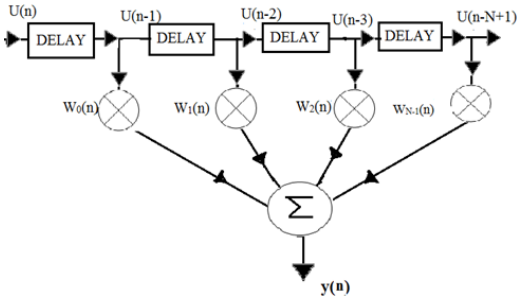

Fig. 3.1 Block diagram of Transversal filter

The structure of a transversal FIR filter shown in Fig. 3.1 with *N* tap weights (adjustable during the adaptation process) with values at time *n* denoted as:

$W_0(n), W_1(n)\ldots\ldots\ldots\ldots..W_{N-1}(n)$ (1)

The tap weight vector, $\underline{W}(n)$ is represented as:

$\underline{W}(n) = [\, W_0(n)W_1(n)\ldots\ldots\ldots W_{N-1}(n)]T$

The tap input vector, $\underline{U}(n)$ is represented as:

$\underline{U}(n) = [U(n)\ U(n-1)\ldots\ldots\ldots..U(n-N+1)]T$

The FIR Filter output, y(n) can then be expressed as:

$$y(n) = \underline{W}^T(n)\ \underline{U}(n) = \sum_{i=0}^{N-1} W_i(n)\ U(n-i) \qquad (2)$$

Where
T : Transpose
n: Time index function
N: Order of Filter

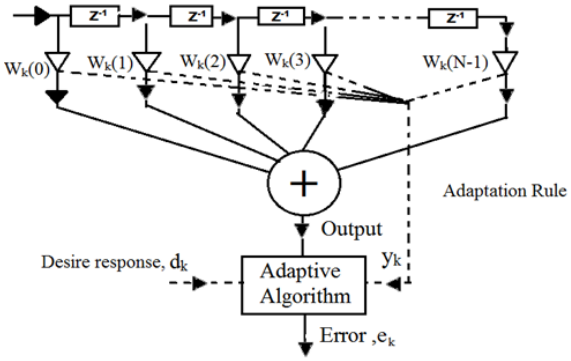

Fig.3.2 Adaptive Transversal Filter

In a transversal filter of length N, as depicted in Fig. 3.2, at each time n the output sample y[n] is computed by a weighted sum of the current and delayed input samples x[n], x[n − 1], ..and the error is being calculated by the adaptive algorithm.

IV. Lms Algorithm

It is a stochastic gradient descent method in that the filter is adapted based on the error at the current time. It was invented in year 1960 by Stanford University professor Bernard Widrow and his first Ph.D. student, Ted Hoff. LMS is linear adaptive filter algorithm and it is consisted of filtering process and adapting process.

FIR filters:

$$y(n) = w_0(n)x(n) + w_1(n)x(n-1) + \ldots + w_{M-1}(n)x(n-M+1) \quad (3)$$

$$\sum_{k=0}^{M-1} w_k(n)x(n-k) = \underline{w}(n)^T\underline{x}(n),\ n= 0, 1, 2, \ldots,\infty \quad (4)$$

Error between filter output y(t) and a desired signal d(t):

$$e(n) = d(n) - y(n) = d(n) - \underline{w}(n)^T\underline{x}(n) \quad (5)$$

Change the filter parameters according to

$$w(n+1) = w(n) + \mu x(n)e(n) \quad (6)$$

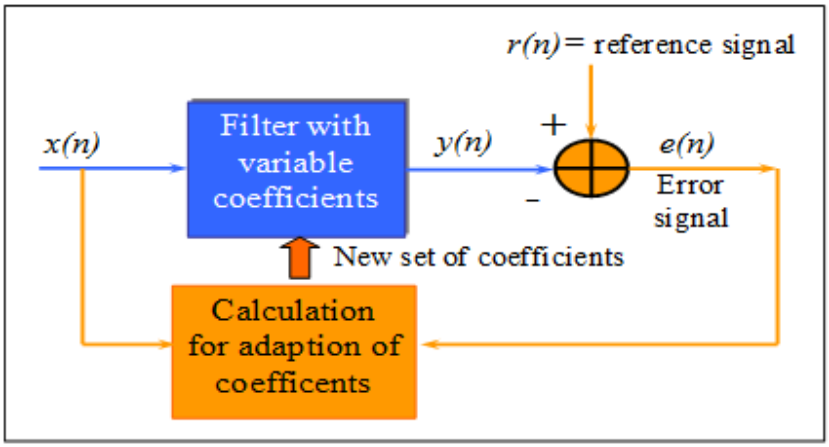

Fig. 4.1 Block diagram of LMS adaptive filter

LMS Algorithm uses a rough gradient approximation, and seeks the wished weight vector This process is used to find the weight vectors for training the ALC (Adaline). The learning rules can be incorporate to the same device that therefore can be auto adapted as there are presented the wished inputs and outputs. The weight vectors values are changed as every combination input-output is processed. This goes on until the ALC gives the correct outputs. This is a truly training process since there is not necessary to clearly calculate the weight vector value.

V. Vedic Multiplier

The Vedic Multiplier is based on a novel technique of digital multiplication which is quite different from the conventional method of multiplication like add and shift Where smaller blocks are used to design the bigger one. The 8 bit number is decomposed into 4 bit and then again the 4 bits are further decomposed into 2 bit numbers. Vedic multiplier is faster among the other conventional multiplier. Speed is improved by parallelizing the generation of partial products with their concurrent summations. Also as the number of bit multiplication increases from 2 x 2 bits to 8 x 8 bits the time delay associated with it is greatly reduced as compared to other multipliers. Fig. 5.1 shows the implementation of 8 bit vedic multiplier.

The designing of Vedic Multiplier is based on a novel Urdhava Triyakbhyam technique of digital multiplication which is quite different from the conventional method of multiplication like add and shift. Urdhava Tiryakbhyam literally means Vertical & Crosswise. It main concept of this sutra is that the generation of all partial products can be done with the concurrent addition of these partial products. The algorithm can be implemented for n x n bits.

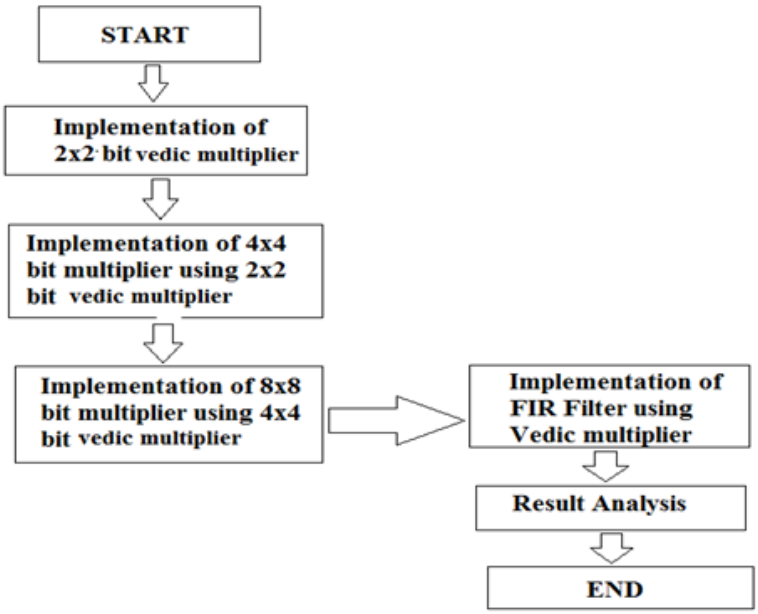

Fig. 5.1 Flow diagram of implementation of 8 bit vedic multiplier

.

5.1 Multiplication of two decimal numbers using Vedic multiplier- 325*738

The multiplication of two decimal number is illustrated in Fig.: 5.2 .The digits on the both sides of the line are multiplied and added with the carry from the previous step. This generates one of the bits of the result and a carry. This carry is added in the next step and hence the process goes on. If more than one line are there in one step, all the results are added to the previous carry. In each step, least significant bit acts as the result bit and all other bits act as carry for the next step. Initially the carry is taken to be zero.

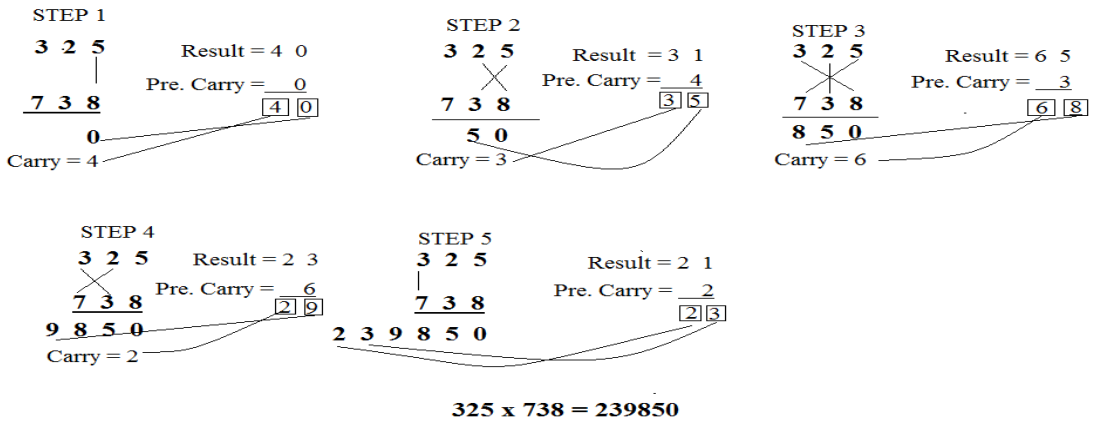

Fig. 5.2 Vedic multiplication of two decimal numbers

VI. Simulation Result

The LMS adaptive filter using vedic multiplier is found to be highly efficient in terms of processing speed. For comparison we have verilog coded the carry slave multiplier for the implementation of LMS adaptive filter. Figure 6.1 shows the simulation results of LMS adaptive filter using vedic multiplier and Figure 6.2 shows the simulation results of LMS adaptive filter using carry slave multiplier. The filter structured in Verilog is synthesized on Xilinx ISE.

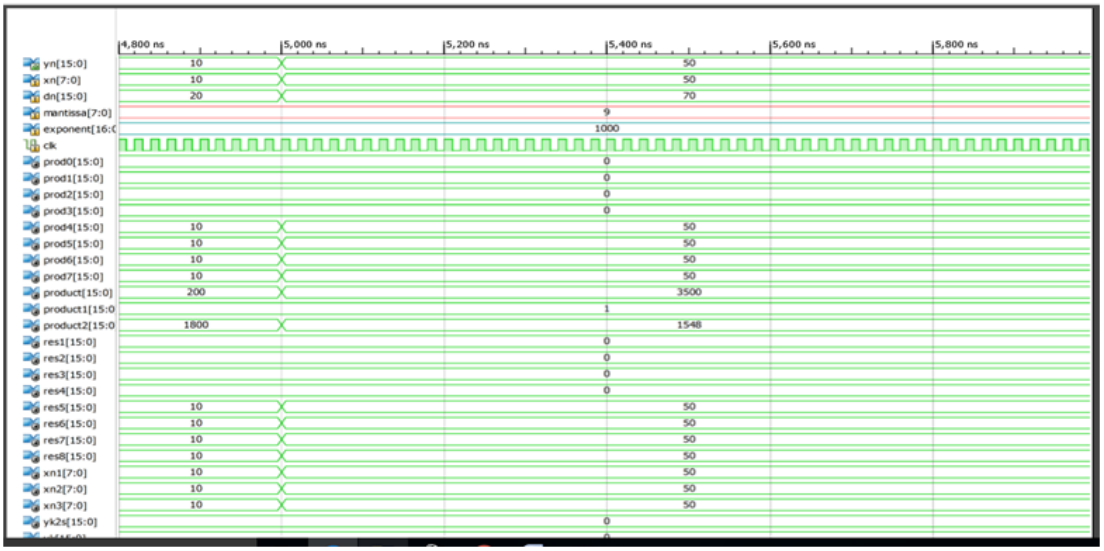

Fig. 6.1 Simulation result of LMS adaptive filter using vedic multiplier

DEVICE UTILIZATION SUMMARY (ESTIMATED VALUES) **Target Device: xc3e50-5pq208**			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	253	768	32%
Number of Slice Flip Flops	176	1536	11%
Number of 4 input LUTs	445	1536	28%
Number of Bounded IOBs	41	124	33%
Number of GCLKs	1	8	12%

Table 1.1: Device Utility Summary for 8 bit LMS Adaptive Filter using Vedic Multiplier

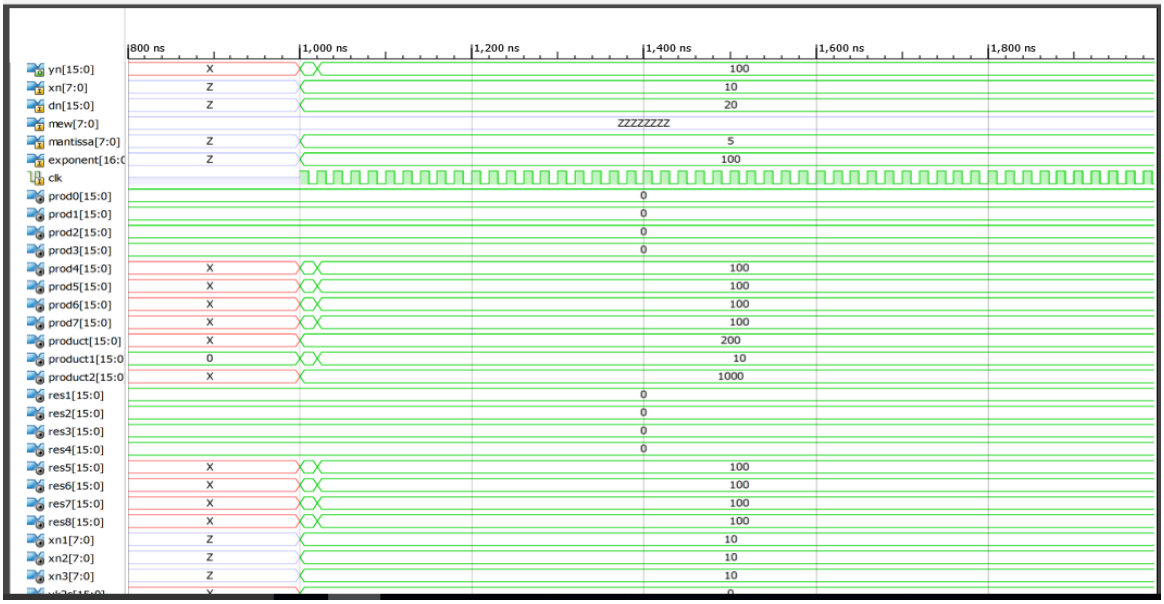

Fig. 6.2 Simulation result of LMS adaptive filter using vedic multiplier

DEVICE UTILIZATION SUMMARY (ESTIMATED VALUES) **Target Device: xc3e50-5pq208**			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	255	768	33%
Number of Slice Flip Flops	32	1536	2%
Number of 4 input LUTs	454	1536	29%
Number of Bounded IOBs	41	124	33%
Number of GCLKs	1	8	12%

Table 1.2: Device Utility Summary for 8 bit LMS Adaptive Filter using Carry Slave Multiplier

VII. Conclusion

The LMS Adaptive filter is being highly adaptive with the changing environment and is capable of providing the best result in the estimation of the time varying parameter. Also the Vedic multiplier which is based on an algorithm Urdhava Tiryakbhyam of ancient Indian Vedic Mathematics provides the best result in terms of speed. Among Adaptive Least Mean Square FIR using Vedic multiplier and Adaptive Least Mean Square FIR using Carry slave multiplier the performance of LMS Adaptive filter using Vedic multiplier was found more satisfactory in terms of processing speed. The Filter is coded using Verilog HDL and further it is Synthesized using the tool Xilinx9.1i. The presented Adaptive Least Mean Square FIR filter system is responsible for providing the best solution for realization and autonomous adaptation of FIR filters, and may be utilized in various communicational techniques and DSP processes such as Signal Channel Equalization, Noise-Cancellation, Echo-cancellation and unknown System identification for adaptive and changing environment.

References

[1]. Pramod Kumar Meher, Senior Member, IEEE, and Sang Yoon Park, Member, IEEE, "Critical-Path Analysis and Low-Complexity Implementation of the LMS Adaptive Algorithm", IEEE Transactions on circuits and systems—I: Regular papers,Vol. 61, no. 3, March 2014.

[2]. Pramod Kumar Meher, Senior Member, IEEE, and Sang Yoon Park, Member, IEEE, "Area-delay-power efficient fixed-point LMS adaptive filter with low adaptation-delay," Trans. Very Large Scale Integr. (VLSI) Signal Process. [Online]. Available: http://ieeexplore.ieee.org,February 2014.

[3]. S. S. Kerur, Prakash Narchi, Jayashree C N, Harish M Kittur and Girish V A, "Implementation of Vedic Multiplier for Digital Signal Processing", International Conference on VLSI, Communication & Instrumentation (ICVCI) 2011 Proceedings published by International Journal of Computer Applications® (IJCA)

[4]. K. Mayyas, Member, IEEE ," Performance Analysis of the Deficient Length LMS Adaptive Algorithm", IEEE Transactions on Signal processing, Vol. 53, no. 8, August 2005.

[5]. Emilio Soria, Javier Calpe, Member, IEEE, Jonathon Chambers, Senior Member, IEEE, Marcelino Martínez, Gustavo Camps, Member, IEEE, and José David Martín Guerrero, " A Novel Approach to Introducing Adaptive Filters Based on the LMS Algorithm and Its Variants", IEEE Transactions on Education, Vol. 47, no. 1, February 2004.

[6]. Onkar Dabeer, Student Member, IEEE, and Elias Masry, Fellow, IEEE, "Analysis of Mean-Square Error and Transient Speed of the LMS Adaptive Algorithm", IEEE Transactions on Information Theory, Vol. 48, no. 7, July 2002

[7]. Lan-Da Van, Associate Member, IEEE, and Wu-Shiung Feng, Senior Member, IEEE, "An Efficient Systolic Architecture for the DLMS Adaptive Filter and Its Applications", IEEE Transactions on Circuits and Systems—II: Analog and Digital Signal Processing, Vol. 48, no. 4, April 2001.

[8]. Saeed Gazor , "Prediction in LMS-Type Adaptive Algorithms for Smoothly Time Varying Environments", IEEE Transactions on Signal Processing, Vol. 47, no. 6, June 1999.

[9]. Shamim Akhter, "VHDL Implementation of Fast NXN Multiplier Based on Vedic Mathematics", Jaypee Institute of Information Technology University, Noida, 201307 UP, INDIA, 2007 IEEE.

[10]. Purushottam D. Chidgupkar and Mangesh T. Karad, "The Implementation of Vedic Algorithms in Digital Signal Processing", Global J. of Engng. Educ., Vol.8, No.2 © 2004 UICEE Published in Australia.

[11]. Himanshu Thapliyal and Hamid R. Arabnia, "A Time-Area- Power Efficient Multiplier and Square Architecture Based On Ancient Indian Vedic Mathematics", Department of Computer Science, The University of Georgia, 415 Graduate Studies Research Center Athens, Georgia 30602-7404, U.S.A.

[12]. E. Abu-Shama, M. B. Maaz, M. A. Bayoumi, "A Fast and Low Power Multiplier Architecture", The Center for Advanced Computer Studies, The University of Southwestern Louisiana Lafayette, LA 70504

[13]. Himanshu Thapliyal and M.B Srinivas, "VLSI Implementation of RSA Encryption System Using Ancient Indian Vedic Mathematics", Center for VLSI and Embedded System Technologies, International Institute of Information Technology Hyderabad-500019, India.

[14]. Abhijeet Kumar, Dilip Kumar, Siddhi, "Hardware Implementation of 16*16 bit Multiplier and Square using Vedic Mathematics", Design Engineer, CDAC, Mohali.