THE INTERNATIONAL JOURNAL OF SCIENCE & TECHNOLEDGE

Adaptive FIR Filter with High Throughput and Low Power Consumption

Nilesh S. Satpute
M. Tech. VLSI Design, Department of Electronics and Communication,
G. H. Raisoni Academy of Engineering and Technology, Nagpur, India
Sanjay B. Tembhurne
Professor, M. Tech. VLSI Design, Department of Electronics and Communication,
G. H. Raisoni Academy of Engineering and Technology, Nagpur, India
Vipin S. Bhure
Professor, M. Tech. VLSI Design, Department of Electronics and Communication,
G. H. Raisoni Academy of Engineering and Technology, Nagpur, India

Abstract: *In this manuscript, an adaptive fir filter for power efficient, area efficient and high throughput design will be delegating using distributed arithmetic (DA). DA is a bit serial computational action and uses set of smaller dynamic parallel look up tables (LUTs), equivalent concurrent realization of filtering weight update proposal for improving the throughput rate. To reduced area complexity, sampling period and critical path, the conditional carry save accumulation of shift accumulator using 10 Transistors circuitry will used in placed of conventional adder based shift accumulation. The Least mean square (LMS) algorithm is introduced to update weight and decline the mean square root error between desired and expected output. For the diminution in power consumption of proposed design, the two separate clocks are introduced; one bit clock for carry save accumulation and it is fastest clock whereas slower for all other computations. The proposed design will include no. of multiplexer, very less no. of LUTs and full adder using half adders.*

Keywords: *Adaptive filter, LUTs, Distributed arithmetic (DA), least mean square (LMS) algorithm.*

1. Introduction

ADAPTIVE FIR filter is a system with a linear filter that has a transfer function managed by variable parameters and a means to adjust those parameters according to a suitable algorithm. Because of the convolution of the optimization algorithms, most adaptive FIR filters are digital filters.

Why Adaptive filters?

Because some parameters of the desired processing operation are not known in advance or are varying.

Adaptive filters are widely used in several digital signal processing applications. The tapped-delay line finite impulse response (FIR) filter whose weights are updated by the famous Widrow–Hoff least mean square (LMS) algorithm is the most popularly used adaptive filter not only due to its simplicity but also due to its satisfactory convergence performance [1]. The direct form configuration on the forward path of the FIR filter results in a long critical path due to an inner-product computation to obtain a filter output. Therefore, when the sampling rate of input signal has a high, it is necessary to lessen the critical path of the structure so that the critical path could not beat the sampling period. In current years, without multiplier DA-based system [2] has gained significant popularity for its high-throughput processing potential and reliability, which result in cost-effective and area–time efficient computing structures. Hardware-capable DA-based design of adaptive filter has been recommended by Allred *et al.* [3] using two separate lookup tables (LUTs) for filtering and weight update. Guo and De Brunner, have improved the design in [3] by using only one lookup table for filtering as well as weight updating. However, the structures in do not support high sampling rate since they involve several cycles for lookup table updates for each new sample. In a recent paper, we have proposed a resourceful architecture for high-speed DA-based adaptive filter with very low alteration delay [6].

This proposes a novel DA-based design for lesser power as well as area, and very high-throughput pipelined realization of adaptive FIR filter with very low adaptation time delay.

The assistance of this brief are as follows.

- Throughput rate is extensively amplified by a parallel LUT update.
- Extended improvement of throughput is achieved by concurrent implementation of filtering and weight updating.
- In this, uses a conditional carry-save accumulation of signed partial inner products to reduce the sampling period instead of Conventional adder-based shift accumulation. The proposed signed carry-save accumulation also helps to lessen the area complication of the proposed design.
- By using a fast bit clock for carry-save accumulation reduction of power consumption is achieved, and for all other operations a much slower clock is introduced.
- An auxiliary control unit for address generation, which is not necessary in the proposed configuration.

1.1. Example as Application

The recording of a heart beat (an ECG), may be despoiled by noise from the mains supply. The input frequency of the power supply and its magnitude may vary from moment to moment. For removing such noise, the notch filter can be used at the mains supply frequency and its surrounding area, but it could be exceptionally degrade the quality of the heart beat recorder since the heart beat reading would also likely have frequency components in the cutoff range.

To avoid this potential loss of heart beat reading or information, an AFIR filter could be used. The AFIR filter would take input from both that is the mains as well as patients and thus it would be easily track the actual frequency components of the distortion and then removing the noise from the recording data. Such technique which is used for designing of adaptive filter generally allows for a filter with a lesser rejection range, so in this case, the output signal has more accuracy in medical systems and other purposes as well.

2. Literature Review

2.1. AUTHOR: Sang Yoon Park, IEEE Transactions on circuits and systems-II: Express Briefs, Vol.60, June 2013.

TITLE**:** "Low power, High Throughput, and Low-Area Adaptive FIR Filter Based on Distributed Arithmetic"

According to him, A novel pipelined architecture of adaptive FIR filter based on DA, the throughput rate for his design is significantly increased by parallel LUT(lookup table) update and concurrent realization of filtering and weight-update operations. The conventional adder-based shift accumulation for distributed arithmetic -based inner-product computation is modified by conditional signed carry-save accumulation in order to decrease the area complexity and the period of sampling. In his proposed design, by using a fast bit clock for carry-save accumulation the power consumption reduced, but a much slower clock for all other operations.

Designs	Length	Throughput	Power (mW)	Area (sq. µm)
Previous	N=16	357.14	12.08	29667
	N=32	357.14	24.00	59244
Author	N=16	312.5	9.41	18264
	N-32	304.87	17.56	36126

Table 1

2.2. AUTHOR: G. Selvapriya PG Scholar, Mr. S. Karthick, Assistant Professor (Sr. G), Department of ECE, Bannari Amman Institute of Technology, India,

TITLE: "High Throughput, Low Area, Low Power Distributed Arithmetic Formulation for Adaptive Filter"

According to them, DA formulation employed for two separate blocks weight update block and filtering operations requires larger area and is not suited for higher order filters therefore causes reduction in the throughput. These problems have been overcome by efficient distributed formulation of Adaptive filters. LMS adaptation performed on a sample-by-sample basis is replaced by a dynamic LUT update using a weight update scheme. Further, parallel LUT update and concurrent implementation of filtering and weight-update operations significantly increases throughput rate. Adder based shift accumulation for inner product computation replaced by conditional signed carry-save accumulation reduces the sampling period and area complexity. Fast bit clock for carry-save accumulation to decrease power consumption. It involves the same no. of multiplexers, half the no. of adders and less significant LUT, compared to the earlier DA-based system.

Parameters	Throughput (per µs)	Power(mW)	Area (Sq. µs)
Previous	77.39	21.16	20347
Them	300.5	10.42	17159

Table 2

2.3. AUTHOR: Aishwarya C, PG scholar and Mr. Vijaybhaskar R, Assist. Prof. Anna University, Regional Centre, Coimbatore, International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682 Vol.23, No23, March 2014.

TITLE: "Enhanced Pipelined Architecture for Adaptive FIR Filter Based on Distributed Arithmetic"

According to them, The throughput rate is increased by parallel DA table update and concurrent filtering and weight-update operations. To reduce critical path and area complexity, carry save accumulation is used for inner product computation instead of conventional adder-based shift accumulation.

Two different clocks are used in the proposed design which may result in reduced power consumption. A fast bit clock is used for carry save accumulation and a separate clock for all other operations. Her design was coded in VHDL for filter lengths N=4 and N=16 and simulated results are observed to verify the architecture.

Designs	Length	Area (sq. µs)	Power(mW)
Previous	N=16	18264	9.41
Author	N=16	17220	8.49

Table 3

2.4. AUTHOR: K. Jebin Roy, R. Ramya, International Journal of Scientific and Research Publications, Volume 4, Issue 3, March 2014

TITLE: "Low-power and low-area adaptive FIR Filter based on distributed arithmetic And LMS algorithm"

In this manuscript, an unusual adaptive FIR filter using distributed arithmetic (DA) for area efficient design is implemented. Distributed arithmetic is bit-serial calculation action and uses LUTs describe and equal implementation of filtering and weight-update operations to appliance very high throughput filter rates irrespective of the filter length. For DA-based inner product computation based on the full adder conditional signed carry save accumulation is swapped and realize by using 10 transistor full adder based carry save accumulation, with the purpose of the proposed system, it can decrease the power consumption and difficulty of area. The least-mean-square (LMS) algorithm adaptation is functioned to update the weight and abate the mean square error between the assessed and chosen output. The weight increment design based adder/subtractor cells is replaced by carry save adder in order to lessen area difficulty. It comprises of less significant LUT, MUXs and half the no. of adders contrasted to the present DA-based system.

Designs	Length	No. of LUTs	Power(mW)
Previous	N=16	1212	9.60
Author	N=16	661	8.60

Table 4

3. Research Methodology to be Employed

3.1. Adaptive LMS Algorithm

On each clock cycle, the LMS algorithm calculates an error value as well as a filter output which is equal to the difference between the current filter output and the preferred response. The estimated error is then used to update the filter weights in every training cycle. The values of filter during the nth iteration are restructured according to the following equations:

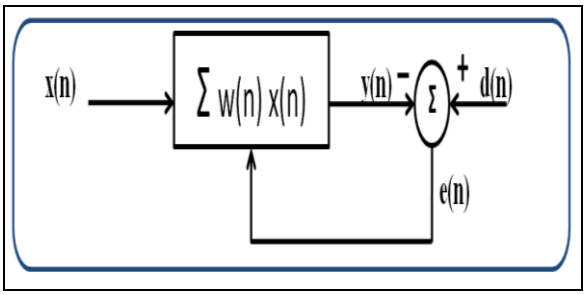

Figure 1: LMS Algorithm

$w(n + 1) = w(n) + \mu * e(n) * x(n)$…….(a)

Where,

$e(n)=d(n)-y(n)$……..(b)

$y(n)=w^{qT}(n)*x(n)$…….(c)

At the nth iteration,

the weight vector w(n) and the input vector x(n) are respectively given by,

$x(n) = [x(n), x(n-1), \ldots, x(n-N+1)]^T$

.....(d)

$w(n) = [w0(n), w1(n), \ldots, wN-1(n)]^T$

.....(e)

d(n) is the desired response,

y(n) is the filter output of the nth iteration,

e(n) is the calculated error during the nth iteration, which is used to update the values, μ is the convergence factor,

N is the length of filter.

3.2. Distributed Arithmetic

The LMS adaptive FIR filter, in each cycle, wants to perform an inner-product computation but the most of the critical paths are contributed because of it; these calculations performed by system are known as DA (Distributed Arithmetic).

Distributed Arithmetic is a system which is bit-serial in nature. It can therefore show to be low down. It turns out when the no. of essentials in a vector is same as the word size, Distributed Arithmetic is fast and it replaces the explicit calculations like multiplications by ROM look-ups an capable technique to realize on FPGAs (Field Programmable Gate Arrays). In DSP hardware designs, DA can be save up to 80% of area.

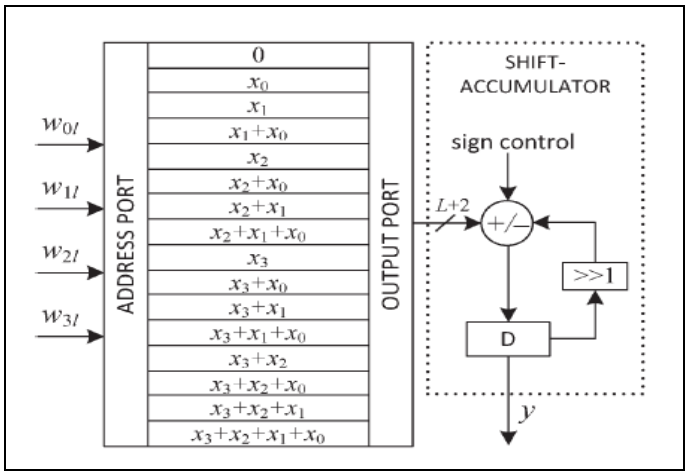

Figure 2: Proposed DA based implementation for 4-point inner product

4. Proposed Da-Based Adaptive Filter Structure

The calculations of adaptive FIR filters of huge orders requires to be decomposed into tiny filtering blocks since Distributed Arithmetic based execution of inner product of long vectors requires a very large lookup table [3]. So, here illustrating the proposed Distributed Arithmetic-based structures of small and large order LMS FIR filters independently in the next sections.

4.1. Small-Order Adaptive Filter Proposed Structure

For length $N = 4$, The DA-based proposed structure of adaptive FIR filter is shown in Fig. 4. It has a weight-increment block along with additional circuits for the calculations of error value $e(n)$, a 4-point inner product block and the barrel shifters control word t.

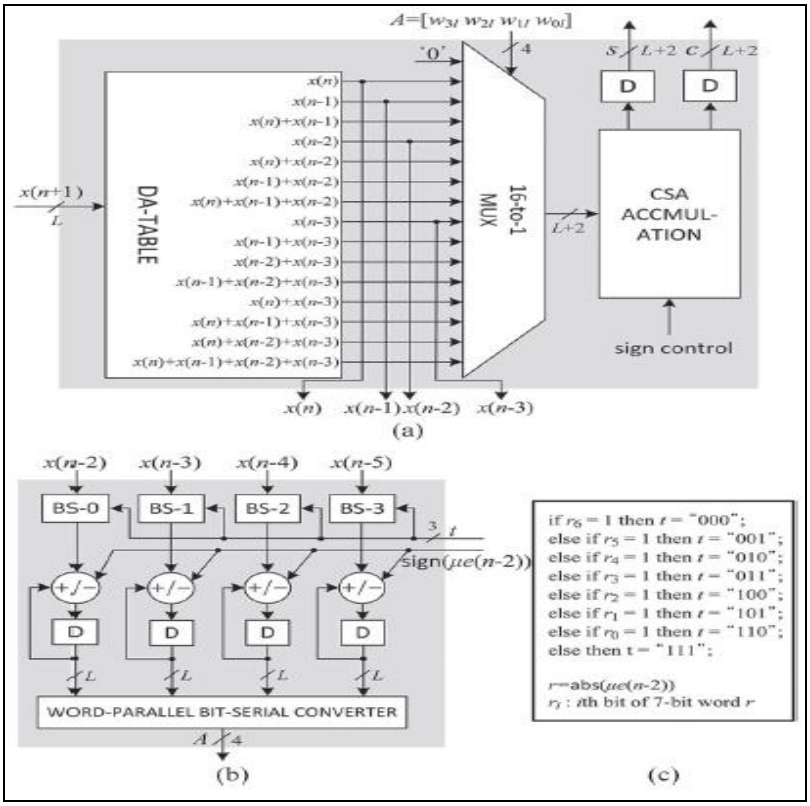

Figure 3: (a) 4-point inner-product Structure. (b) the weight-increment Structure for N = 4. (c) Logic for control word generation for the barrel shifter for L = 8.

The 4-point inner-product building block [shown in Fig. 3(a)] includes a Distributed Arithmetic table consisting of 15 registers array which saves the partial inner products yl for $0 < l \leq 15$ and multiplexer used to choose the one of the content of those registers. Bit slices of weights $A = \{w3l\ w2l\ w1l\ w0l\}$ for $0 \leq l \leq L - 1$ are giving to the multiplexer as control in LSB-to-MSB order, and the carry-save accumulator take input from the output of the multiplexer (shown in Fig.2). After L bit cycles, the carry-save accumulator accumulates all the partial inner products and produce a sum and a carry word of size (L+2) bit each. The sum and carry words are shifted added with an input carry "1" to produce filter output which is then subtracted from the required output $d(n)$ to get the error $e(n)$.

As in the case in [Fig3], all the bits of the error excepting the most important one are disregarded, such that multiplication of input by the error is implemented by a right shift during the no. of locations specified by the no. of primary zeros in the error magnitude. The error magnitude is decoded to produce the barrel shifter control word *t*. The logic for the production of barrel shifter control word is shown in Fig. 3(c). The convergence factor μ is usually taken to be (1/*N*).
We have taken μ =1/*N*.
However, one can take μ as $2^i/N$, where *i* is any small integer value. The no. of shifts '*t*' is enlarged by '*i*' and the barrel shifter input is pre-shifted by '*i*' locations to decrease the hardware difficulty.
Fig. 3(b) shows a system for *N* = 4 consist of four barrel shifters and four adder/subtractor cells called as the weight-increment unit.
The barrel shifter shifts the different input values for $k = 0, 1, \ldots, N-1$ by proper no. of locations. The barrel shifter yields the preferred increments to be subtracted from or added with the current weights. The error sign bit is used as the control for adder/subtractor cells that is when sign bit is zero, the barrel-shifter output is added with or when sign bit is one, the barrel-shifter output is subtracted from the content of the corresponding current value in the weight register.

4.2. Large-Order Adaptive FIR Filter Proposed Structure.
The inner-product calculations can be decomposed into *N/P* (assuming that *N* = *PQ*) small adaptive filtering system of filter length *P* as

$$y = \sum_{k=0}^{P-1} w_k x_k + \sum_{k=P}^{2P-1} w_k x_k \ldots + \sum_{k=N-P}^{N-1} w_k x_k$$

Each of these *P*-point inner-product calculation blocks will accordingly have a weight-increment unit to update *P* weights. The proposed structure for *N* = 16 and *P* = 4 is consists of four inner-product blocks of length *P* = 4. The (*L*+2)-bit sums and carry produced by the four blocks are added by two separate binary adder trees. The Output of 4-point inner-product blocks is the addition of Carry-in bits with sum words. Since The sum words is half of the carry words weight, two carry-in bits are place at the first level binary adder tree of carry words as input carry, which is equal to insertion of four carry-in bits to the sum words. Assuming that μ = 1/*N*, we truncate the four Least Significant Bits of *e*(*n*) for *N* = 16 to compose the sign-magnitude word length separator be *L* bit. It should be pointed that the truncation does not involve at the performance of the adaptive FIR filter very much since the proposed design requires the place of the most significant one of $\mu e(n)$.

5. References

1. Sang Yoon Park, June 2013, "Low power, High Throughput, and Low-Area Adaptive FIR Filter Based on Distributed Arithmetic,"IEEE Transactions on circuits and systems-II: Express Briefs,Vol.60.
2. G.Selvapriya, M.Mano, K.RekhaSwathiSri, PG Scholars, Mr S.Karthick, Assistant Professor (Sr. G), Department of ECE, Bannari Amman Institute of Technology, India, "High Throughput, Low Area, Low Power Distributed Arithmetic Formulation for Adaptive Filter", International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014
3. Aishwarya C PG scholar and Mr. Vijaybhaskar R Assist. Prof. Anna University, RegionalCentre, Coimbatore, "Enhanced Pipelined Architecture for Adaptive FIR Filter Based on Distributed Arithmetic," International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682 Vol.23, No23, March 2014.
4. K. Jebin Roy, R. Ramya, "Low Power and Low Area Adaptive FIR Filter based on DA and LMS Algorithm" International Journal of Scientific and Research Publications, Volume 4, Issue 3, March 2014
5. D. J. Allred, H. Yoo, V. Krishnan, W. Huang, and D. V. Anderson, Jul. 2005,"LMS adaptive filters using distributed arithmetic for high throughput," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 52, no. 7, pp. 1327–1337.
6. R. Guo and L. S. DeBrunner, Sep. 2011, "Two high-performance adaptive filter implementation schemes using distributed arithmetic," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 58, no. 9, pp. 600–604.