

# A Comparative Analysis of FIR Filters using Distributed Arithmetic Formulation

Anushree Ashokan  
Department of Electronics  
and Communication  
SIRT, Bhopal

Himanshu Nautiyal  
Department of Electronics  
and Communication  
SIRT, Bhopal

Pallav Parmar  
Department of Electronics  
and Communication  
SIRT, Bhopal

## ABSTRACT

In this paper we are comparing the FIR filters by distributed arithmetic (DALUT) design using sharing lookup table and reconfigurable implementation of distributed arithmetic (RIDA) method. DA-based design with look-up table (LUT)-sharing technique for the computation of filter outputs and weight-increment terms of block least mean square BLMS algorithm. Besides, it offers significant saving of adders which constitute a major component of DA-based structures. While in the reconfigurable implementation of distributed arithmetic (DA) for post-processing applications is described. The input of DA is received in digital form and its analog coefficients are set by using the floating-gate voltage references. This is a major advantage of the DALUT structure for reducing its area delay product (ADP); particularly, when a large order adaptive digital Filter (ADF) is implemented for higher block-sizes.

## General Terms

Finite Impulse Response Filter, Distributed Arithmetic (DA) Technique

## Keywords

FIR Filter, Distributed Arithmetic (DA) Technique, Look Up Table (LUT), Multiply and Accumulate (MAC).

## 1. INTRODUCTION

Adaptive digital filters (ADFs) are broadly used in numerous signal-processing applications, such as echo deletion, system identification, noise elimination and channel equalization etc. [1]. Amongst the prevailing ADFs, least mean square (LMS)-based finite impulse response (FIR) adaptive filter is the supreme prevalent one due to its inherent simplicity and suitable convergence performance and characteristics. Mohanty et al. [1] had suggested a DA-based design that used a novel look-up table (LUT)-sharing method for the calculating the filter outputs and weight-increment terms of BLMS algorithm. The count of adders of the proposed structure does not rise with escalation of the block size; and the number of flip-flops used will not be reliant on the block-size. This is a foremost advantage of the proposed structure for minimizing its area delay product (ADP); particularly, when a large order ADF is made for higher block-sizes. [2] A reconfigurable implementation of distributed arithmetic (DA) for post-processing applications. The input of DA is accepted in digital form and its analog coefficients are calculated by means of the floating-gate voltage references.

The derivation based filters are mainly designed for a signal attenuated by a "noise" like signal so that the value of least square error stuck between the filtered output received and the preferred output is reduced [4]. The filter is contrasted with both the Levinson and Widrow filters. The least squares error

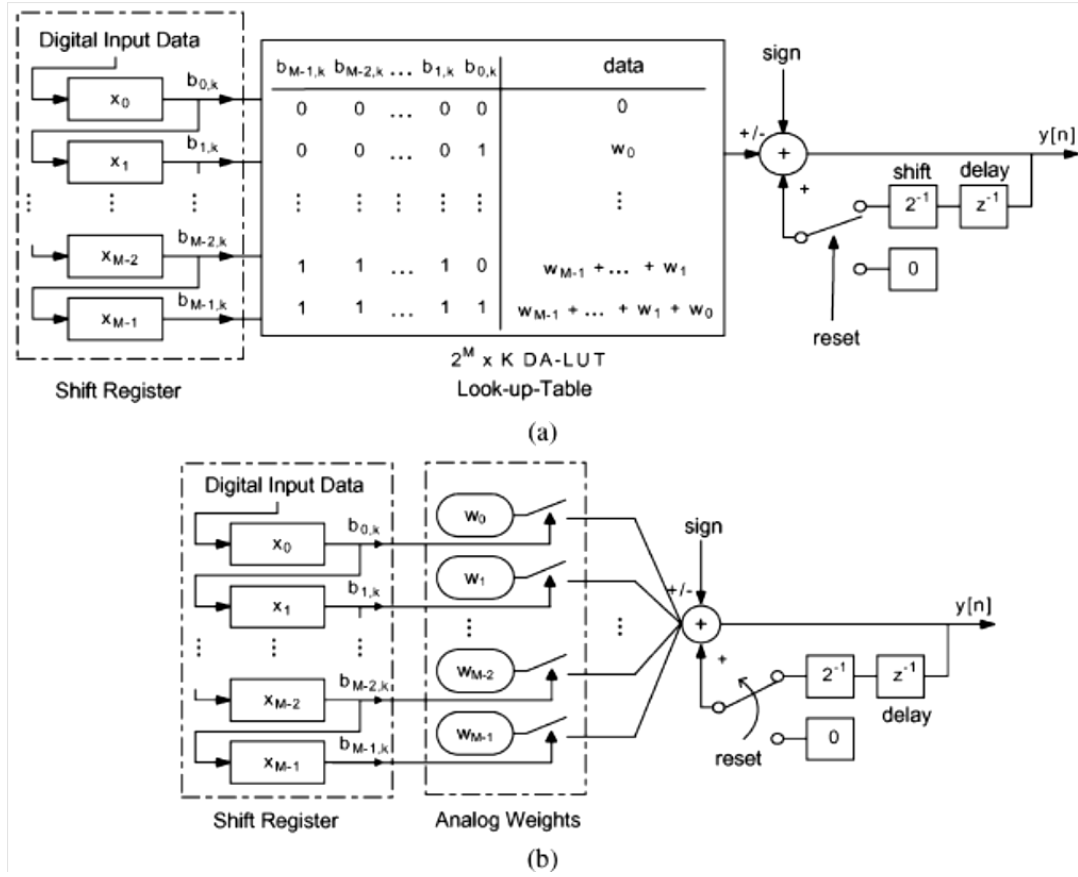
(LSE) and mean square error (MSE) based filters are designed for optimum best delay deconvolution [5]. In the partition-based finite impulse response–infinite impulse response (FIR-IIR) filters, the channel zeros are separated into two regions. The first region comprises with the selected channel zeros within a unit circle, and the second region is composed of the left over channel zeros outside the unit circle. The methods used for partitioning are named as, optimum partitioning and ring-based partitioning. Resultant FIR-IIR filters are compared with the FIR and FIR-IIR unit circle best delay inverse filters in terms of their LSE. Furthermore, they are at all times causal and stable, making them appropriate for real-time implementations [5].

Ozalevli, [6] presented a reconfigurable implementation of distributed arithmetic (DA) for post-processing applications. The input of DA is received in digital form and its analog coefficients are set by using the floating-gate voltage references. Wong, [7] presents a technique for approximating finite-impulse-response (FIR) filters with infinite-impulse-response (IIR) structures through extending the vector fitting (VF) process, used broadly for continuous-time frequency-domain rational approximation, to its discrete-time counterpart called *VFz*. *VFz* directly calculates the candidate filter poles and iteratively rearranged them for more better approximation.

Pei et al. put forward a cepstrum-based method to design finite-and infinite-impulse-response (IIR) fractional-delay (FD) filters. The maximal-flatness criteria on frequency responses are formulated as a system of linear equations to unravel the truncated complex cepstrum. Under a fixed filter order, the set of regularized complex cepstrum requires to be calculated once and stored, and the specific set for an arbitrary FD is received by simply multiplying the stored set with the delay value. Lei et al. [9] present a discrete-time hybrid-domain vector fitting algorithm, called *HD-VFz*, for the IIR approximation of FIR filters with an arbitrary combination of time- and frequency-sampled responses. The main routine comprises a two-step pole refinement process depending on a linear least-squares solve and an eigenvalue problem. By the means of a hybrid-domain data approximation and digital partial fraction basis with relative stability concern, *HD-VFz* shows fast computation and amazing fitting accuracy in both time and frequency domains.

The rest of this paper is arranged as follows: Mathematical formulation is presented in Section II. The new-LUT update scheme is discussed in Section [3], and the planned structure for DA-based BLMS ADF is presented in Section IV. Conclusion is presented in Section V.

The basic architecture of the system:



**Fig. 1. Basic DA hardware architecture.  $b_{ij}$  is the input bit for  $k_{th}$  cycle of operation and  $y[n]$  is the output. (a) Digital implementation. (b) Proposed hybrid mixed-signal implementation using digital input data and stored analog weights. Digital input data is processed in the analog domain [2].**

## 2. HYBRID DA ARCHITECTURE [2]

The hybrid DA architecture consists of the following components, which are a 16-bit shift register, an array of tunable FG voltage references (epots), inverting amplifiers (AMP), and sample-and-hold (S/H) circuits, as shown in Fig. 2. The timing of the digital data and control bits guides the DA computation and is illustrated in Fig. 3. Digital inputs are supplied to the system by in a specific manner with the help of a serial shift register.

The operation of DA can be derived from the inner product equation as follows:

$$y[n] = \sum_{i=0}^{M-1} x[n-i]w[i] \quad (1)$$

In the case of FIR filtering,  $x$  is the input vector and  $w$  is the weight vector.

$$y[n] = - \sum_{i=0}^{M-1} w_i b_{i0} + \sum_{i=1}^{K-1} 2^{-j} \sum_{i=0}^{M-1} w_i \quad (2)$$

The digital input words present here are the digital bits,  $b_{ij}$  in (2), which selects the epot voltages to form the precise sum of weights which are necessary for the DA computation at the  $i$ th bit. In the shift register the clock frequency is the value which is affected by the input data precision and the length of the filter  $M$  and is equal to  $M.K$  times the sampling frequency. As soon as the  $j_{th}$  input word was loaded into the top shift register, firstly the data is processed serially and the data from

this register is latched,  $K$  times the sampling frequency. In a condition where the amount of area that was used was not a design concern, then ideally an  $M$ -tap FIR filter should have  $M$  shift registers and a clock that is  $K$  times faster than the sampling frequency. The epots are used in DA so that it stores the value of analog weights. The selected weights are then added by using a charge amplifier structure which has same size capacitors, and a two-stage amplifier AMP. The epot voltages in addition with the rest of the analog voltages in the system are compared to a reference voltage, 2.5V. When the RESET signal is enabled the addition operation is carried out through the means of inverting amplifier. The output of inverting amplifier becomes equal to the negative sum of the selected weights for  $Cin_i = Cfb_{amp1}$ . For the first computational cycle, the outcome of the addition stage is the summation  $\sum_{i=0}^{m-1} w_i b_i(k-1)$ , in (2), which signifies the addition of weights for the LSBs of the digital input data. For the DA computation, a delay, an invert and a divide-by-two operation are used for the feedback path. For that purpose, S/H circuits S/H1 and S/H2 and inverting amplifiers AMP1 and AMP2 are used for the implementation. The S/H circuits are used for feedback as it saves the present output of amplifier and use it for the next cycle of computation. Clocks CLK1 and CLK2 are used for holding the analog voltage, to avoid overlapping while the next stream of digital data is introduced to the addition.

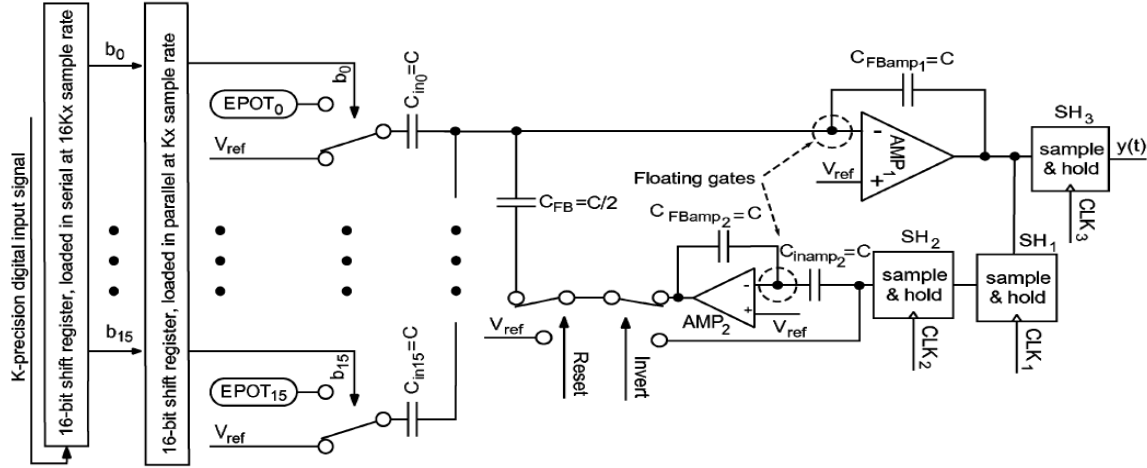


Fig. 2. Implementation of the 16-tap hybrid FIR filter.  $b$  is the input bit for  $j_{th}$  cycle of operation and  $y(t)$  is the output. Epots store the analog weights. S/Hs, are used to obtain the delay and hold the computed output voltage [2].

### 3. DISTRIBUTED ARITHMETIC BASED ON LOOK UP TABLE (DALUT)

In this paper, we have formulated the DA-BLMS procedure for sharing of LUTs for the calculation of filter output and weight-increment terms.

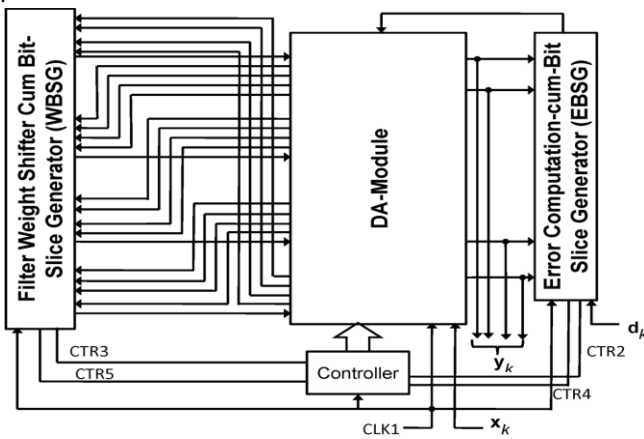


Fig. 3. DA-based structure for implementation of BLMS adaptive FIR filters (for  $N = 16$  and  $L = 4$ ), where  $x_k = \{x(4k), x(4k-1), x(4k-2), x(4k-3)\}$ ,  $y_k = \{y(4k), y(4k-1), y(4k-2), y(4k-3)\}$  and  $d_k = \{d(4k), d(4k-1), d(4k-2), d(4k-3)\}$

The key contributions of this paper are:

- DA-based formulation of BLMS procedure where both convolution operation to calculate filter output and correlation operation to calculate weight-increment term could be achieved by using the same LUT.
- This procedure helps in saving external logic and power consumption, by minimization of no. of LUT words used as per the outputs

We have derived a DA-based structure for BLMS-ADF using the proposed DA m-formulation and a novel LUT updating scheme. The most noticeable feature of the proposed scheme is that the number of adders wanted by the structure does not

increase proportionately with filter order, and the count of flip-flops essential for the structure is not dependent on the block-size. Apart from that, the proposed design has considerably a smaller amount of LUT access than the existing DA-LMS structure for greater block-sizes.

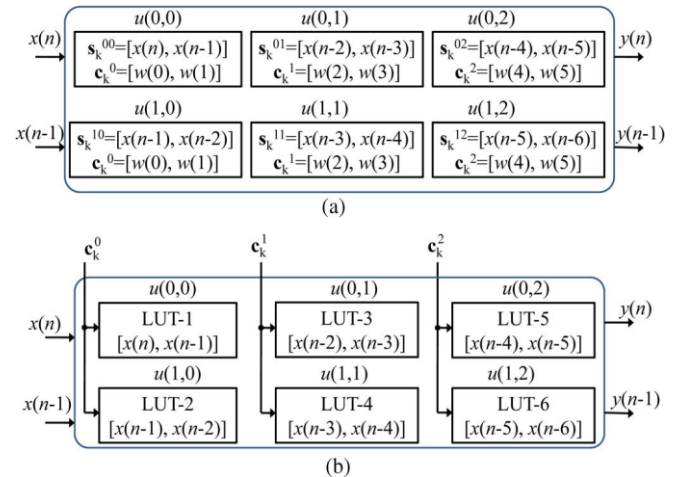


Fig. 4i. (a) Inner-products of FIR filter of length  $N = 6$ , and block-size  $L = 2$ . The input-vectors  $s_k^{i,j}$

corresponding to inner-product  $u(i, j)$  is shown inside the box. (b) LUT arrangement for DA-based computation of the FIR filter of,  $N = 6$ , and  $L = 2$ . Each LUT here stores  $2^2$  possible values of partial inner-product  $u(i, j)$  of input vector  $s_k^{i,j}$  and bit-vector of  $(c_k^j)$  of length  $L$ , for  $0 \leq i \leq 1$  and  $0 \leq j \leq 2$

Proposed DA-BLMS structure is comprised of one DA-module, one error bit-slice generator (EBSG) and one weight-update cum bit-slice generator (WBSG). WBSG updates the filter weights and generates the required bit-vectors in accordance with the DA-formulation. EBSG computes the error block according to (3) and generates its bit-vectors. The DA-module updates the LUTs and makes use of the bit-vectors generated by WBSG and EBSG to compute the filter output and weight-increment terms according to (15) and (16).

#### 4. RESULT ANALYSIS

The measurement results are obtained from the chips that were fabricated in a 0.5-  $\mu$ m CMOS process. This CMOS process was chosen for prototyping and to prove the proposed concept since the programming and retention characteristics of the CMOS floating-gate devices are well characterized in this process. In this process, it was shown that the stored charge on the floating gate drifts around  $10^{-3}$ % over the period of 10 years at 25 C [15], and this makes the use of floating-gate transistors suitable for the DA implementation [2].

**Table 1 Comparison of the various parameters**

Parameters	Algorithms	
	Mohanty's method[1]	Ozalevi's method[2]
Number of filter taps	16	16
Used chip area	0.401mm <sup>2</sup>	1.125mm <sup>2</sup>
Power consumption	2.363 mW	16 mW

#### 5. CONCLUSION

It is shown that DA processing declines the offset as the exactitude of the digital input data escalates[2]. Also, the gain error in this implementation is primarily initiated by the two inverting stages (implemented using AMP and AMP), and can be reduced using special layout methods only at these stages. The measurement result explains that the output signal of the filter tracks the ideal response very precisely. The programmable analog coefficients of this filter will support the implementation of adaptive systems that can be used in areas such as an adaptive noise deletion and adaptive equalization. [2]. [1] a DA formulation of BLMS algorithm where both convolution and correlation are performed using a shared LUT for the calculation of filter outputs and weight increment terms, respectively. This results in a significant saving of LUT words and the no. of adders which is the main set up the chief hardware components in DA-based computing structures.

#### 6. REFERENCES

- [1] B. K. Mohanty and P. K. Meher, "A High-Performance Energy-Efficient Architecture for FIR Adaptive Filter Based on New Distributed Arithmetic Formulation of Block LMS Algorithm," *IEEE Trans. on Signal Process.*, Vol. 61, No. 4, Feb. 15, 2013 pp. 921-932.
- [2] E. Özalevli, W. Huang, P.E. Hasler, and D.V. Anderson, "A Reconfigurable Mixed-Signal VLSI Implementation of Distributed Arithmetic Used for Finite-Impulse Response Filtering," *IEEE Trans. on Circuits and Systems—I*, vol. 55, no. 2, Mar. 2008, pp. 510-521.
- [3] C. R. Guarino, "Adaptive Signal Processing Using FIR and IIR Filters," *Proceedings Of The IEEE*, VOL. 67, NO. 6, JUNE 1979 pp. 957-958.
- [4] T.E. Tuncer and M. Akta, "LSE and MSE Optimum Partition-Based FIR-IIR Deconvolution Filters With Best Delay," *IEEE Trans. on Signal Process.*, vol. 53, no. 10, Oct. 2005, pp.3780-3790.
- [5] E. Ozalevli, W. Huang, P.E. Hasler, and David V. Anderson, "A Reconfigurable Mixed-Signal VLSI Implementation of Distributed Arithmetic Used for Finite-Impulse Response Filtering," *IEEE Trans. on Circuits and Systems—I: Regular Papers*, vol. 55, no. 2, Mar. 2008, pp. 510-521.
- [6] N. Wong and C. Lei, "IIR Approximation of FIR Filters Via Discrete-Time Vector Fitting," *IEEE Trans. On Signal Process.*, vol. 56, no. 3, Mar. 2008, pp.1296-1301.
- [7] S.C Pei and H.S Lin, "Tunable FIR and IIR Fractional-Delay Filter Design Oct.2009, pp2195-2206.
- [8] S.C. Pei and H.S. Lin, "Tunable FIR and IIR Fractional-Delay Filter Design and Structure Based on Complex Cepstrum," *IEEE Trans. on Circuits and Systems—I: Regular Papers*, vol. 56, no. 10, Oct. 2009, pp 2195-2206.
- [9] C.U. Lei and N. Wong, "IIR Approximation of FIR Filters via Discrete-Time Hybrid-Domain Vector Fitting," *IEEE Signal Process. Lett.*, Vol. 16, No. 6, June 2009 533-536