Designing of FBLMS Adaptive filter

Sonali Dhobale
M.Tech VLSI (III Sem)
P.C.E, Nagpur

Prof.(Mrs.) P.J.Suryawanshi
M.Tech VLSI (Co-ordinator)
P.C.E, Nagpur

ABSTRACT
This paper proposes a design and adaptive digital filter using Fast Block Least Mean Squares (FBLMS) adaptive algorithm. The filter structure is based on Distributed Arithmetic (DA), which is able to calculate the inner product by shifting, and accumulating of partial products and storing in look-up table, also the desired adaptive digital filter will be multiplier less. Thus a DA based implementation of adaptive filter is highly computational and area efficient. Furthermore, the fundamental building blocks in the DA architecture map well to the architecture of todays Field Programmable Gate Arrays (FPGA). FPGA implementation results conforms that the proposed DA based adaptive filter can implement with significantly smaller area.

General Terms
Fast Block Least Mean Squares (FBLMS) adaptive algorithm, Distributed Arithmetic (DA), Field Programmable Gate Arrays (FPGA), Look-up-table (LUT).

Keywords
Fast Fourier transform(FFT) , Read Only Memory (ROM) , Inverse Fast Fourier Transform (IFFT),FBLMS , DA.

1. INTRODUCTION
An Adaptive filters are widely used in the area of signal processing such as echo cancellation, noise cancellation, channel equalization for communications and networking systems. Necessity of adaptive filters implementations is growing up in many fields. An adaptive filter is a filter that self-adjusts its transfer function according to an optimizing algorithm. Because of the complexity of the optimizing algorithms, most adaptive filters are digital filters that perform digital signal processing and adapt their performance based on the input signal. Adaptive filter are required when either the fixed specifications are unknown or time invariant filters cannot satisfy the specifications. Adaptive filters are time-varying since their parameters are continually changing in order to meet a performance requirement. The hardware implementation requires various of performances such as high speed, low power dissipation, small chip area and good convergence characteristics. However it is difficult to satisfy these characteristics simultaneously, so efficient algorithms and efficient architectures are desired. The design of adaptive filter algorithm is an important part within the design of adaptive filter. The fast block least mean square (FBLMS) algorithm is one of the fastest and computationally efficient adaptive algorithm since here the process of filtering and adaption is done in frequency domain by using overlap and save method via FFT. The popularity of the FPGA is due to balance that FPGAs provide the designer in terms of flexibility, cost, and time-to-market. In this project, I will be analyzing the design of an FBLMS algorithm based adaptive filter.

2. DISTRIBUTED ARITHMETIC (DA)
DA was first introduced by Croisier et al. and further developed by Peled and Lui. The DA is a direct method for sum of products operations, partial products can pre-compute by difference equation and storing in look-up-table (LUT) contained in memory, input signals are used for addressing. The product can be computed by scaling accumulate of partial products from memory, therefore multipliers do not necessary for this method.
Consider the following inner product of two L dimensional vectors a and x, where a is a constant vector, x is the input sample vector, and y is the result.

$$y = \sum_{k=0}^{L-1} \mathbf{a_k x_k} \qquad (1)$$

Using B-bit 2's complement binary representation scaled such that $|x_k| \leq 1$ produces

$$\mathbf{x_k} = -\mathrm{b}_{k0} + \sum_{n=1}^{B-1} \mathrm{b}_{kn} 2^{-n} \qquad (2)$$

Where b_{kn} are the bits (0 or 1) of x_k , b_{k0} is the most significant bit, and $b_{(B-1)}$ is the least significant bit. Substituting (2) into (1) yields

$$y = \sum_{k=1}^{L-1} \mathbf{a}_k \left[-\mathrm{b}_{k0} + \sum_{n=0}^{B-1} \mathrm{b}_{kn} 2^{-n} \right] \qquad (3)$$

$$= -\sum_{k=1}^{L-1} \mathbf{a_k} \mathrm{b}_{k0} + \sum_{n=1}^{B-1} \left[\sum_{k=1}^{L-1} \mathbf{a}_k \mathrm{b}_{kn} \right] 2^{-n} \qquad (4)$$

The computation in distributed arithmetic is represented by (4). The values of b_{kn} are either 0 or 1, resulting in bracketed term in (4) having only 2^B possible values. Since a is a constant vector, the bracketed term can be recomputed and stored in memory using either lookup table (LUT) or ROM. The lookup table is then addressed using the individual bits of input samples, x_k with the final result y computed after B cycles, regardless of lengths of vectors a and x.

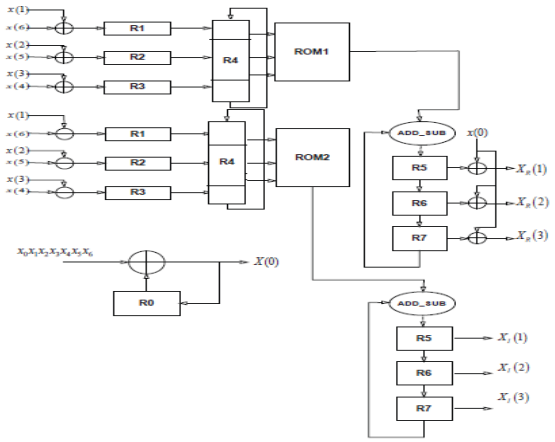

Figure1: Architecture of FFT using DA

3. FBLMS ALGORITHM

Consider a BLMS based adaptive filter, that takes an input sequence x(n), which is partitioned into non-overlapping blocks of length P each by means of a serial-to-parallel converter, and the blocks of data so produced are applied to an FIR filter of length L, one block at a time. The tap weights of the filter are updated after the collection of each block of data samples, so that the adaptation of the filter proceeds on a block-by-block basis rather than on a sample-by-sample basis as in conventional LMS algorithm.

With the j-th block, (j εZ) consisting of x (j P + r) , r ε Zp = 0,1, ..., P -1, the filter coefficients are updated from block to block as,

$$w(j+1) = w(j) + \mu \sum_{r=0}^{P-1} x(jP+r)e(jP+r).....(5)$$

Where $w(j) = [w_0(j)w_1(j).....w_{L-1}(j)]^t$ the tap weight vector corresponding to the j-th block, x (j P + r) = [x (jP +r) x (jP +r -1) ....x (jP + r - L+ l)]^t and e(jP + r) is the output error at n = jP + r , given by,

$$e(jP+r) = d(jP+r) - y(jP+r).....(6)$$

The sequence d (j P + r) is the so-called desired response available during the initial training period and y (jP + r) is the filter output at n = j P + r , given as,

$$y(jP+r) = w^t(j)x(jP+r).....(7)$$

The parameter μ, popularly called the step size parameter is to be chosen as 0 ≤μ≤[2/PtrR] for convergence of the algorithm. For the lth sub-block within the ith block,0 ≤ l ≤K- 1 i.e., for n = jP+r, r = 0,1, ... P-1,j = iK+ l, the filter output y (n) = w^t (j) X (n) is obtained by convolving the input data sequence x(n) with the filter coefficient vector w^t (j) and thus can be realized efficiently by the overlap-save method via M = L+ P -1 point FFT, where the first L -1 points come from the previous sub-block, for which the output is to be discarded. Similarly, the weight P-l update term in (5) above, viz., $\sum_{r=0}^{P-1} x(jP+r)e(jP+r)$ can be obtained by the usual circular correlation technique, by employing M point FFT and setting the last P -1 output terms as zero.

4. PROPOSED IMPLEMENTATION

The major computational block in FBLMS algorithm is FFT/IFFT. Each N-point FFT (and IFFT) requires approximately $N \log_2 N$ real multiplications. The throughput of conventional FBLMS based adaptive filters is limited by the computational complexity involved in computation of FFT/IFFT operations. It is possible to enhance the throughput of such systems by employing some efficient techniques to compute FFT/IFFT.

DA is one of the such techniques, in which, by means of a bit level rearrangement of a multiply accumulate terms, FFT can be implemented without multipliers. There are many fast Fourier transform (FFT) algorithm exists like radix-2, Cooley-Tukey, Winograd, Good-Thomas, Rader etc.

In our proposed architecture of FBLMS algorithm based adaptive filter we replace FFT (and IFFT) block, with an DA based FFT (and IFFT) block. Since the DA based FFT block provides only half of the conjugate symmetric outputs without calculating others (reaming can calculated by conjugating them) and in DA based IFFT block we require to feed only half of the conjugate symmetric coefficients not all, which is not possible in existing FBLMS based adaptive filters since here radix2 based FFT (and IFFT) blocks are employed and in radix2 there is no such facilities are there to calculate only half of the conjugate symmetric outputs hence in this case half of the processed data are redundant. Since in our proposed FBLMS algorithm based adaptive filter we require to calculate only only half of the conjugate symmetric coefficients, under that condition the hardware requirements for our proposed system is approximately half of that of existing one. In our proposed architecture shown in Fig. 3 the computation of frequency domain outputs requires 8($\mathcal{N}$2 + 1) or 8($\mathcal{M}$† + 1) number of multiplication and required number of addition is 16$\mathcal{N}$ +2.5($\mathcal{N}$1+$\mathcal{N}$2)+2, here total number of multiplications are much less than that of required number of multiplications in the existing FBLMS algorithm based adaptive filter at the expanse of increased memory and adder requirement, which drastically reduces the hardware complexity for higher order filters. It results with a adaptive filter which has high throughput and low power dissipation with reduced area requirement.

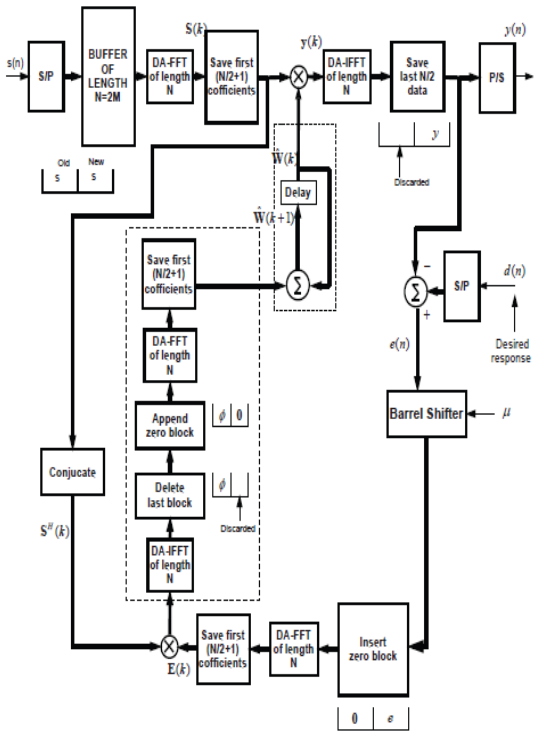

Figure 2. Proposed DA based FBLMS after optimization.

5. CONCLUSION

In this paper, we have proposed a new hardware-efficient FBLMS adaptive filters and its implementation details were presented. The concept of DA involves for implementation of FFT block without any hardware multiplier using LUT and adders. Due to reduced hardware complexity the proposed DA based FBLMS adaptive filter is best suitable for implementation of higher order filters in FPGA efficiently with minimum area requirement, low power dissipation.

6. REFERENCES

[1] Asit Kumar Subudhi, Biswajit Mishra , Mihir Narayan Mohanty "VLSI Design and Implementation for Adaptive Filter using LMS Algorithm" International Journal of Computer & Communication Technology ,Volume-2, Issue-VI, 2011,p84-p87.

[2] Sudhanshu Baghel , Rafiahamed Shaik "FPGA Implementation of Fast Block LMS Adaptive Filter Using Distributed Arithmetic for High Throughput" 2011 IEEE,p443-p447.

[3] Sudhanshu Baghel and Rafiahamed Shaik "Low Power and Less Complex Implementation of Fast Block LMS Adaptive Filter Using Distributed Arithmetic"2011 IEEE,p214-p219.

[4] Amrita , Rajesh Mehra "Embedded Design of an Efficient Noise Canceller for Digital Receivers" International Journal of Engineering Science and Technology (IJEST) Vol. 3 No. 2 Feb 2011,p1252-p1057.

[5] PATEL Shobhit, PANDYA Killol, BHALANI Jaymin, KOSTA Yogesh"Adaptive noise cancellation using least-mean-square algorithm" International Journal on Science and Technology ,Volume1,Issue1,2010, p019-p028.

[6] Tian Lan , Jinlin Zhang "FPGA Implementation of an Adaptive Noise Canceller"2008 IEEE, p553-p558.

[7] John G. Prokis & Dimitris G. Manolakis "Digital signal processing " Prentice Hall of India.