ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.8

Area Efficient LUT Optimization for DA-Based BLMS Adaptive Filter

Ravi L.S[1], Vishalakshi[2]
Assistant Professor[1], PG Student[2]
Department of ECE
RIT, Hassan, Karnataka, India

Abstract:
The aim of this paper is to analyze the contents of lookup tables of DA-based BLMS ADF and based on that intra-iteration lookup table sharing is proposed to reduce its hardware resources, energy consumption, and iteration period. The proposed lookup table optimization scheme offers a saving of lookup table content for block size 4 and higher saving for larger block sizes over the conventional design approach. Here also presents the design of a register-based lookup table matrix for maximal sharing of lookup table contents and full-parallel lookup table-update operation. Based on the proposed design approach, a distributed arithmetic-based architecture for the block least mean square adaptive filter is derived, which is scalable for larger block sizes as well as higher filter lengths. The hardware complexity of the proposed structure increases less than proportionately with input block size and filter length.

Key words: Adaptive filter (ADF), Block least mean square (BLMS), Look-up table (LUT), Distributed arithmetic (DA), Finite impulse response (FIR).

1. INTRODUCTION

Adaptive filters (ADFs) are widely used in various signal-processing applications. Finite-impulse response (FIR) ADF based on the least mean square (LMS) algorithm is the most popular one due to its inherent simplicity and satisfactory convergence performance. However, the delay in availability of the feedback error for updating the weights according to the LMS algorithm does not favor its pipeline implementation when sampling rate is high. The block LMS (BLMS) ADF is a derivative of the LMS ADF for fast and computationally efficient implementation of ADFs.

2. EXISTING SYSTEM

In the existing multiplier-based structures it uses either transpose form configuration or direct form configuration. But, in the multiplier-less structures transpose form configuration is used, in contrast the direct-form configuration is used in distributed arithmetic-based structure. But, for Recursive-FIR filter do not find any particular block-based design. Then, the block structure obtained is not efficient for variable filter coefficients and large filter lengths, for example SDR channelizes.

3. PROPOSED SYSTEM

In the proposed system analyze the contents of distributed arithmetic-based BLMS adaptive filter to find the repetitive look up table words which may be shared to minimize hardware components, the number of look up table entries, energy consumption and iteration period can be analyzed by using intra-iteration look up table. Some of the contributions to this project are as following.

- To minimize the number of look up table words the intra-iteration look up table sharing technique is used.
- To facilitate inter-iteration look up table reuse, the full-parallel look up table update & look up table access structure is used to complete the look up table-update in a single cycle. This causes a significant saving in the iteration period over the conventional design approach.

The proposed model for distributed arithmetic-based block least mean square ADF for block size L=4 & filter length N=6 is as shown in Figure.1.

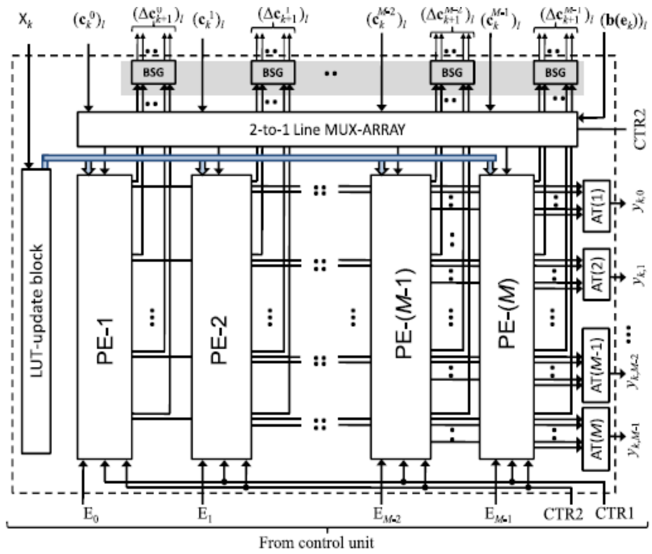

Figure.1. Proposed structure for block FIR filter

The proposed distributed arithmetic block least mean square structure consists of one WBSG, one distributed arithmetic-module and one EBSG. In the weight-update cum bit-slice generator it generates the required bit-vectors and updates the filter weights in harmony with the distributed arithmetic-formulation. Error bit slice generator calculates the error block & generates its bit-vectors. The distributed arithmetic-structure updates the look up table's contents and perfectly uses the bit-vectors produced by error bit slice generator and weight-update cum bit-slice generators to compute the weight-increment and filter output terms.

4. IMPLEMENTATION TECHNIQUES

4.1 LUT Optimization Strategy for the Proposed System

The look up table's duplication corresponding to consecutive iterations of the distributed arithmetic-based least mean square adaptive filter, and based on that the fifty percent of the subsidiary look up table contents are updated.

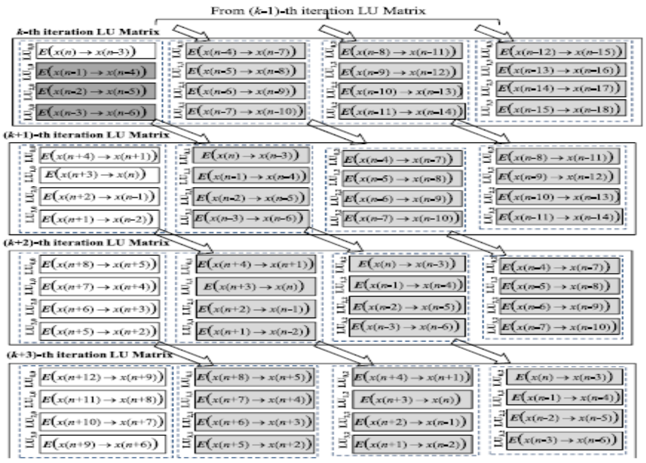

Figure.2. LUT Content of the LU Matrix of block size L=4

For filter length N = 16, 256 look up table words are needed to implement the look up matrix for block size L =4. The contents of look up matrix of block least mean square filter for block size L = 4 are as shown in Figure.2.3. The look up table content is exhibited by the enumeration function *E(.)*, which determines the sum of 16 certain sequence of an input vector.

4.2 Intra-Iteration LUT Sharing

The look up table content does not change during an iteration when its contents depends on the argument $(s^{ij}_{k,p})$ of the look up table enumeration function E. To find the duplicate values in the look up tables of one column of the look up analyzation of the arguments $(s^{ij}_{k,p})$ are made in corresponds to one column of the look up matrix.

4.3 Inter-iteration for Look Up Table Reuse

As shown in Figure.2, the contents of look up table of the first $(M-1)$ columns of the look up's at any given iteration can be used in the last $(M-1)$ columns of the look ups of the next given iteration period, it does not require any update. Only the first column of the look ups requires to be updated during any given iteration period.

Thus, out of M-columns of the look up matrix the contents of only one column of look ups required to be updated during any given iteration period, then the rest of the $(M\text{-}1)$ columns of the look ups of the prior iteration can be reused. The design of look up table-update unit for the proposed structure is expected to be somewhat simpler until the look up table-update unit needs to update only one column of look ups.

5. RESULT

The output waveform of Top module is obtained as shown in figure 3. The output waveform of PE and the RTL schematic of the PE are shown in figure 4 and 5 respectively. The synthesis results are shown in table 1. After encounter the power and area are reduced as shown in table 1.

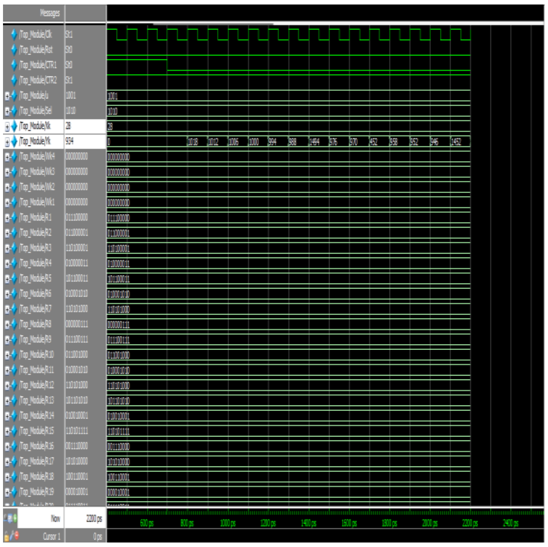

Figure.3. Output waveform Top Module

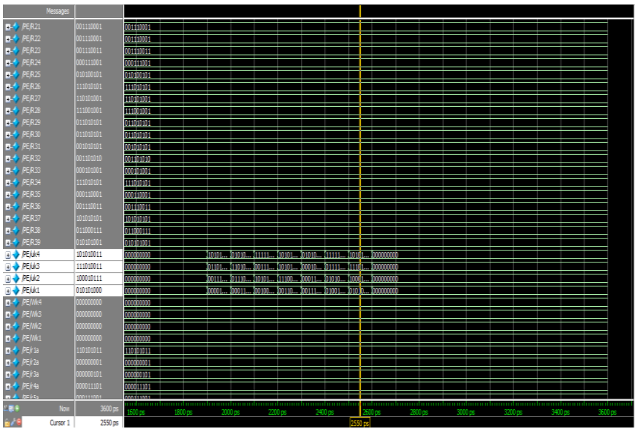

Figure.4. Output waveform of PE Unit

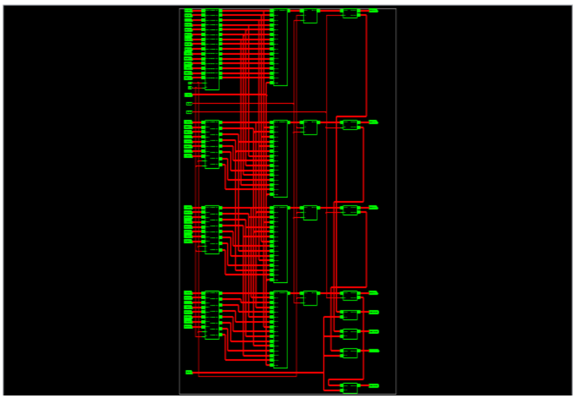

Figure.5. RTL Schematic of PE Unit

Table.1. Area using carry save adder

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	294	3,840	7%	
Number of 4 input LUTs	729	3,840	18%	
Logic Distribution				
Number of occupied Slices	400	1,920	20%	
Number of Slices containing only related logic	400	400	100%	
Number of Slices containing unrelated logic	0	400	0%	
Total Number of 4 input LUTs	742	3,840	19%	
Number used as logic	729			
Number used as a route-thru	13			
Number of bonded IOBs	94	97	96%	
IOB Flip Flops	13			
Number of GCLKs	1	8	12%	
Total equivalent gate count for design	9,005			
Additional JTAG gate count for IOBs	4,512			

6. CONCLUSION

A Distributed Arithmetic-based design of block least mean square adaptive filters detailed LUT content analysis shows that it is possible in reduction of the look up table update complexity and look up table size. Here, an intra-iteration LUT sharing scheme is proposed for block sizes 4, and based on that look up table content analysis is used to optimize the look up table words, compared to the previously existing Block least mean square adaptive filter structure. To facilitate Look up table sharing and a full-parallel look up table-update, a parallel-access Register-based look up table matrix is designed. It updates the look up table's contents in a single cycle.

6.1 FUTURE SCOPE

- This Project can be extended to synthesize for ASIC Application and can be designed to be more efficient for Higher order FIR Filters.
- This design can be enhanced to use for any real time applications like Digital Radio, equalizers for audio systems etc, and
- The number of slices should shrinked for still more optimizing the design.

7. REFERENCES

[1].P.K.Meher and S.Y.Park, "Hihg-throughput pipelined realization of adaptive FIR filter based on distributed arithmetic," in proc. IEEE/IFIP 19th Int. Conf. VLSI Syst-Chip, Oct. 2011,pp. 428-433.

[2].S. Haykin and B. Widrow, Least-Mean-Square Adaptive Filters. Hoboken, NJ, USA: Wiley, 2003.

[3].B.K.Mohanty and P.KMeher. "Delayed block LMS algorithm and concurrent architecture for high-Speed implementation of adaptive FIR filters," in proc. IEEE Region 10 Conf. TENCON, Hyderabad, India, Nov.2008, pp. 1-5.

[4].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. Signal Process., vol. 61, no. 4, pp. 921-932. Feb. 2013.