ASIC Implementation of Shared LUT Based Distributed Arithmetic in FIR Filter

Conference Paper · October 2018

CITATIONS
10

READS
406

2 authors:

Grande NagaJyothi
Madanapalle Institute of Technology & Science
13 PUBLICATIONS **154** CITATIONS
SEE PROFILE

Sridevi Sriadibhatla
VIT University
40 PUBLICATIONS **308** CITATIONS
SEE PROFILE

ASIC Implementation of Shared LUT Based Distributed Arithmetic in FIR Filter

$NagaJyothi.Grande^1$, $Sriadibhatla.Sridevi^2$
School of Micro and Nano-Electronics Engineering
VIT University
Vellore, Tamil Nadu, India
E-mail: grande.nagajyothi@vit.ac.in and sridevi@vit.ac.in

***Abstract*—This paper presents a brief on the implementation of reconfigurable shared LUT (look-up-table) based Distributed Arithmetic (DA) for the higher order finite-impulse response (FIR) filters whose filter coefficients can be changed during run time. In this architecture all the multipliers and adders are replaced by the register banks, multiplexers and the shifters. The throughput rate of the design is increased by having shared LUTs instead of ROM in the DA FIR filter architecture. By implementing this concept in ASIC, the area, area delay product (ADP), minimum cycle period (MCP) and energy per sample are reduced when compare with the conventional DA architecture. The architecture supports 95 MHz sampling frequency.**

***Index Terms*—Distributed Arithmetic, LUT, FIR Filter, Reconfigurable FIR filter**

I. Introduction

Finite Impulse Response (FIR) digital filters plays a paramount role in all digital signal processing application like software defined radio(SDR) [1], channel equalizer, image processing, speech processing. The emerging of various applications causes the filter to design more efficient and effectively. However multiple constant multiplication (MCM) based architectures of FIR filters cannot perform the dynamically change of filter coefficients [2]. These architectures engross more area and power when the filter order increases. Reconfigurability plays a crucial role in the VLSI architectures to perform system compatibility and re usability. Reconfiguration can be done at system level, functional level and logical level. In system level reconfiguration can be done in processor. In functional level, ALUs can be reconfigured. In logical level,LUTs can be reconfigured. In this paper we are concentrated on the logical level reconfiguration.

Several multipliers-less techniques were developed depending upon the performance of filter coefficients in the multiplication operation. Distributed Arithmetic (DA) is one of the principal architecture which has computational efficiency and high throughput processing capability. DA is a bit serial operation used to perform inner product generation, vector to vector multiplication of convolution in a single step, which is an imperative operation in digital filters [3]. The advantage of DA is its ability of mechanization. DA was first introduced by Croisier [4] and Zohar [5] given breif explanation on

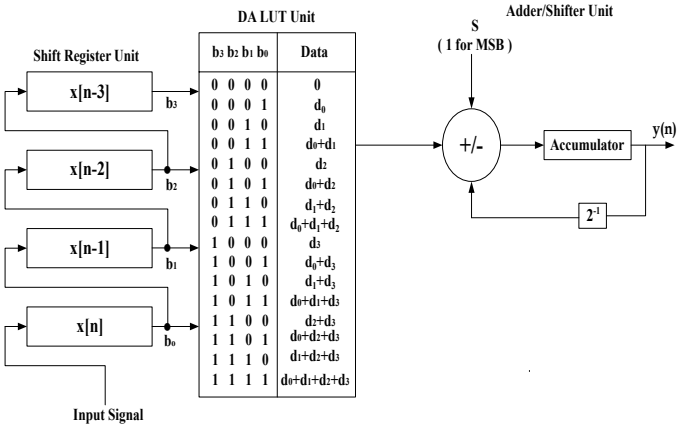

Fig. 1. Conventional DA Architecture

DA . H.Schroder [6] and C.S.Burrus [7] have given ideas to develop the speed in the algorithm. White was proposed the mathematical formulation or calculation of the vector dot product [3]. In DA based architecture, the ROM is used to store the all possible combination values of the filter coefficient; the ROM size increases with the order of filter, to overcome this so many effects have been taken and decrease the size of DA. The basic conventional DA architecture is as shown in figure 1. P.K.Meher developed systolic decomposition of DA architecture for higher order FIR filter [8]. Chen and Chiueh proposed the rewritable RAM based shared LUT DA based FIR filter [9]. P.K.Meher also proposed ASIC implementation of reconfigurable DA [10]. Anderson proposed reconfigurable mixed signal implementation of DA in FIR filters [11] .In paper [11] architecture, filter coefficients are stored in analog form.

The organization of the paper is as follows. Section II gives the formulation and overview of DA algorithm. Section III describes Shared LUT based DA FIR filter architectures. Section IV presents ASIC implementation of reconfigurable shared LUT based DA.Comparative analysis and simulation result of the FIR filter architecture is also explained. Finally, section V concludes the paper.

978-1-5386-1716-8/17$31.00

II. FORMULATION OF DA ALGORITHM

Let us consider the inner product of two vectors d and x given by

$$y = \sum_{k=1}^{K} d_k x_k \tag{1}$$

Where d_k is fixed coefficient, x_k is the input signal and K is the number of input words. x_k is a 2's complement binary number scaled such that $|x_k| < 1$, then x_k can be expressed as:

$$x_k = -b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n} \tag{2}$$

where $x_k = b_{k0}, b_{k1}, ...b_{k(N-1)}$

By substituting equation (2) in (1) and interchanging the summation order finally we get the equation as follows:

$$y = \sum_{k=1}^{K} d_k[-b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n}] \tag{3}$$

Equation (3) is the conventional form of inner product expression. By interchanging the summation order, finally we get equation as shown below

$$y = \sum_{n=1}^{N-1} [\sum_{k=1}^{k} d_k b_{kn}] 2^{-n} + \sum_{k=1}^{K} d_k(-b_{k0})] \tag{4}$$

Equation (4) gives computation of distributed arithmetic and the inner product is computed by

$$y = \sum_{l=1}^{L-1} [2^{-l} C_l - C_0] \tag{5}$$

where

$$C_l = \sum_{k=0}^{K-1} [d(k) S(k)_l] \tag{6}$$

$S(k)_l$ has K point bit sequence .For straightforward DA implementation, let us consider K composite number K=MR where M and R are the positive integers, one can map the index k into (r+mR) for r=0,1,2,.....R-1 and m=0,1,2,3.....M-1 then finally we get expression as

$$S_{l,m} = \sum_{r=0}^{R-1} [d(r + mR) S(r + mR)_l] \tag{7}$$

For any impulse response d(k),2^R possible values of $S_{l,m}$ corresponds to 2^R permutations can be performed. Finally the output equation can be written as :

$$y = \sum_{l=1}^{L} 2^{-l} [\sum_{m=0}^{M-1} S_{l,m}] \tag{8}$$

III. PROPOSED OF SHARED LUT DA BASED FIR FILTER FOR ASIC IMPLEMENTATION

The proposed shared LUT DA architecture consists of SIPO shift register unit, reconfigurable partial product generator unit, and pipeline adder tree and pipeline shift adder tree blocks. The block diagram for ASIC implementation of Shared LUT DA based FIR filter is as shown in figure [2].

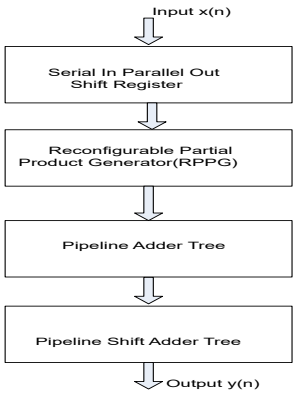

Fig. 2. Shared LUT DA based FIR Filter

A. Serial In Parallel Out Shift Register

The serial input data for every sampling rate is given to the serial in parallel out (SIPO) shift register and the output from the SIPO of size k is given to the reconfigurable partial product generator (RPPG) block.

B. Reconfigurable Partial Product Generation

The outputs from SIPO are taken as the selection lines to the $2^R : 1$ multiplexer where R is the length for m vectors. The RPPG block consists of register bank and the multiplexer. The register bank has 2R-1 registers and multiplexer has L number of $2^R : 1$ muxs. The RPPG generates L partial products to have L bit slice in parallel using the shared LUT register banks. The register bank stores d(2m),d(2m+1),d(2m)+d(2m+1) values. These are given as inputs to the multiplexer and the selection lines are taken from the output of the SIPO shift register. (w+1) bit partial outputs are produced by the partial product generation block.

The outputs from the partial product block are given to the pipeline adder tree. The architecture for the RPPG is as shown in the figure 3. By this technique we can reduce the area occupied by the LUT. By this shared LUT register bank technique the through put is increased and it is more prefer than the ROM based memory when the filter order is increased.

C. Pipeline Adder Tree (PAT) and Pipeline Shift Adder Tree

The output from the RPPG is given to the pipeline adder tree, from PAT the output is given to the pipeline shift adder tree as an input and from pipeline shift adder tree we finally get the output.

D. Pipeline Adder Tree (PAT) and Pipeline Shift Adder Tree

The output from the RPPG is given to the pipeline adder tree, from PAT the output is given to the pipeline shift adder tree as an input and from pipeline shift adder tree we finally get the output.

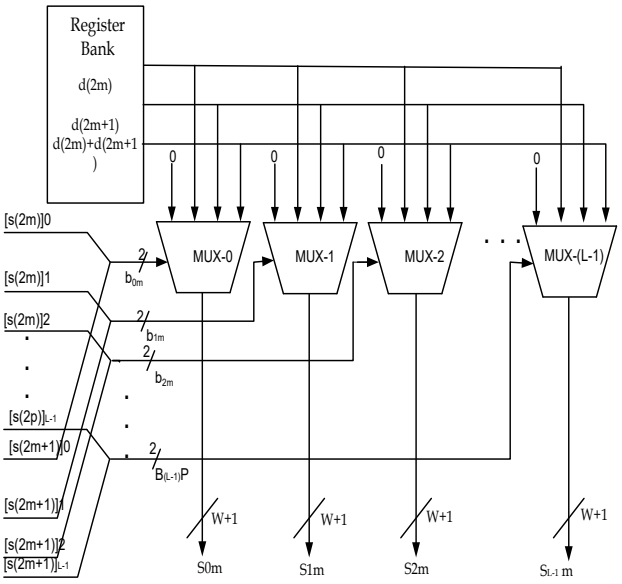

Fig. 3. Reconfigurable partial product for R=2

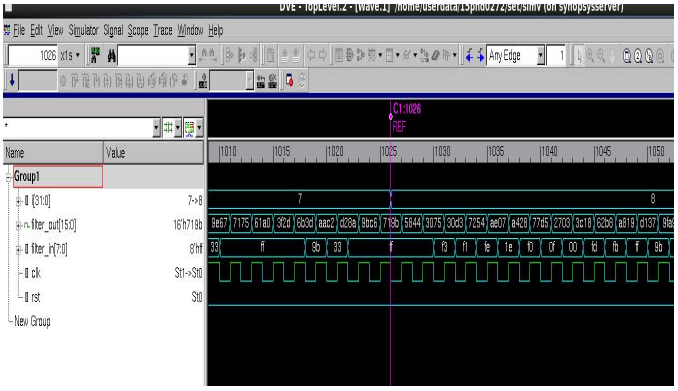

Fig. 4. Simulation result for 16-tap DA based FIR Filter

IV. IMPLEMENTATION RESULTS AND DISCUSSION

The time complexity and hardware implementation of the proposed and existing DA based FIR filter architectures are given in table [1].The DA based systolic architecture [8], DA based pipelined architecture [11] are compared with the proposed architecture. The proposed architecture produces more number of outputs per cycle when compare with the other architectures. The proposed architecture is implemented in ASIC, written in Verilog HDL language and synthesized by CMOS 90nm library in Synopsys Design compiler. The area occupied by shared LUT DA architecture is more when compare with the pipeline architecture [12] and proposed architecture occupy less area when compare with the architecture [8] but area delay product(ADP),power delay product and minimum cycle period (MCP),maximum sampling period(MSP) is low when compared with all other DA architectures. The simulation result for 16 tap Shared LUT reconfigurable DA FIR filter is as shown in the figure [4]. Figure [5] shows the layout diagram for the proposed shared LUT DA based FIR filter.

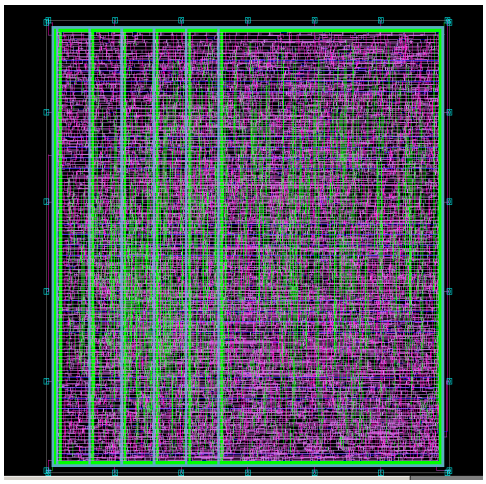

Fig. 5. Layout chip diagram for the proposed Shared LUT DA FIR filter with 90nm technology

TABLE I
PERFORMANCE COMPARISON OF ASIC SYNTHESIS RESULT FOR CMOS 90NM TECHNOLOGY FOR AN 16-TAP FIR FILTER

Design	Area	ADP	PDP	MSP	MSF
Structure[8]	71195	127439	60.76	1.79	558
Structure[12]	18257	94936	32.38	5.20	192
Proposed structure	24163	32757	13.03	1.48	630

Units : Area: $Sq.\mu m$ ADP: $Sq.\mu m \times ns$, PDP: $mW \times ns$, MSP: ns, MSF: MHz

V. SUMMARY AND CONCLUSION

We have suggested shared LUT reconfigurable DA based FIR filter for high throughput, low area and low power. The throughput for the design is increased by using of shared LUT architecture for storing the coefficients. By using this architecture the hardware cost of the design is also reduced. The proposed design has less power consumption and area when compare with the systolic DA architecture.

REFERENCES

[1] Hentschel, M. Henker, and G. Fettweis,"The digital front - end of software radio terminals," IEEE Personal Commun. Mag., vol. 6, no. 4, pp. 40-46, Aug. 1999.

[2] A.K. Oudjida, A. Liacha ,"Multiple Constant Multiplication algorithm for High speed and low power design," IEEE Trans. on circuits and System (TCAS II) ,vol.63, no.2,pp.176-180,February 2016

[3] S. A. White, "Applications of distributed arithmetic to digital signal Processing : A tutorial review," IEEE ASSP Mag., vol. 6, no. 3, pp. 4-19, Jul. 1989.

[4] A Croisier, D. J. Esteban, M. E. Levilion and V. Rizo, "Digital Filter for PCM Encoded Signals," U.S. Patent 3,777,130, December, 1973.

[5] S. Zohar,"New Hardware Realization of Non recursive Digital Filters", IEEE Trans. On Computers, Vol. C-22, pp. 328-338, April 1973.

[6] H. Schroder, "High Word-Rate Digital Filters with Programmable Table Look-Up," IEEE Trans. on Circuits and Systems, Vol. CAS-24, No. 5, pp. 277-279, May, 1977.

[7] C. S. Burrus, "Digital Filter Realization by Distributed Arithmetic," International Symposium on Circuits and Systems, Munich, April 1976.

[8] P. K. Meher, "Hardware-efficient Systolization of DA- based calculation of finite digital convolution," IEEE Trans. Circuits Syst. II, vol. 53, no. 8, pp. 707-711, Aug. 2006.

[9] K.-H. Chen and T.-D. Chiueh, "A low-power digit- based reconfigurable FIR filter," IEEE Trans. Circuits Syst. II, vol. 53, no. 8, pp. 617-621, Aug.2006.

[10] Sang Yoon Park, and Pramod Kumar Meher, "Efficient FPGA and ASIC Realizations of DA-Based Reconfigurable FIR Digital Filter," IEEE Trans. on circuits and systems-ii: express briefs, 2014.
[11] E. Ozalevli, 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. Circuits Syst. I, vol. 55, no. 2, pp. 510-521, Mar. 2008.
[12] P. K. Meher and S. Y. Park, "High-throughput pipelined realization of adaptive FIR filter based on distributed arithmetic," in Proc. 2011 IEEE/IFIP 19th Int. Conf. VLSI, System-on-Chip, (VLSI-SOC11), Oct. 2011, pp. 428 - 433.