ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 7, Issue 11, November 2018

FPGA Implementation of Reconfigurable FIR Filter using Carry Bypass Adder

Shaik Rizwan[1], Shaik Rasool[2]

Assistant Professor, Dept. of ECE, PBR Visvodaya Institute of Technology and Science, Andhra Pradesh, India[1,2]

Abstract: Software-Defined Radio (SDR) is a radio communication system where components that have been traditionally implemented in hardware (e.g. mixers, filters, amplifiers, modulators/demodulators, detectors, etc.) are instead implemented by means of software on a personal computer or embedded system. Reconfigurable Finite Impulse Response (RFIR) filter plays an important role in SDR systems, whose filter co-efficient change dynamically during runtime. In this paper, Low Cost Carry Bypass adder Reconfigurable Finite Impulse Response (LC-CBA-RFIR) is introduced to perform the RFIR filter operations. DRAM-based Reconfigurable Partial Product Generators (DRPPG) consists of MUX and dual port distributed RAM, which has co-efficient to perform a FIR filter operation. With the help of Verilog code, the RFIR filter architecture was verified in Modelsim software. The same Verilog code was used to analyse the FPGA performances such as LUT, flip flop, slice and frequency. After implementing FPGA, all the performance improved in LC-CBA-RFIR method compared to the conventional methods.

Keywords: Reconfigurable FIR, Carry bypass adder, DRAM-based Re-configurable partial product generator

I. INTRODUCTION

Finite Impulse Response filter plays an important role in several signal processing applications in communication schemes, which performs interference cancellation, channel equalization, spectral shaping, matched filtering and more. Nowadays, various implementation and architecture methods have been presented to improve the performance of filters in terms of system complexity and speed. The FIR filter used in the majority of the Digital Signal Processing applications is based on electronic systems. The FIR filter coefficients change rapidly during execution time, in several application scenarios such as a digital up-converter, digital down converter, multi-channel filter and software-defined radio systems. Compared to the conventional non RFIR filter designs with reconfiguration / without reconfiguration, RFIR filters were consuming less resource and power. The discrete FIR filtering detects extensive utility in low-power and high performance Embedded Computing Systems that range from wireless transmitters/ receivers to image and video processing units. Present day research work mainly focused on the design of fully Integrated Circuits that used for wireless applications and employee in the most advanced fabrication methodologies. The main problems in portable telecommunication equipment are long battery life as well as weight, reduced cost and size, which needs low-power and small area integrated devices. The adaptive filter significantly used in the DSP applications. The tap delay line FIR filters whose weights updated by the "Widrow-Haff Least Mean Square" algorithm used as an adaptive filter not only because of its simplicity and but also due to its satisfactory convergence performance.

So, the adaptive filter significantly employed in DSP applications. The shared LUT design for DA based Reconfigurable FIR Digital Filter minimized the hardware cost by decomposing the RAM. The power efficient FIR filter implementation for DSP applications based on FPGA with the support of Xilinx 14.5. Several forms of the structures were analyzed and observed that the pipeline FIR filter structure take a number of registers and indirectly it consumes more resources and power. So it is fit only for high speed DSP application. In multi-media applications and mobile communications, RFIR filters are required because of their main advantage like low-cost, less area, low power and high speed operation. The pipelined modified booth multiplier is used for RFIR filter architecture. This architecture has changed the order of the filter to reach significant savings in power consumption than existing architectures but this strategy is not possible for the low-power applications. Low power 8-bit based RFIR filter with minimum power consumption system improved efficiency but it used only for 8-bit data. Digital RFIR filter method consists of low power serial multiplier and serial adder, shift/adder, shift/multiplier combinational booth multiplier, folding transformation in linear phase architecture. The normal adder has a long critical path and consumes more power. Also, hardware utilization and the execution time is more in previous works. In this work, the CBA is used instead of normal adder. Due to the CBA adder, the RFIR architecture achieves better performance in terms of less area, power, and delay. In FPGA implementation, the number of LUTs, slice and flip-flop decreased in CBA-RFIR for different kinds of Virtex devices such as Virtex - 4 and Virtex - 5. This paper is composed as follows. In section 2, described some previous related work. In Section 3, shows Proposed LC-CBA-RFIR design architecture. In Section 4, mentioned experimental setup and results. The conclusion is made in Section 5.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 7, Issue 11, November 2018

II. LITERATURE SURVEY

S.Y. Park, and P.K. Meher [1] illustrated a novel pipelined architecture for a lower power, high throughput and low area adaptive filter based on DA. The throughput rate of the FIR design was maximized by the parallel LUT update and concurrent implementation of the filtering and weight operation. Reduction of the power consumption was improved by using a fast Bit-clock used for Carry-Save Accumulation (CSA) but it has a much slower clock speed for all the other operations. S. Ramanathan, G. Anand, P. Reddy, and S.A. Sridevi [2] have presented a low-power adaptive FIR filter based on DA with high-throughput, lowpower and area. The Least Mean Square (LMS) algorithm is employed to update the weight and reduce the Mean Square Error (MSE) between the current filter outcome and the desired response. The pipelined DA table decreases switching activity and decreased power. The main limitation of this paper is that it is significantly focused on power consumption. N. Sriram, and J. Selvakumar [3] used Pipelined Modified Booth Multiplier (PMBM) method used for implemented low power RFIR filter architecture. But limitation of this method is that delay value is high because of decreased system speed and throughput. K.M. Basant, P.K.Meher, S.K. Singhal, M.N.S. Swamy [4] introduced the high-performance VLSI architecture for RFIR using DA. Here the author has analyzed the two kinds of structures and conclude that the direct form structure needs less number of registers compared to the transpose form structure. Reconfigurable block-based FIR filter with DA provides the scalability for higher block sizes and larger filter lengths. But the limitation of this method only discussed the structures for a block size of 4. R. Jia, H.G. Yang, C.Y. Lin, R. Chen, X.G. Wang, and Z.H. Guo [5] introduced novel RFIR filter design based on statistics centric reconfigurable (SCR) FIR filter architecture. The experimental results were analyzed by considering performance parameters such as area, speed, and power for the high-order FIR filters and they have concluded that proposed RFIR filters have improvements in their performance over the conventional FIR filter but did not discuss about the dynamically reconfigurable mechanism. For existing work, they have used normal full adder, pipeline adder, and parallel adder etc. That adder based techniques occupied more area, more power, high critical path, and more hardware utilization in FPGA implementation. To overcome these problems, LC-CBA-RFIR method is introduced for evaluating FPGA implementation results.

III. PROPOSED LC – CBA - RFIR METHODOLOGY

The CBA-RFIR technique consists of Serial-in-Parallel-out Shift Register, Distributed Read Access Memory based Reconfigurable Partial Product Generator, Pipeline Adder Tree shifter and pipeline shift add three elements on the FPGA implementation. The CBA-RFIR technique based on RFIR filter structure has been implemented in FPGA with CBA. The FPGA methodology is developed from a dedicated hardware to a heterogeneous system, which is why it's a popular choice in the communication base stations instead of being a prototype platform. The reconfigurable DA-based on FIR filter is used for FPGA implementation by employing the CBA-RFIR technique. The LUT's are developed by using the DRAM with FPGA implementation. The multiple numbers of the partial inner-products Sl, are retrieved from the DRAM simultaneously, so only one LUT value is read from DRAM, per cycle. Furthermore, if "L" is the bit width of the input, the duration of the sample period of the design is L times the operating clock period. That is not suitable for the application requiring high-throughput. DRAM is employed to develop LUT for each bit slice because of its high-resource consumption. Hence, decompose the partial inner-product generator into Q parallel sections and every section has R time-multiplexed operations corresponding to Rbit slices. When L is a Composite number provided by $L = RQ$ (Here R and Q are two positive integers), index l in Eq. (1) is mapped with $r + qp$ for $r = 0, 1, 2, \ldots, -1$ to modify in Eq. (2) as

$$y = \frac{1}{2}\sum_{l=0}^{L-1}\left(\sum_{p=0}^{P-1}(S_{l,p})\right) \quad \text{-- (1)}$$

$$S_{l,p} = \sum_{m=0}^{M-1}(h(m + pM))[S(M + pM)] \quad \text{-- (2)}$$

Here, $l = 0, 1, 2, \ldots, L-1$ and $p = 0, 1, 2, \ldots, P-1$ since the sum of partial product is Sl,p of the M samples.

$$\sum_{q=1}^{Q-1}(2^{-RP})\left[\sum_{r=0}^{R-1}2^{-1}\sum_{P=0}^{P-1}r + q, R, P\right] \quad --(3)$$

In Eq. (3), $q$represents as index and r represents time index. The structure of the CBA-RFIR time multiplexed DA based FIR filter by employing DRAM is shown in the Fig.1.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 7, Issue 11, November 2018

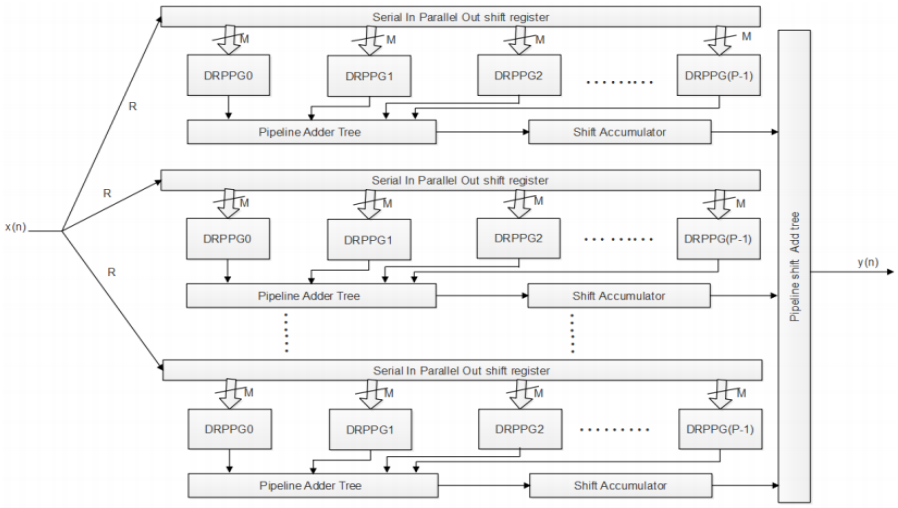

Figure.1 Architecture for Proposed CBA-RFIR time-multiplexed DA based FIR filter by employing DRAM.

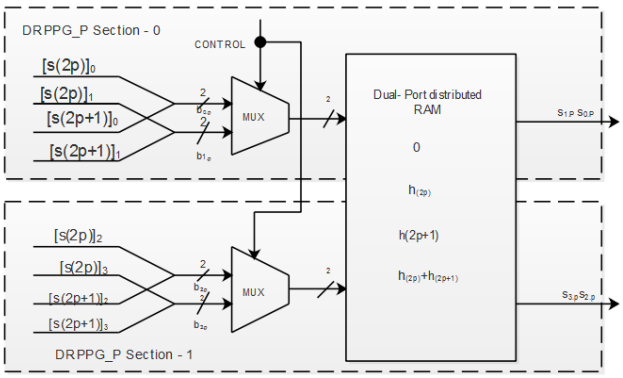

Figure.2 The structure of DRAM base DRPPG

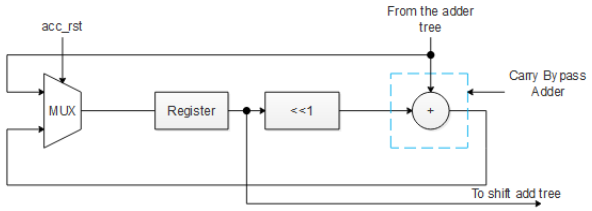

Figure.3 The structure of shift accumulator

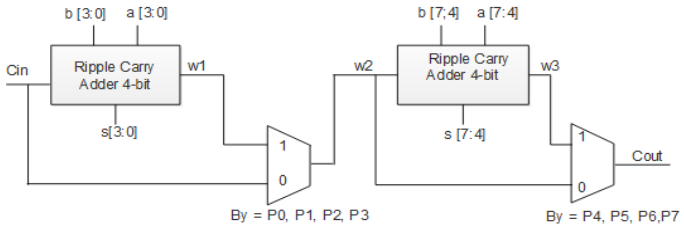

Figure.4 Block diagram of the carry bypass adder

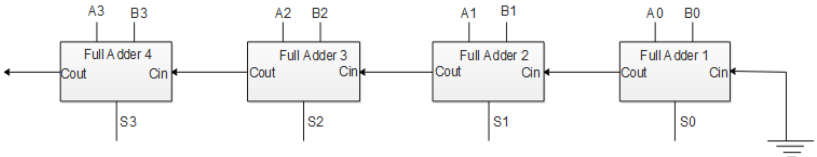

Figure.5 4-bit ripple carry adder

DOI 10.17148/IJARCCE.2018.71109

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 7, Issue 11, November 2018

To develop Eq. (3), the CBA-RFIR structure has Q section and every section consists of P DRAM based Reconfigurable Partial Product Generators (DRPPG) and the PAT to compute the rightmost result followed by Shift Accumulator (SA) which performs over R cycles according to the second summation. However, it employs dual-port DRAM to decrease the total size of the LUTs by half than that of 2-DRPPGs in which two different sections shares a single DRAM. In the Fig.2 shows the structure of DRAM based on DRPPG. In rth cycle is P DRPPG in q^{th} section operate P – partial inner product $Sr+q$, used for p=0,1,2,……,$P-1$to add by using the Pipeline Adder Tree (PAT). The outcomes of the PATs are accumulated through AS on the R cycle shown in the Fig. 3. The accumulated rate is reset at each R cycle by control signal to keep the accumulator register ready to be utilized for calculation of the next filter output. The $fclk$ maximum operating clock period and the CBA-RFIR techniques up ports the input sample range by f_{clk}/R.

In the CBA, Ripple Carry Adder (RCA) is employed to add four-bits at a time and the carry generated will be propagated to next stage with the help of multiplexer utilizing selected input as Bypass logic. Bypass logic is formed from the production values as it is computed in the CLA. Depending on the carrying value and bypass logic, the carry is propagated to the next stage. The CBA is an adder implementation, which improves the delay of an RCA. The 4-bit CBA design requires 4-FA circuits. The input buses would be a 4-bit A and 4-bit B with a carry- in (Cin) signal. The output would be a 4-bit bus X and Carry-out ($Cout$) signal. The first 2-FAs would add the first 2-bits together. The $Cout$ signal from the second-FA ($C1$) would drive the selected signal for three 2:1 multiplexers. Fig. 4 shows the block diagram of the carry bypass adder. The 4-bit RCA is shown in Fig. 5. Multiple FA are cascaded in parallel to add an N-bit. For N-bit parallel adder, there are N-number of FA circuits. An RCA is a logic circuit, in which $Cout$ of each FA is Cin of the succeeding next significant FA that is known as RCA due to each carry bit gets rippled into the next stage. In RCA the sum and carry-out bits of any half adder stage are not valid until Cin of that stage occurs. Propagation delay is the time elapsed between the application of input and occurrence of the corresponding output.

For example, for a NOT gate, when the input is “zero” the output will be “one”. The time taken for the NOT gate’s output to become “zero” after the application of logic “one” to NOT gate’s input is the propagation delay. Similarly the carry propagation delay is the time elapsed between the application of the carry in signal and the occurrence of the $Cout$ signal. Sum out $S0$and $Cout$ count of the FA “one” is valid only after the delay of the 1-bit FA. In the same way, sum out $S3$ of the 4-bit FA is valid only after the joint propagation delays of 1- bit FA to 4-bit FA. The final outcome of the RCA is valid only after the joint propagation delay of the FA circuit inside it.

IV. RESULTS

The LC-CBA-RFIR design timing diagram was verified in Modelsim using Verilog code. RTL schematic was taken from Xilinx ISE tool. FPGA performance was analyzed for different devices of Virtex-4 and Virtex-5 by using Xilinx ISE tool.

(i). FPGA synthesis: This FPGA synthesis is implemented in Xilinx tool for different devices such as Virtex-4 and Virtex-5 From this tool, the performance parameter like LUT, flip-flop, Slices, and Frequency has been calculated.

A). LUT: A LUT, which stands for LUT, in general terms it is basically a table that determines what is the output for any given input(s). In the context of combinational logic, it is the truth table. This truth table effectively defines how combinational logic behaves.

B). Flip-flop: Flip-flops are binary shift registers used to synchronize logic and save logical states between clock cycles within an FPGA circuit. On every clock edge, a flip-flop latches the 1 or 0 (TRUE or FALSE) value on its input and holds that value constant until the next clock edge.

C). Slices: Logic resources are resources on the FPGA that perform logic functions. Logic resources are grouped in slices to create configurable logic blocks. A slice contains a set of LUTs, flip-flops, and multiplexers. A LUT is a collection of logic gates hard-wired on the FPGA.

D). Frequency: Frequency is defined as the rate at which something occurs over a particular period of time or in a given sample. Table 1 is the comparison of the 8-bit input sample to analyze performance parameters such as LUTs, the number of flip-flops, slices, and operating frequency for different FPGA devices such as vertex 5 and vertex 6. This result has been taken for different bits and taps like 8 B & 3T, and 8 B & 7T. From this table, it is concluded that the LUT, flip-flop, slices reduced and operating frequency is increased in LC-CBA-RFIR method than the existing RFID method. Due to the reduction of those parameters, the area has been minimized in filter architecture.

The RTL schematic of FIR filter is shown in Fig.6, which is taken from Xilinx ISE software using Verilog code. This architecture is having a separate code for each block such as a counter, reg_bank, GRPPG, and an accumulator. Input is stored in a registered bank in the form of bitwise. That register bank input is performed DRPPG operation, which contains MUX and DRAM. In DRAM, the coefficient value is stored to perform FIR filter operation. DRPPG output is performed on the accumulate operation then it will give RFIR filter output in “y”.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 7, Issue 11, November 2018

Table-1. Implemented on different Xilinx FPGA devices for various tap of 8 bit FIR filter

8 – bit Input					
Target FPGA	**DESIGN**	**LUT**	**Flip-Flop**	**Slice**	**Frequency (MHz)**
Virtex4 xc4vfx12	**Existing**	142/10944	98/10944	94/5472	108.312
	LC-CBA- RFIR	**112/10944**	**87/10944**	**78/5472**	**110.011**
Virtex5 xc5vlx20T	**Existing**	178/12480	98/12480	76/3120	89.432
	LC-CBA- RFIR	**144/12480**	**87/12480**	**73/3120**	**94.150**

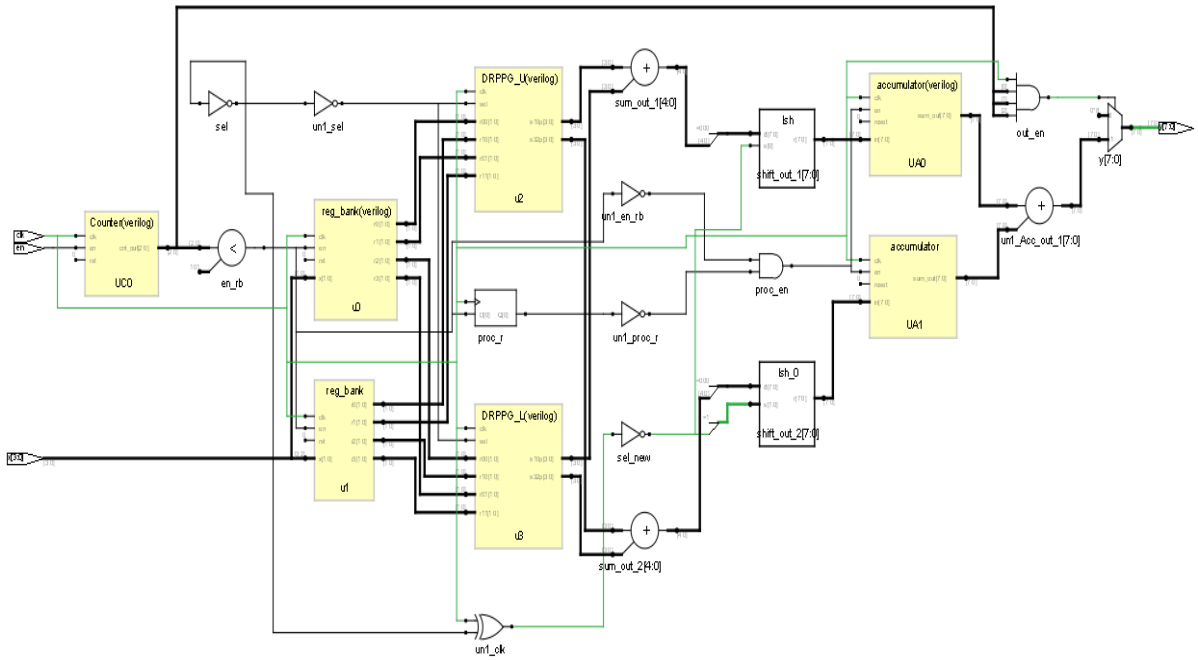

Figure.6 RTL schematic diagram of 8B & 3T

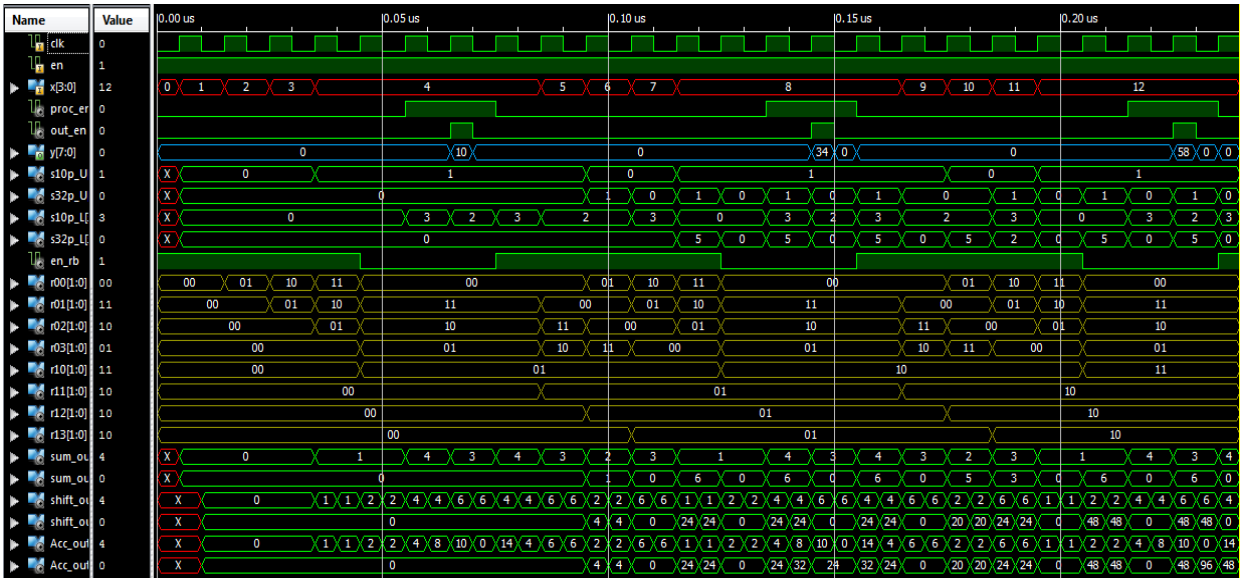

Figure.7 Output wave form of 8B & 3T

DOI 10.17148/IJARCCE.2018.71109

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 7, Issue 11, November 2018

The output waveform of 8B & 3T is shown in Fig.7. The input value is represented as a red color waveform. For example, '4, 3, 2, and 1' is input value, which is stored in register bank in the form of r00, r01, r02, r03, r10, r11, r12, and r13. That register bank is represented as a brown color waveform. DRPPG output is denoted as s10p_U, s32p_U, s10p_L, and s32p_L, which gives the output base on MUX selection line and a DRAM in DRPPG. This input such as '4, 3, 2, and 1' is stored in the registered bank, which performs the filter operation according to Section. 3.3.2. Here, consider co-efficient as '0, 1, 2, and 3' for four different inputs. The output value 10 is stored in "y", which is represented as blue color. When proc_en and out_en are in a high state (1), the output is generated in "y". From this waveform, it is clear that the RFIR architecture is working perfectly.

CONCLUSION

In this paper, LC-CBA-RFIR architecture has been implemented in ModelSim software by writing Verilog code. Area, power and the delay parameters are evaluated for different bits and taps like 8 B & 3T, and 8 B & 7T. Using FPGA implementation, LUT's, slices, flip-flops and the frequency improved in LC-CBA-RFIR architecture. In future, this FIR filter design will be performed by using Carry Increment Adder (CIA) to further reduce the hardware utilization like LUT, slices, and flip flop as well as area, power and delay.

REFERENCES

[1]. S.Y. Park, and P.K. Meher, "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, No.6, pp.346-350, 2013.
[2]. S. Ramanathan, G. Anand, P. Reddy, and S.A. Sridevi, "Low Power Adaptive FIR Filter Based on Distributed Arithmetic", Int. Journal of Engineering Research and Applications, Vol.6, No.5, pp.47-51, 2016.
[3]. N. Sriram and J. Selvakumar, "A Reconfigurable FIR Filter Architecture to Trade Off Filter Performance for Dynamic Power Consumption", Int. J. Adv. Comput. Theor. Eng.(IJACTE), Vol.2, No.1, pp.112-119, 2013.
[4]. K.M. Basant, P.K.Meher, S.K. Singhal, and M.N.S. Swamy, "A high-performance VLSI architecture for reconfigurable FIR using distributed arithmetic", Integration, the VLSI Journal, Vol.54, pp.37-46, 2016.
[5]. R. Jia, H.G. Yang, C.Y. Lin, R. Chen, X.G. Wang, and Z.H. Guo, "A Computationally Efficient Reconfigurable FIR Filter Architecture Based on Coefficient Occurrence Probability", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol.35, No.8, pp.1297-1308, 2016.
[6]. S.J. Lee, J.W. Choi, S.W. Kim, and J. Park, "A reconfigurable FIR filter architecture to trade off filter performance for dynamic power consumption", IEEE transactions on very large scale integration (VLSI) systems, Vol.19, No.12, pp.2221-2228, 2011.