International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 10 ǁ October 2022 ǁ PP. 138-145

Design and Analysis of High Throughput and Energy Efficient FIR Filter using Retiming and Pipelining in Verilog Coding

Shikhar Bajpai1 · Krishna Raj2

ABSTRACT
In this research, recent advances in low power, high throughput, and less area architectures and algorithms are analysed based on the Finite Impulse filter (FIR) design. Filter designs are applied in several applications like medical diagnosing, image/Speech processing and arithmetic computations. In various applications, a FIR is a channel whose impulse response is of limited period, because of it settles to zero in finite time.
A methodology to improve the throughput and energy efficiency of finite impulse response (FIR) filters through the effective application of retiming and two-level pipelining is presented in this paper. Improving throughput and energy efficiency of the filter while minimizing latency and hardware complexity is a challenge to be addressed. Two-level pipelining bifurcate the multiplication and addition operations. Retiming is applied to break addition operation.
The major challenges of FIR filters are simultaneous approximationin both magnitude and phase responses. Likewise, the designer faces lot of issues while constructing the alternative trade-offs. The filter design must provide goodefficiency and simplicity. Hence, this research reviewed the recent filtering application-based articles to extract the exact problem in designing the FIR filter realization. A 16 tap FIR Filter is designed in VHDL Language and performed simulation in XilinkVivado 2019.1
Keywords: *Finite impulse response (FIR), medical diagnosing, image&Speech processing, Retiming, Two-level pipelining*

Date of Submission: 23-09-2022 Date of acceptance: 06-10-2022

I. INTRODUCTION

High throughput and energy efficient FIR filter structures are the essential components of most of the real time digital signal processing (DSP) applications. FIR filters are often used in feed-forward control systems to generate an inverse response filter of the plant. As filter length increases the critical path length of direct form FIR filter increases. One of the ways in Improvement in input data processing rate or throughput can be achieved through the application of pipelining and parallel processing. Pipelining reduces the critical path length and thereby improves throughput at the cost of increased latency.

The transposed FIR filter has inherently shorter critical path compared to its direct form counterpart as it consists of only a multiplier and an adder. With increase in number of taps, the bit-width of the final adder in the accumulation path increases. As a result, the critical path length increases with filter length. The bit-width of the delays in the accumulation path also increases in accordance with the increase in adder bit-width. This increases input capacitance and area over the direct form filter.

Retiming is a technique in which alteration to the critical path of FIR filter is done by changing the placement of the registers in the original FIR filter circuit. The placement of registers has to be made such that the functionality is not altered and critical path is minimized. The critical path minimization reduces overall clock period thereby increasing the clock frequency in retiming. At times, this may lead to increased register counts. The register minimization retiming provides the best compromise of clock period and register count.

An M-level pipelining divides the circuit into M sections reducing the CPD to the worst-case delay among M sections. Though the throughput is increased, the latency is increased by (M-1) clock periods. The area also increased due to the increase in number of pipeline delays used. An L-fold parallel processing will improve the sample rate L-fold without latency at the expense of L-fold increase in area. Low complex, high throughput and energy efficient FIR filter structure with reduced latency is proposed here. The proposed filter is developed through the appropriate application of two level pipelining and retiming. Though the application of two level pipelining increased the latency by one clock period, remarkably good improvement in performances in terms of CPD, power, MUF, ELT PDP and ADP is achieved.

II. HISTORICAL REVIEW

- Lee *et al.* combined the direct and transpose form structures to reduce latency of FIR filter. Reduction in latency at the cost of increased area was achieved by shifting some of the input registers to the accumulation path and. was found to be very suitable for delay and power efficient applications. [4]
- Khoo *et al.*developed an efficient direct form FIR filter by dividing filter into many subsections subjected to a cycle-time constraint. [6]
- Samueli had reduced the filter implementation cost function through the application of timing-driven logic optimizations. [7]
- Thorough review of various hardware efficient FIR filter designs was done by Chandra and Chattopadhyay. [8]
- Jiang and Brayton synthesized the logic design from a complexity perspective. Yagain and Vijaya have applied retiming to multiplier-less multiple constant multiplication (MCM) algorithm. [11]
- Meher has improved the CPD of conventional direct form filters at the cost of increased hardware complexity through the application of cut-set and flexible retiming. [13]
- Thakral *et al.* applied unfolding transformation to a FIR filter using carry increment adders (CIA) and Vedic multipliers. [14]
- Swati and Himanshu developed FIR filter using Wallace multiplier. [15]
- Two versions of efficient hybrid adders and improved booth multipliers are discussed.
- Rai *et al.* used carry increment adders and Vedic multipliers for the implementation of FIR filter and achieved improvement in CPD and energy efficiency at the cost of increased area in comparison with conventional direct form FIR filter. [17]
- Ting *et al.* developed a High throughput and energy efficient adaptive LMS predictor for electronic support measures receivers through fine grain pipelining of multiplier and pipelining of FIR filter at the cost of very large increase in area. [18]
- Meher and Park improved the throughput of FIR filter used for adaptive algorithm through two level pipelining at the cost of slight increase in area and latency. [19]
- Meher explored various pipelining possibilities in DSP circuits. [20]
- Mathias *et al.* improved the CPD of transposed form FIR filters through the pipelining to structural adders at the expense of increased hardware complexity. [21]
- Pramod and Shahana developed a high throughput adaptive filter architecture using modified transposed form FIR filter. [22]
- Pramod and Shahana developed two versions of high throughput FIR filters using retiming and modified CSLA based adders. [25]

III. RETIMING, PIPELINING &PARALLEL PROCESSING

Retiming is the technique of moving the structural location of latches or registers in a digital circuit to improve its performance, area, and/or power characteristics in such a way that preserves its functional behavior at its outputs. Retiming was first described by Charles E. Leiserson and James B. Saxe in 1983.[1]

The technique uses a directed graph where the vertices represent asynchronous combinational blocks and the directed edges represent a series of registers or latches (the number of registers or latches can be zero). Each vertex has a value corresponding to the delay through the combinational circuit it represents. After doing this, one can attempt to optimize the circuit by pushing registers from output to input and vice versa - much like bubble pushing. Two operations can be used - deleting a register from each input of a vertex while adding a register to all outputs, and conversely adding a register to each input of vertex and deleting a register from all outputs. In all cases, if the rules are followed, the circuit will have the same functional behavior as it did before retiming.

Pipelining defines the temporal overlapping of processing. Pipelines are emptiness greater than assembly lines in computing that can be used either for instruction processing or, in a more general method, for executing any complex operations. It can be used efficiently only for a sequence of the same task, much similar to assembly lines.

Each task is subdivided into multiple successive subtasks as shown in the figure. For instance, the execution of register-register instructions can be broken down into instruction fetch, decode, execute, and writeback.

Parallel processing can be described as a class of techniques which enables the system to achieve simultaneous data-processing tasks to increase the computational speed of a computer system.

A parallel processing system can carry out simultaneous data-processing to achieve faster execution time.

A parallel processing system can be achieved by having a multiplicity of functional units that perform identical or different operations simultaneously. The data can be distributed among various multiple functional units.

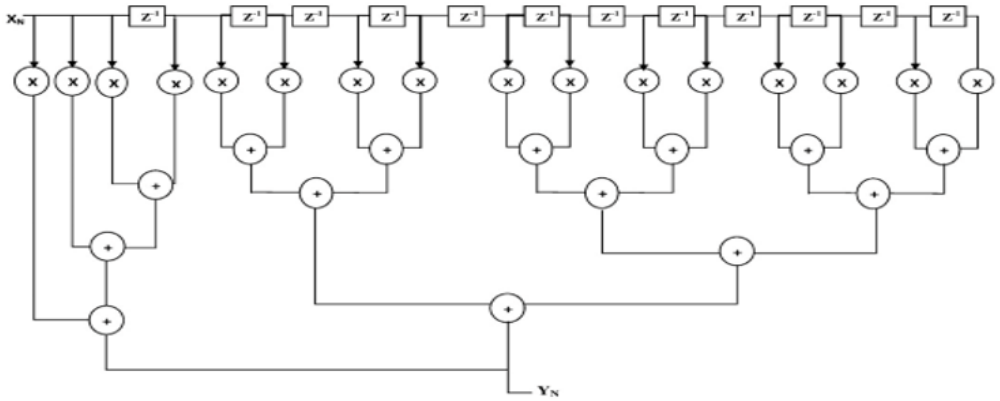
IV. PROBLEMS DEFINITIONS

For timing, area and power analysis, we should keep in mind multiple factors that make it a better signal in terms of these factors.

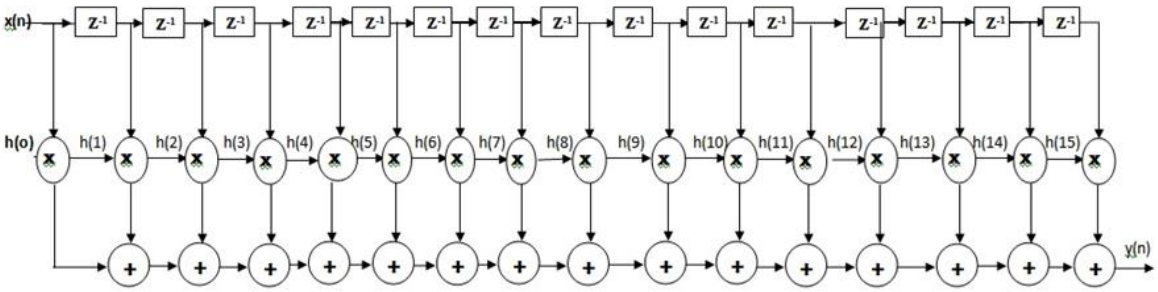
The best fir filter consists of combination of less latency, high throughput better power and less area involved.

There are many different methods for achieving the same. Many researchers have done a remarkable job earlier with different methods like windowing method, the frequency sampling methods and optimal filter design methods. Here we are using pipelining and retiming methods for better enhancement and performance of fir filter design.

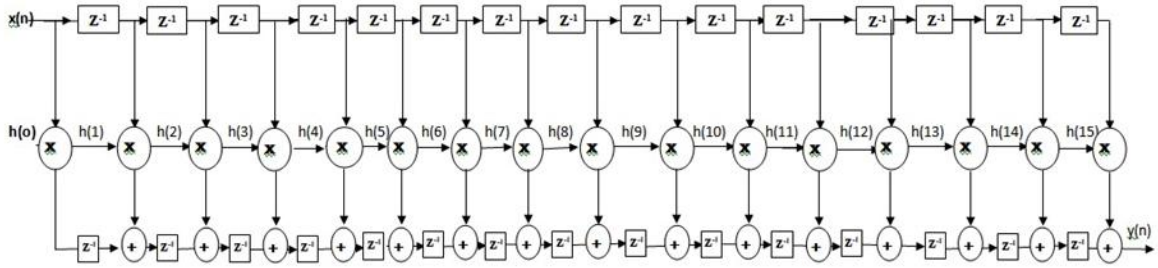
V. BLOCK DIAGRAM



Design A: 16 Tap FIR Filter



Design B: 16 Tap FIR Filter with Pipelining



Design C: 16 Tap FIR Filter with Retiming & Pipelining

VI. MATHEMATICAL EQUATIONS.

The factors which we will be covering in this report are described and compared below.

6.1 Critical path delay (CPD)

Critical path delay can be defined as the delay between a transmitter and a receiver.
Critical Path Delay = Propagation delay+(N-input AND gate+(K-input AND gate-2)) *Delay of the AND-OR gate+ Multiplexer Delay.
After Implementation, Click of Synthesis Report Summary to get the overall delay and delay of each component partwise as well as delay of the particular path.

6.2 Maximum Usable Frequency (MUF)

Maximum Usable Frequency is defined as the inverse of Critical Path Delay. MUFis the highest radio frequency that can be used for transmission between two points
MUF = 1/ CPD.

6.3 Effective LatencyELT

ELT stands for Effective Latency. For combinational circuits, the latency is simply the propagation delay of the circuit and its throughput is just 1/latency.

6.4 POWER

To calculate total power consumed by the Filter in mW, we need to pre define some points and that is:
First go to Report Power Tab under the "Implementation" section.
Under Environment Dialogue Box, Set Ambient Temperature to 65 deg. C, Temp Grade to "Commercial" and Process to "Maximum" to get the "Power"

6.5 Area

Area is used to measure the total area the filter is consuming in order to show the output. It is calculated in sq. micro meter
To check the Area of the Design, after implementation, go to Synthesis Report summary to get the hardware complexity.

6.6JUNCTION TEMPERATURE

Junction Temperature is the highest operating temperature of the actual semiconductor in an electronic device

6.7FINITE IMPLUSE RESPONSE FILTER

To calculate Fir filter design, the combined eqn. for n tap filter is:

$$y[n] = \sum_{k=0}^{N-1} h[k] \cdot x[n-k]$$

PERFORMANCE EVALUATION

On comparing all the three designs, we found remarkable change in power, junction temperature and CPD which increased our throughput and found little change in area which did not affect the throughput much. Hence, we found Design C to be much more effective with high throughput.

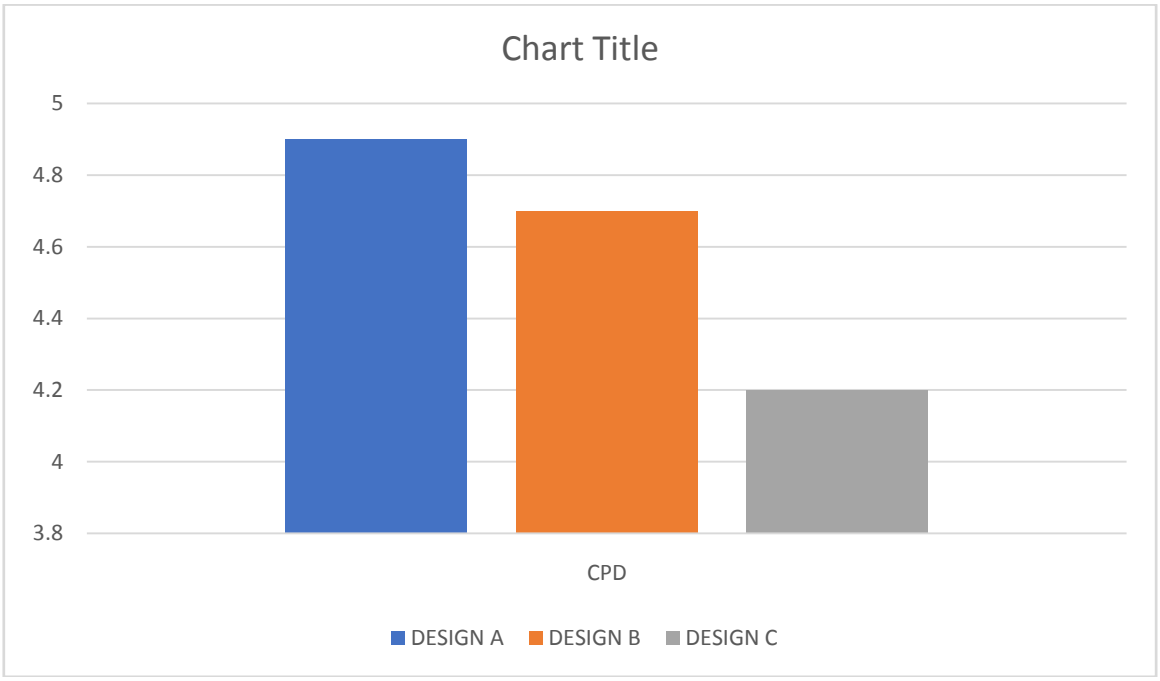


Chart Title
5
4.8
4.6
4.4
4.2
4
3.8
CPD
DESIGN A
DESIGN B
DESIGN C

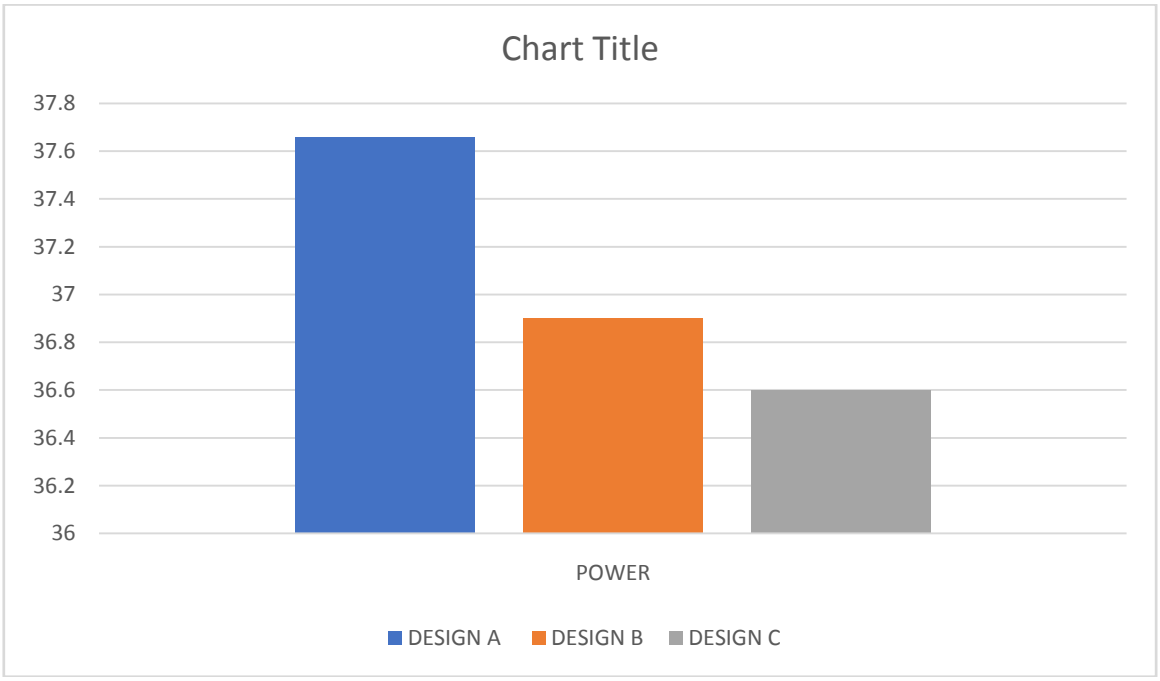


Chart Title
37.8
37.6
37.4
37.2
37
36.8
36.6
36.4
36.2
36
POWER
DESIGN A
DESIGN B
DESIGN C

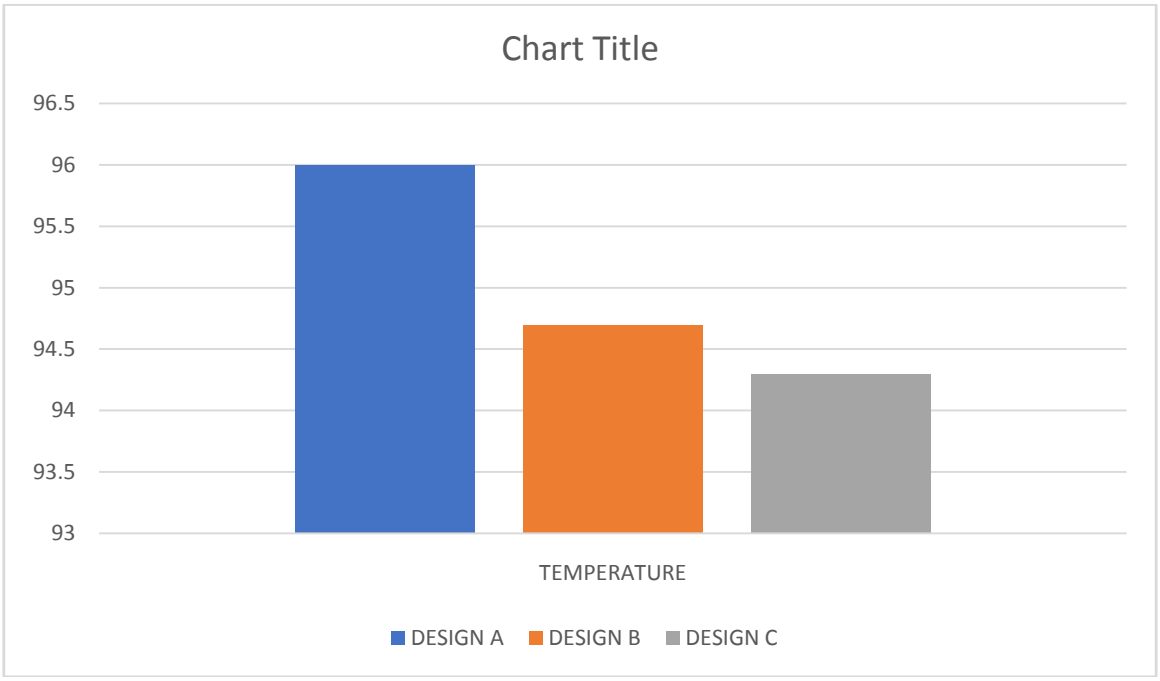


Chart Title
96.5
96
95.5
95
94.5
94
93.5
93
TEMPERATURE
DESIGN A
DESIGN B
DESIGN C

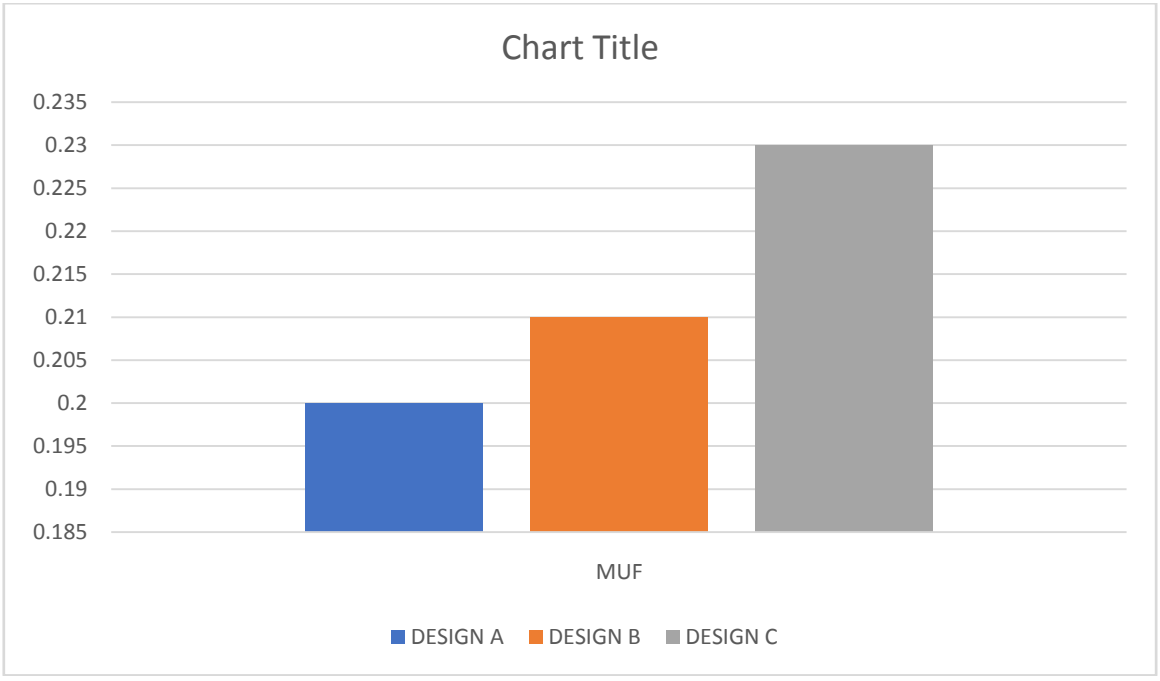
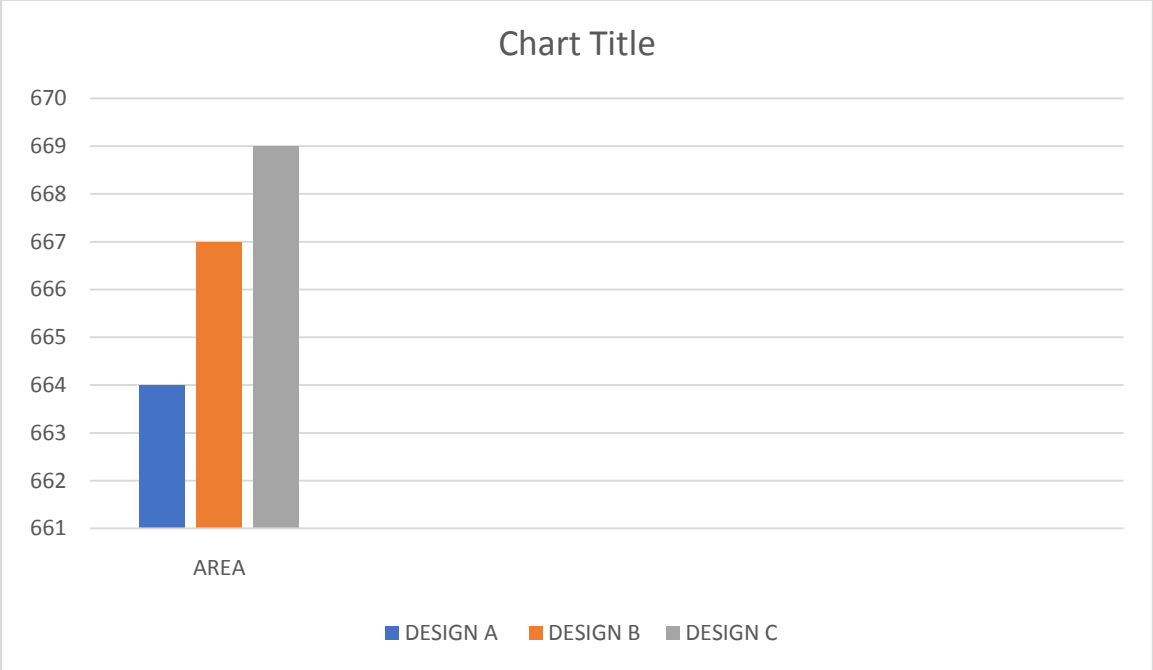


Chart Title
0.235
0.23
0.225
0.22
0.215
0.21
0.205
0.2
0.195
0.19
0.185
MUF
DESIGN A
DESIGN B
DESIGN C



TABULAR FORM:

PARAMETERS	DESIGN A	DESIGN B	DESIGN C
CPD	4.9	4.7	4.2
POWER	37.7	36.9	36.6
MUF	0.2	0.21	0.23
TEMPERATURE	96	94.7	94.3
AREA	664	667	669

VII. CONCLUSION

In design A, we simply made a 16 Tap FIR Filter with no pipelining and Retiming.
We then introduced Pipelining to our design B and obtained the results.
Then we introduced Retiming to the design C and obtained the results. On comparing all the three design results, we come to the conclusion that Power consumed got reduced by 3%, Temperature got reduced by 1.7% and latency got reduced by 14% when we introduced Retiming and Pipelining while the complexity/ Area increased by 0.7% only.
Thus, the throughput is increased by introducing the two methods i.e. Retiming and Pipelining.

REFERENCES

[1]. S.Carlo, and O. Tokunbo,(2015) "FPGA implementation of LMS-based FIR adaptive filter for Real time Digital Signal Processing Applications". Proc. IEEE International Conf. Digital Signal Processing.
[2]. N.S. Johannes and L. Boris,(2017) "Optimal feedforward preview control by FIR filters," IFAC Papers OnLine , 5115–5120.
[3]. K.K. Parhi,(2007) VLSI Digital Signal Processing Systems Design and implementation. Interscience.
[4]. H.R. Lee, C.W.Jen, C.M.Liu, (1996)"New hardware-efficient architecture for programmable FIR filter," IEEE Trans. Circuits Sys., 43, 637–644.
[5]. K. Azadet and C.K. Nicole,(1998) "Low-power equalizer architectures for high-speed modems," IEEE Comm. Mag.
[6]. K. Khoo, Z. Yu and N.W. Alan,(2001) "Design of optimal hybrid form FIR filter," Proc. IEEE sym.
[7]. H. Samueli,(1989) "An improved search algorithm for the design of multiplierless FIR filters with powers-of-two coefficients," IEEE Trans. Cir. Syst., pp. 1044–47.
[8]. A. Chandra and S. Chattopadhyay, (2016)"Design of hardware efficient FIR filter: A review of the state-of-the-art approaches," Engineering Science and Technology, an International Journal, 19(1), pp. 212–226.
[9]. N. Maheshwari and S. Sapatnekar,(1998) "Efficient retiming of large circuits," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 6(1) 74– 83'
[10]. C.E. Leiserson and J.B. Saxe,(1991) "Retiming synchronous circuitry," Algorithmica, 6, 5–35..
[11]. J.R. Jiang and R.K. Brayton,(2006) "Retiming and resynthesis: a complexity perspective, " IEEE Trans. Computer-Aided Design of Integr. Circuits Syst.,25(12), 2674–86.
[12]. D. Yagain and K.A. Vijaya,(2014) "Design of synthesizable, retimed digital filters using FPGA based path solvers with MCM approach: comparison and CAD tool,' VLSI Design, Hindawi Pub. Corp., Article ID 280701.
[13]. P.K. Meher,(2016) "On efficient retiming of fixed-point circuits," IEEE Trans. VLSI Syst. 4(4).

[14]. S. Thakral, D. Goswami and R.S. Sharma,(2016) "Design and implementation of a high speed digital FIR filter using unfolding," Proc. IEEE 7th Int. Con., Bikaner, India.
[15]. C. Swati and J. Himanshu,(2015) "FIR filter designing using wallace multiplier," International Journal of Engineering and Technical Research, 3(6), 276-78.
[16]. P. Pramod and T.K. Shahana,(2020) "Efficient modular hybrid adders and Radix-4 booth multipliers for DSP applications', Microelectronics Journal, Print ISSN 0026-2692, https://doi.org/10.1016/j.mejo.2020.104701.
[17]. S.N. Rai, B.S.P. Shree and Y.P. Meghana,(2018) "Design and implementation of 16 tap FIR filter for DSP applications," Proc. Sec. Int. Conf. Adv. in Electron. Comp. and Comm., Bengaluru, India,pp. 9-10.
[18]. L. Ting, R. Woods and C. F. N. Cowan,(2005) "Virtex FPGA Implementation of a pipelined adaptive LMS predictor for electronic support measures receivers," IEEE Trans. Very Large Scale Int.Syst., 13(1), 86-95.
[19]. P.K. Meher and S.Y.Park,(2014) "Critical-path analysis and low-complexity implementation of the LMS adaptive algorithm," IEEE Trans. Circuits Syst.-I: Reg. Papers, 61(3), pp.778-788.
[20]. P.K. Meher,(2016) "Seamless pipelining of DSP circuits", J. Cir.Syst.SignalProcess.,vol.35(4), 1147–62.
[21]. F. Mathias, K. Martin and Chip-Hong,(2015) "Efficient structural adder pipelining in transposed form FIR filters," Proc. IEEE Int. Conf. Digital Signal Processing (DSP), Singapore, pp. 21-24.
[22]. P. Pramod and T.K. Shahana,(2018) "High throughput adaptive filter architecture using modified transpose form FIR filters," Journal of Advanced Research in Dynamical and Control Systems10(15), 68-82.
[23]. S. Jalaja and V.A.M. Prakash,(2018) "Different retiming transformation technique to design optimized low power VLSI architecture," AIMS Electronics and Electrical Engineering, 2(4), 117-130.
[24]. P. Pramod and T.K. Shahana,(2019) "Delay and Energy Efficient Modular Hybrid Adder for Signal Processor Architectures," IETE Journal of Research, Print ISSN: 0377-2063 Online ISSN: 0974-780X, https:// doi.org/ 10.1080/03772063.2019.1627917
[25]. P. Pramod and T.K. Shahana,(2019) "High throughput FIR filter architectures using retiming and modified CSLA based adders', IET Circuits, Devices and Systems, Print ISSN 1751-858X, Online ISSN 1751-8598, 30 https://doi.org/10.1049/iet-cds.2019.0130
[26]. R. Zhang, X. Fang, Y. Liu, J. Liao, B. Li, and M. Q.-H. Meng,(2012) "Design of a real-time ECG _lter for resource constraint computer," in Proc. IEEE Int. Conf. Inf. Autom. (ICIA), 846-849.
[27]. S. Pal and M. Mitra, (2010) Detection of ECG characteristic points using multiresolution wavelet analysis based selective coef_cientmethod,"Measurement, vol. 43(2), 225-261. [28] Li, J., Deng, G., Wei, W., Wang, H., & Ming, Z. (2017). "Design of a Real-Time ECG Filter for Portable Mobile Medical Systems.", IEEE Access, 5, 696–704.

Shikhar Bajpai. K. Raj

Shikhar Bajpai received his Bachelor of Technology degree in the field of Electronics from JSS – ATEN, Noida in year 2020. His research area includes Raman Amplifier and its efficiency. Presently he is pursuing his Masters of Technology from HBTU – Kanpur.

Krishna Raj has received the Bachelor of Engineering Degree from M.M.M.E.C. Gorakhpur, U.P., India in 1992 and he received Master of Engineering Degree from M.N.R.E.C. Allahabad, U.P., India in 1995.
He possesses Fellow membership of IETE and he is member of IEEE and IE. Presently he is professor in Electronics Engineering Department of H.B.T.U. Kanpur, U.P., India. VLSI architecture, Digital Signal Processing and Computer Arithmetic are his current research areas.