A Reconfigurable Architecture Based FIR Filter Design Using Multiplier Less Architecture

Ms. Shreedevi H Patil[1], Mr. Shreyas G[2], Mr. Ranganath N[3], Mr. Someshwara L[4], Mr. Kunjan D. Shinde[5]
[1, 2, 3,4]Dept of Electronics and Communication Engineering
[5]Asst. Professor, Dept of Electronics and Communication Engineering
[1, 2, 3, 4] PESITM, Shivamogga, Karnataka, India

Abstract- *Nowadays signal processing has become a prime task before processing of real-world data, Filtering one such basic and necessary block which is more often used in most of signal processing applications. Finite Impulse Response (FIR) filter is most preferred form in filtering process, whose impulse response is of finite period as a result of it settles to zero in finite time on the other hand an Infinite Impulse Response (IIR) filter recursive in nature and hence FIR filters are one such system which can be implemented practically. The prime building blocks of an FIR filter are Delay element, Adder & Multiplier. Design of FIR filter using these components results in conventional design methodology. In the proposed work, we have incorporated a canonical signed digit (CSD) representation for the implementation of FIR filter with multiplier-less architecture. The proposed work will be carried out with MATLAB and Xilinx tool for designing and analysing of FIR filter to process the audio signal. Verilog HDL coding will be used to design CSD based FIR filter on Xilinx and verify its functionally using the same.*

Keywords- Finite Impulse Response (FIR) Filter, Infinite Impulse Response(IIR) Filter, Canonical signed digits(CSD) Multiplier Less Architecture, Windowing.

I. INTRODUCTION

Finite impulse-response (FIR) filtering is one of the most widely used operations performed in DSP systems. FIR filter is the basic building block of any communication system. Digital filters remove noise and interference from the original signal and are used for modification of various attributes of the signal. Digital filters are more accurate, more versatile and highly stable, thereby a preferred technique over its analogue counterpart. Implementation of a general purpose multiplier in an FIR or IIR structure by means of Very Large Scale Integration (VLSI) technique is becoming a very complicated and costlier task to be performed. Solution to above problem has been found by considering multiplication operation as a repeated addition and hence substituting the filter coefficients as a sequence of shifts and additions.

This Multiplier- less filters are less power hungry and require less hardware blocks for its Implementation. In this article we have made an attempt in realizing the discrete tap coefficients by means of canonical signed digit (CSD)which is popular and efficient than the equivalent binary representation. Set of multiplier-less FIR coefficients, as optimized through algorithm. This article explains how well the proposed systems works in designing an FIR filter.

II. LITERATUREREVIEW

The following are the papers we have referred for the study and analysis of the proposed work and the work carried out by various authors are listed in brief as follows; In [1] the authors have designed FIR Filters of higher order are more accurate than of lower order filter using system generator, in the study it is noted that the design consumes more area and has complex structure at higher order to which the solution is a FPGA based implementation. In paper [2] the authors have proposed filter is two times faster FIR filter architecture than the conventional FIR filter implementation, such a design has an application over wireless communication system for swift processing. In [3] the authors have used Distributed arithmetic architecture for implementing the FIR filter, In this concept the symmetry property of FIR system is used to reduce the size while bit-parallel approach and pipelining in structure is used to increase the speed of the their proposed system for optimized cases. In [4] the authors implemented 256-Tap Parallel FIR Digital Filter Implementation Using Distributed Arithmetic Architecture. The study of [4,5] paper is future scope for our work. In [5,6,8] the authors have designed FIR filter using FCSD, DE based CSD & Shared CSD representation which are the extended version of CSD representation used. These techniques can be referred for future enhancement of the CSD that we have applied in the presented work. In [7] authors have concentrated to optimize the FIR filter in all aspects of design and observed its behavior on reconfigurable device.

III. DIGITALFILTERS

Digital filter can be thought as a linear time invariant system. It is a filter that operates on a discrete time signal and performs mathematical operation on samples/ discrete time signal to attain a descried response/ to meet certain aspect of the signal. A digital filter system usually consists of sample and hold unit along with necessary ADC, the sampled input signal is fed to the processing unit. Digital filter can be implemented on various platform based on the design requirements. For example, if we are using a microprocessor for implementing the digital filter, the necessary mathematical operations are written as a program at higher level. Where as in some high performance applications, like FPGA or ASIC the filter is implemented in the form of algorithms.

Digital filters may be more expensive than an equivalent analog filter due to their increased complexity, but they make practical many designs that are impractical or impossible to implement using analog filters. Digital filters can be designed to have very high order, and are often finite impulse response (FIR) filters. While using the same in the context of real-time implementation, some issues regarding latency (time to respond after the input has applied) associated with sample & hold unit or ADC & DAC at the input and final stage respectively has to be taken in to consideration. In this paper we have limited our study to FIR filter design on a reconfigurable platform.

FIR filter is a digital filter with digital inputs. The impulse response of FIR filter is of finite duration. These filters require no feedback therefore, also

In conclusion, a better attendance monitoring system should be developed based on its portability, accessibility and the accuracy of the collected attendance information.

$$y(n) = \sum_{k=0}^{N-1} b(k)x(n-k)$$

Where

y(n)	output samples of FIR Filter
x(n)	Input samples of FIR Filter
x(n-k)	Delayed versions of input samples
b_k	FIR filter coefficients
N	Order of the FIR Filter

Generalized FIR filter structure is shown in the figure 4 below, here we are using Direct Form I (DF-I) representation to implement the FIR filter structure, the block are represented as „D" delay units, multiplication of coefficient ⊗as & addition as ⊕. For N order FIR filter, each value of the filter is the weighted sum of the most recent values of the input as: Here, x(n) is the output sequence, h(n) represents the coefficients of filter and Y(n) is the output response.

FIR filter can be designed in mainly two structures: direct form and transposed form. Direct form FIR filter is area efficient while the transposed form filter is delay efficient property can be observed.

Figure 1 & fig. 2. Shows the direct form and transformed form representation of FIR filter, It is observed from the structure that the number of multipliers required in transposed form are reduced by half compared to direct form 1 and it may be noted that in the later stage of transposed form structure the adder is of double the bit size. A significant increase in the delay is observed which is further discussed in the results section.

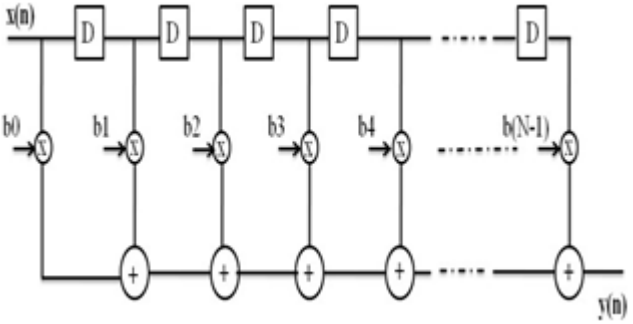

Figure.1. Generalized FIR Filter structure using Direct FormI

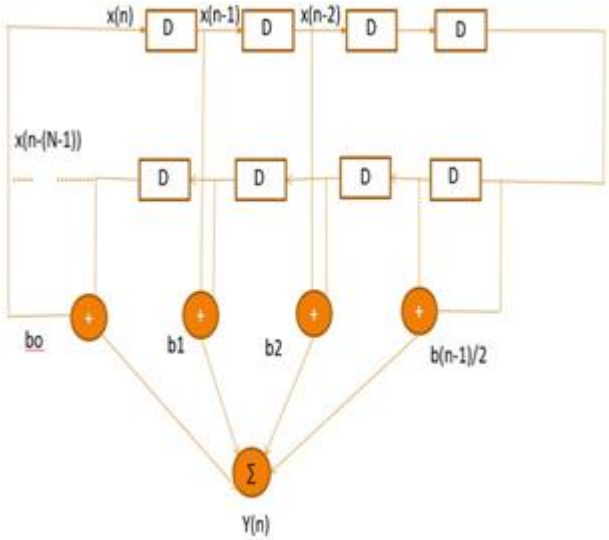

Figure 2. Generalized block diagram of Transpose structure for FIR filter

IV. SELECTION OF FILTER COEFFICIENTS

In any of the filter design the filter coefficients will describe the nature and behaviour of filter. Selecting the filter coefficients will directly results in a effective filter. Further the coefficients can be generated in many different ways some of the methods are Equiripple, least square method and Windowing method. In this paper we are not concentrating on how to generate the filter coefficients because there are several effective methods proposed early and in this paper we are concentrating on how good the filter can be implemented using various structures on to a FPGA platform.

The Equiripple and least square methods of generating coefficients leads to a ripple in the passband of the which reduces the quality of the filter and hence we are preferring these methods to generate the coefficients for FIR filter. Further, the windowing technique is popular method and the table 1 gives a comparative analysis on various windows available. From the table 1 we can opt Hamming or Blackman windowing. In our study will be opting Hamming window, as the 1st side lobe of the window is better than the blackman window and in our FIR filter design we expect the attenuation in the stopband to be around -60dB.The figures 3 to Fig.6 gives a magnitude response of the windows mentioned above. In the proposed work we are using a baseband signal as voice signal and it exits from 300Hz to 3.4KHz we are interested to pass only this signal and attenuates rest off the signal, when we implemented this FIR Filter to pass the signal effectively the order of the signal Filter may be around 300 to400.

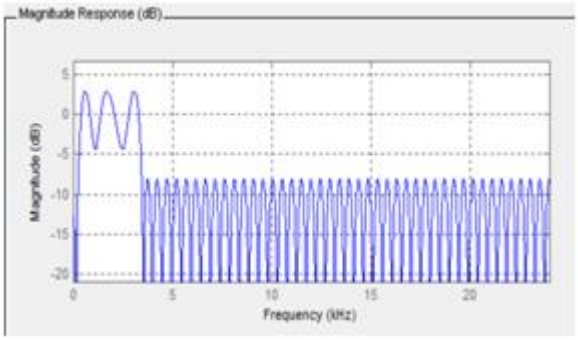

Figure 3. Filter response by Equiripple method

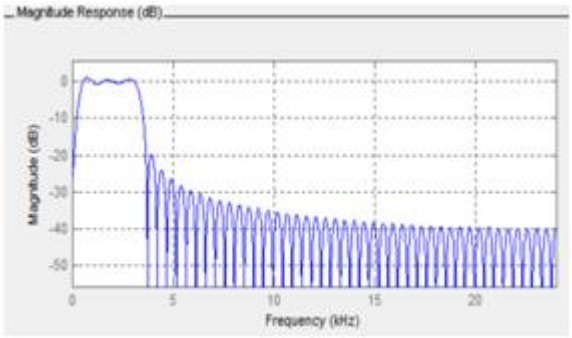

Figure 4. Filter response by Least Square method

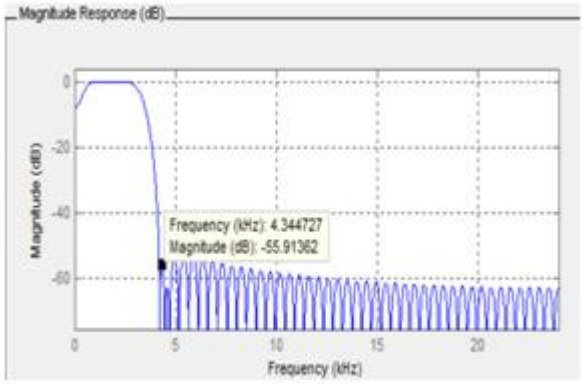

Figure 5. Filter response by Hamming windowing method

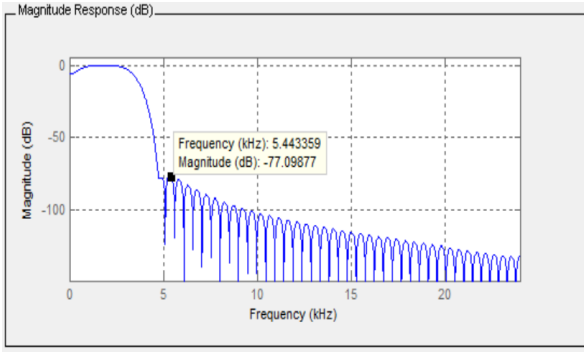

Figure 6. Filter response by Blackman windowing method

Table 1: Comparative analysis of various windowing techniques for FIR Filter

Windows	Magnitude	Order
Kaiser	-21.7db	100
Rectangular	-22.02db	100
Gaussian	-22.44db	100
Triangular	-27.45db	100
Bartlett	-28.28db	100
Bartlett-Hann	-40.93db	100
Hanning	-45.98db	100
Bohman	-52.36db	100
Hamming	-55.6db	100
Blackman	-76.29db	100

V. RESULT AND DISCUSSIONS

The following is the proposed work carried out to understand the impact of FIR filter architectures on a FPGA (reconfigurable) platform. The results are discussed in three different parts as explained below

VI. SIMULATIONS USING SIMULINK IN MATLAB

The simulation is carried on the Matlab tool with simulink and the FIR filters are designed with hamming and blackman windowing technique. For the purpose of simulation the filters order is considered as 100 & 400 to observer the

attenuation on the same. An input source with noise is fed as a discrete input to the digital filter and the result is observed on a time scope which plots the discrete responses obtained from the filter. For simplicity the sampling rate is set at 48000Hz.

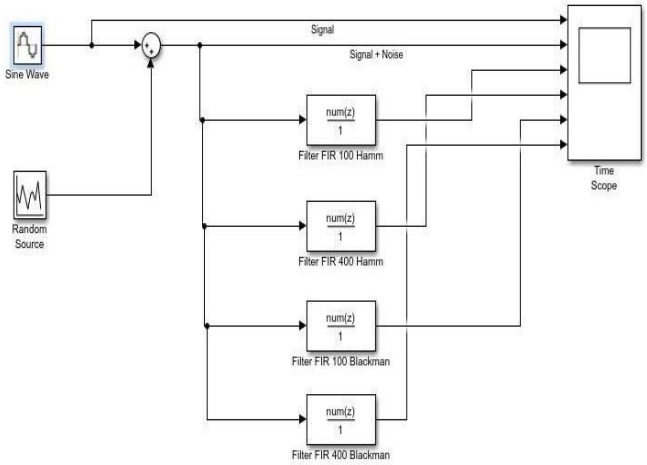

Figure 7. Simulation schematic of FIR filters using Hamming and Blackman windowing techniques with filter order as 100 and 400 respectively for both the types.

Figure 8 to Fig. 11 show the FIR filter response from frequencies 100 to 500 Hz and the same can also be observed at 3300 to 3500Hz and at rest of the frequencies (500 to 3300Hz) the response of the filter remains constant.

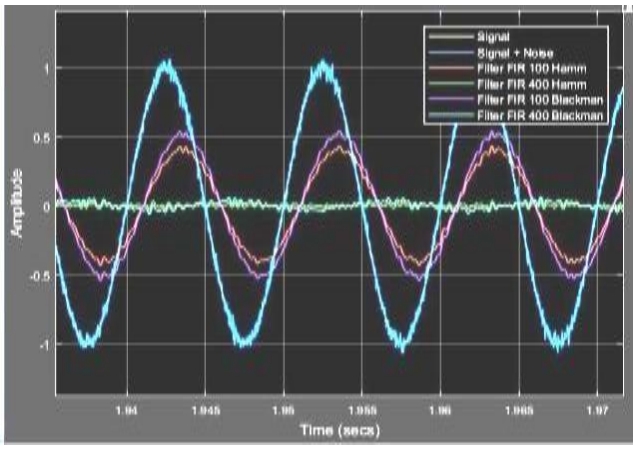

Figure 8. Simulation results for Frequency=100Hz

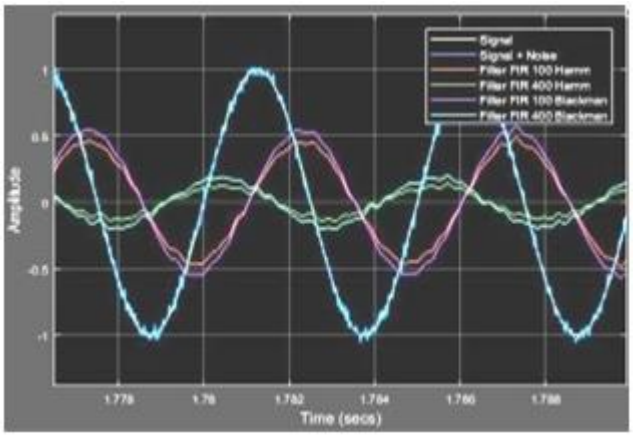

Figure 9. Simulation results for Frequency=200Hz

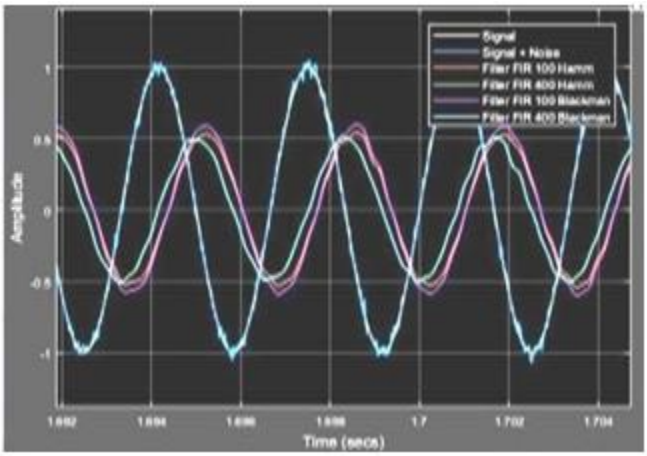

Figure 10. Simulation results for Frequency=300Hz

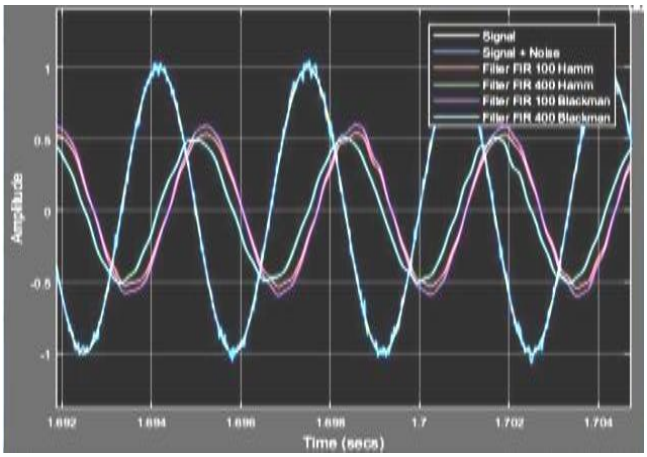

Figure 11. Simulation results for Frequency=500Hz

VII. FUNCTIONAL VERIFICATION USING HDL CODING

To measure the impact of FIR filter architectures on FPGA platform, we have opted Spartan 3 kit and used Xilinx ISE tool to simulate and observe the filter response.

Figure 12 given below shows the simulation of Verilog HDLS code for tap 10 FIR filter designed using Direct form and transposed form and using CSD technique. For a common set of input we are able to generate a uniform output sequence. This implies that the filter designed is working properly.

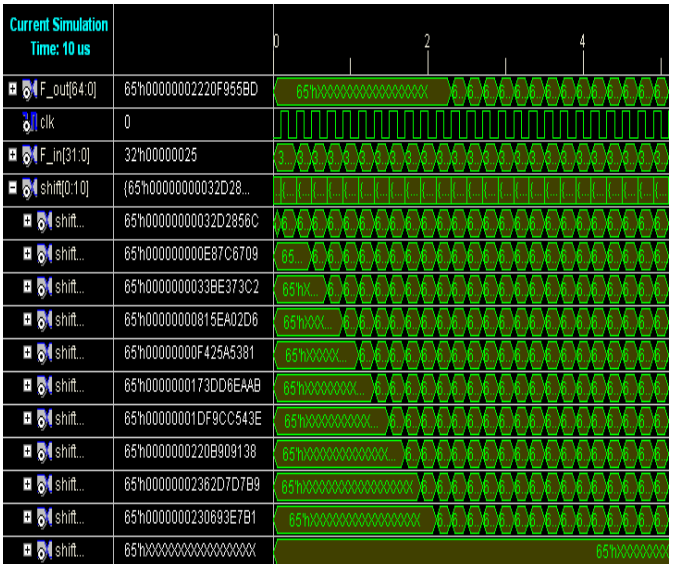

Figure 12.Simulation results for FIR filter on Xilinx ISE tool.

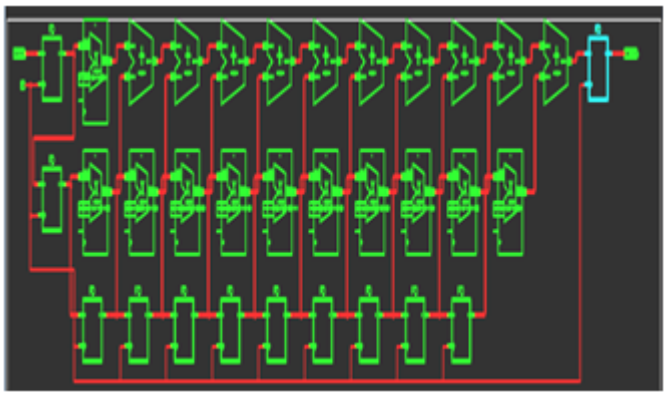

Figure 13. RTL schematic for Direct Form 1 based FIR Filter

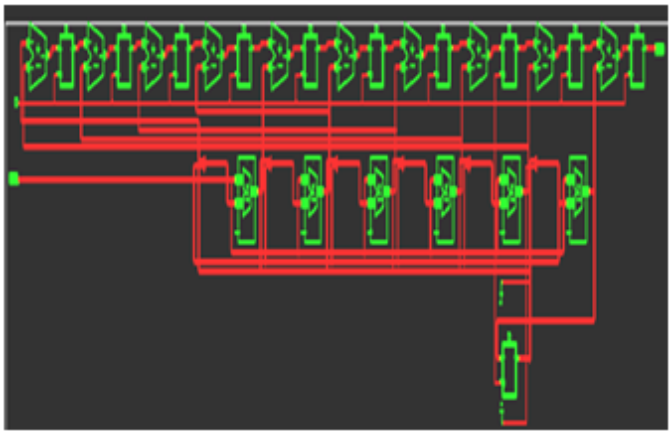

Figure 14. RTL schematic for Transpose structure based FIR Filter

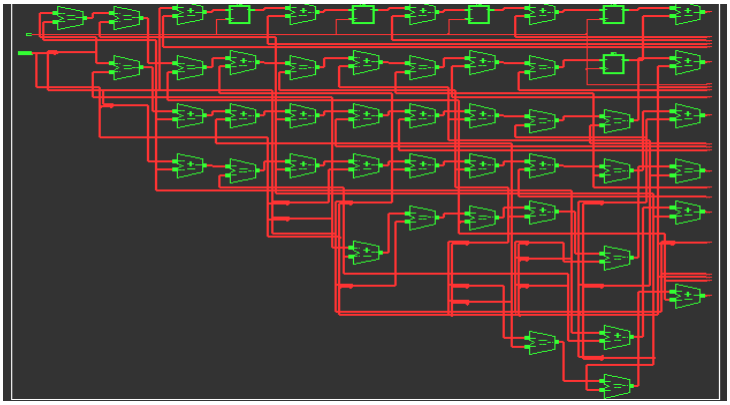

Figure 15. RTL schematic for Canonical signed digit based FIR filter

Figure 13 to Fig.15 gives the RTL schematic of the FIR filter described using various architectures, from the implementation details of FPGA it is clear that the direct form 1 is the simples way to implement the FIR filter where as transposed form less number of multipliers with wide adder in the later stage of transposed form FIR filter. The CSD based filter uses no multiplier and hence it can also be referred as a multiplier less architecture for FIR filter.

VIII. COMPARITIVE ANALYSIS

Table 2. Comparative analysis of FIR filter architectures

Parameter	Direct Form 1	Transposed Form	Transposed Form using CSD
Minimum Period	29.299ns	7.132ns	7.039ns
Maximum Frequency	34.131MHz	140.205MHz	142.069MHz
Minimum Input arrival before clock	1.572ns	15.483ns	30.075ns
Maximum output required time after clock	6.216ns	6.216ns	6.216ns
Multipliers 32 bit	11	6	0
Adder/ Subtraction	10 (64bit) Adder NA NA	NA NA 10 (65bit) Adder	32 (64bit) Adder 27 (64bit) Sub 10 (65bit) Adder
Register 32 bit	11	NA	1
64bit	01	6	6
65 bit	NA	11	12
Latch 32 bit	NA	NA	NA
Counter 32bit	NA	NA	NA
No. of Slices	3252	1931	1905
No. of 4 input LUT	6312	3169	3335

Three major architectures for FIR filter are analysed in the presented work and are tabulated in table 2, using simulink we have understood the nature of response of the filter and we came to know verified that the higher order filter gives near ideal filter response. From the simulation we have verified the same using Verilog HDL coding for lower order FIR filter. Form the comparative analysis we can say that

highest frequency response is observed in transposed form CSD based FIR filter and lowest area in the form of number of slices in FPGA can be observed and hence the CSD based design is an effective choice or filter implementation on reconfigurable platform.

IX. CONCLUSION

FIR filter with CSD architecture has a major impact on the filter architecture in the form of highest operating frequency and lowest area consumed to implement the structure.

X. FUTURE SCOPE

The variants of CSD proposed in [5,6] can be explored for further optimization of the structure.

REFERENCES

[1] Rajib Das, Amrita Guha and Ananya Bhattacharya, "FPGA Based Higher Order FIR Filter Using XILINX System Generator",111978-1-5090-4620- 1/16/$31.00 ©2016IEEE.

[2] S.Rengaprakash, M. Vignesh,N. Syed Anwar, M. Pragadheesh, E. Senthilkumar, M. Sandhyaand J. Manikandan, "FPGA Implementation of Fast Running FIR Filters",978-1-5090-4442-9/17/$31.00 c 2017 IEEE.

[3] Shaheen Khan, Zainul Abdin Jaffery, "Low Power FIR Filter implementation on FPGA using Parallel DistributedArithmetic",978-1-4673-6540-6/15/$31.00 ©2015 IEEE.

[4] AmitaNandala, T. Vigneswarn b, Ashwani K. Ranaa and Arvind Dhakac, "An Efficient 256-Tap Parallel FIR Digital Filter Implementation Using Distributed Arithmetic Architecture",AmitaNandal et al. / Procedia Computer Science 54 (2015) 605 –611.

[5] Neha Goel,Ashutosh Nandi, "Design of FIR Filter Using FCSD Representation",978-1-4799-6023-1/15$31.00 © 2015 IEEE DOI 10.1109/CICT.2015.85.

[6] Abhijit Chandra and Sudipta Chattopadhyay, "Differential Evolution Based Design of Multiplier-less FIR Filter using CSD Representation",978-1-4673- 4700-6/12/$31.00 c 2012 IEEE.

[7] Pratap Kumar Dakua, Addala Satya Narayana Varma, Kappala Jhansi and Dr P.K.Nanda, "An Optimized Design of Area Delay Power Efficient Architecture for Reconfigurable FIR Filter", 978-1-5090-4620-1/16/$31.00 ©2016IEEE.

[8] Tram Thi Bao Nguyen and Hanho Lee, "Shared CSD Complex Constant Multiplier for Parallel FFT Processors", 978-1-4673-9308-9/15/$31.00 ©2015IEEE