International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 12 | Dec 2023 www.irjet.net p-ISSN: 2395-0072

FPGA Based Design of 32 Tap Band Pass FIR Filter Using Multiplier-Less Techniques

Amit Kumar Rana[1], Nirbhay Kumar Singh[2]

[1]ME Student, Department of ECE, NITTTR Chandigarh, India
[2]ME Student, Department of ECE, NITTTR Chandigarh, India

Abstract - *This paper deals with the implementation of a 32 tap multiplier less band-pass FIR filter for digital signal processing applications. The transposed architecture is employed to design this FIR filter on a field-programmable-gate-array (FPGA) using Spartan 3E, xc3s1200efg320-5, and Virtex 4, xc4vfx20ff672-10 chips from Xilinx Inc. The VHSIC Hardware Description Language (VHDL) is used for designing this FIR filter. The main aim of this paper is to give a comparison between Spartan 3E and Virtex 4 on the basis of hardware utilization by a band-pass FIR filter. Here, the FIR filter is implemented using the Equiripple method as it meets the specifications with less complexity. Canonical Signed Digit (CSD) and Factored- Canonical Signed Digit (FCSD) representations are used to represent the filter coefficients and it is observed that the FIR filter with FCSD representation reduces the delay and area of the FIR filter. Simulation results show that the 32 tap multiplier less FIR filter using the Virtex 4 chip is 48.40% faster than the Spartan3E chip for the given specifications.*

Key Words: FIR filter; FPGA; VHDL; Filter coefficient; CSD; FCSD.

1. INTRODUCTION

Digital signal processing is the analysis, interpretation, and manipulation of real-word signals like audio, video, voice, pressure, temperature, etc. Digital signal processing techniques are used in various applications, like multimedia and communication. A basic aspect of digital signal processing is filtering. Filtering is a technique that is used to modify the frequency properties of the input signal to meet the design requirements. Filters are mostly used in applications like noise reduction, image processing, biomedical signal processing, video processing, audio processing, and the analysis of financial and economic data [1]. Filters are of two types: digital and analog filters. In applications that require flexibility and programmability, analog filters can be replaced with digital filters. Digital filters transform the digital representations of the analog signal to eliminate noise from the signal. A digital filter can be designed using either infinite impulse response (IIR) or finite impulse response (FIR) methods [2, 3]. The design of the FIR filter is chosen because it is stable, simple to implement, and has linear phase.

DSP applications require large-order FIR filters. However, due to the multiplicity of calculations, the complexity increases as the filter order increases. For an efficient digital filter, the order of the FIR filter must be as small as possible. There are basically three methods for designing FIR filters: the window method, optimal filter design method, and the frequency sampling technique. In all three methods, optimal filter design methods are the best methods to find the optimal solution in order to reduce complexity [4]. There are basically two design environments for the implementation of digital FIR filters: field-programmable gate array (FPGA)-based and digital signal processor–based. The FPGAs are the best choice for the hardware design of DSP applications, and particularly digital filters [2, 5]. The FPGAs provide flexibility in design and hardware parallelism for high-speed applications.

The main elements of digital FIR filter design on FPGAs are the register banks. The register banks are used to store the samples of the signal, multipliers for multiplication of the signal samples to filter coefficients, and an adder to implement addition operations. Although, design and implementation of the digital FIR filters are easy, they are expensive due to their large order and require more memory. In digital circuits, the design multiplier block can be replaced by using either canonical signed digit (CSD) or factored-canonical signed digit (FCSD) representation. Canonical singed digit and Factored: Canonical signed- digit representation are used for representing the filter coefficients in order to reduce the area, delay, and design complexity of the FIR filters. The effect of coefficient symmetry and replacing the multiplier blocks by shifting and addition operations are evaluated. It is important to select the proper design approach for the specific applications. FIR filters have two structures: direct form and transposed form. In direct-form structures, filter coefficients multiply the signal samples and are combined in an adder block. A modification over direct form structure is called transposed form structure, and it is employed for filter design. In this work, the design and implementation of a digital band-pass FIR filter on Spartan 3E, xc3s1200efg320-5, and Virtex 4, xc4vfx20ff672-10, are considered. This digital filter is used as a test bench to evaluate the algorithm's performance for different applications, such as radio communication [6, 7].

This paper is presented in this way: Section I presents an introduction to this work; Section II consists of an overview of the digital FIR filter; the design simulation of the FIR filter

is presented in Section III; Section IV describes FPGA synthesis; and finally, Section V concludes this paper.

2. DIGITAL FIR FILTER

A digital filter is a type of digital system that filters discrete time signals [8]. A digital filter is an algorithm that can be implemented in software or hardware, and operates on digital input signals, and also produces digital output signals. It can be represented by a block diagram as shown in Fig. 1 [2].

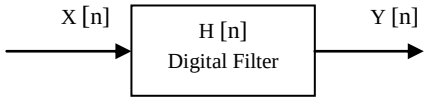

Fig. 1. Basic block diagram of Digital filter

Digital filters are the main category of linear time-invariant DSP systems, which are implemented for modification of the frequency characteristics of the input signal to meet design criteria. Digital filters have various advantages that are not present in the analog filters, examples includes excellent linear phase response, performance that does not change with environment changes, and tunable frequency response when designed with a programmable processor using adaptive filters. Adaptive filters can be used to filter multiple channels or inputs without the need to reproduce the hardware and can be operated over a wide range of frequencies. The steps for designing a digital filter are illustrated in Fig. 2 [4, 9].

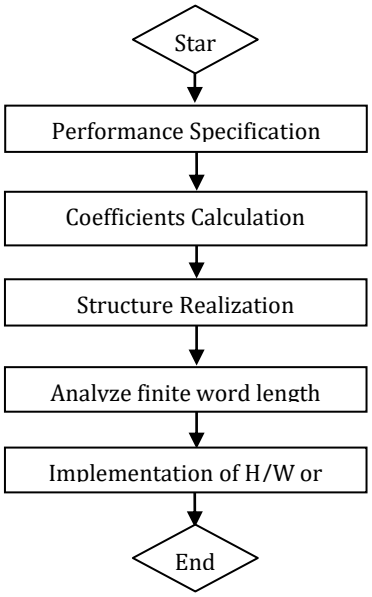

Fig. 2. Steps for FIR filter design

The digital FIR filters with constant coefficients are linear time-invariant filters. The output of an N-order digital FIR filter involves a convolution operation that could be written as equation [6]:

$$Y[n] = H[n] * X[n] \quad (1)$$

Another name for FIR filters is non-recursive digital filters because these filters do not have feedback. The response of the FIR filter can be represented as:

$$Y[n] = \sum_{k=0}^{N-1} H[k]\, X[n-k] \quad (2)$$

Where

H[k] represents the coefficients of filter.

N represents the length of the FIR filter.

Basically, FIR filters are very useful for various electronic applications where a true linear phase is required over the whole range of frequencies [4]. The implementation of FIR filters is a challenging task due to time delays, area, and power in the digital circuit design. Therefore, different architectures are shown in the literature; these are mainly of two types: direct form and transposed form structures. The direct-form realization of the FIR filter is shown in Fig. 3. A modification of the direct structure is known as a transposed structure, as shown in Fig. 4. In a transposed structure, the same input signal is multiplied by various coefficients. In this work, transposed structure is used in order to reduce delay and area as compared to direct structure [2, 15].

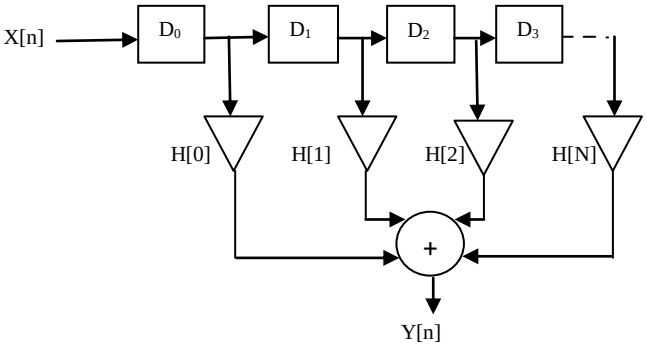

Fig. 3. Direct form of FIR filter

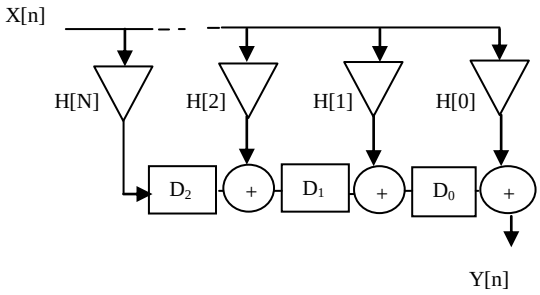

Fig. 4. Transposed form of FIR filter

Factored-Canonical Signed Digit (FCSD) representation is a modified form of Canonical Signed Digit (CSD) representation. FCSD representation replaces multiplication

operations with addition and shift operations on the basis of a prime factor of coefficients. FCSD is the combination of factorization and Canonical Signed Digit representation of filter coefficients, which reduces the number of adders and also the cost of hardware. It gives a greater reduction in filter area, but there is a decrease in clock speed. The major drawback of this algorithm is that it increases the delay. The FCSD algorithm makes a trade-off between convergence calculation and complexity [2]. This example shows the comparison between CSD and factored-CSD algorithms:

w = 105 * u
= (1101001) * u % 105 in binary representation
= (101'01001) * u % 105 in signed digit representation
= (128- 32+8+1) * u
= (u << 7) –(u << 5) + (u << 3) + u
Cost of Canonical Signed Digit is 3 adders
w = 105 * u
= (7*15) * u
= (u << 3– u) * (u << 4 – u)
Cost of Factored- Canonical Signed Digit (FCSD) is 2 adders

Therefore, the above example concludes that the number of adders can be reduced by using the FCSD technique as compared to CSD.

3. DESIGN SIMULATIONS

There are basically three methods for designing FIR filters: the window method, the optimal filter design methods, and the frequency sampling technique. In all three methods, optimal filter design methods are the best method to find the optimal solution [4].

Optimal filter design methods are of two types: one is the equirripple method, and the other is the discrete least squares method. The equirripple method is also called the Chebyshev approach method or Remez method. In this method, the mathematical process is very complex, which is difficult to realize, but the main advantage is that it has a lower order as compared to the least squares method, and this method can control the frequency edge more accurately [10].

The design of the FIR filter using the window method includes truncating of infinite time duration impulse response by using a set of time-limited weighted window functions. This method results in a very low convergence of the truncated series, especially in the vicinity of discontinuities, which makes the method unsatisfactory for approximating digital filters [11]. The aim of the window method is that the ideal frequency response of the desired filter is 1 for pass band frequencies, and for stop band frequencies, it must be 0, after which the discrete Fourier transform (DFT) of the ideal frequency response is taken for retrieving the filter impulse response. To design a finite impulse response filter, the filter coefficients must be restrained in number by multiplying a finite width window function [12]. Some of the windows commonly used are Kaiser windows, Rectangular windows, Tukey windows, Gaussian windows, Hann windows, and Hamming windows. But in this work, the Equiripple method is used for designing the FIR filter to reduce the design complexity [15].

The required parameters for the implementation of a 32 tap multiplier less band-pass FIR filter using the Equiripple method are the filter length, first stopband frequency, first passband frequency, second passband frequency, second stopband frequency and sampling frequency. In this simulation, the filter length is 32 tap, first stopband frequency is 7200Hz, and first passband frequency is 9600Hz, second passband frequency is 12000Hz, second stopband frequency is 14400Hz and sampling frequency is 48000Hz. The design specifications of a multiplier less band-pass FIR filter are given in Table I:

Table 1. Design Specifications of Band-Pass Filter

Filter Parameter	Value
Filter Length	32 Tap
First Stopband	7200Hz
First Passband	9600Hz
Second Passband	12000Hz
Second Stopband	14400Hz
Sampling Frequency	48000Hz

The fixed-point method is considered in this work because it reduces computational complexity and increases speed performance, but the major drawback of this method is that it reduces accuracy. As the magnitude response of the band-pass multiplier less FIR filter is shown in Fig. 5, the passband ripple and stopband attenuation have some fluctuations as compared to the floating point method.

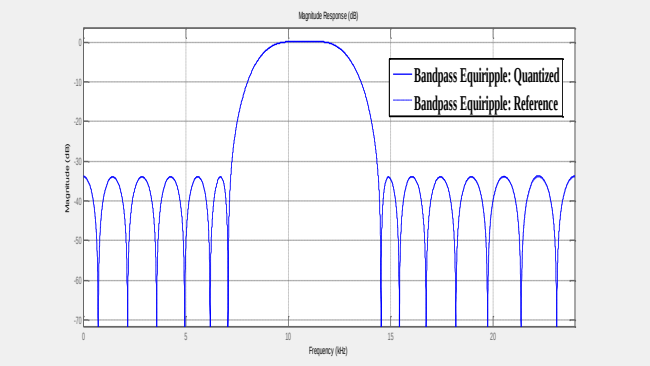

Fig. 5. Magnitude Response of Band Pass FIR Filter

The phase response of a band-pass FIR filter is shown in Fig. 6. As FIR filters have linear phase, the graph shows the linear phase characteristics up to stopband frequency.

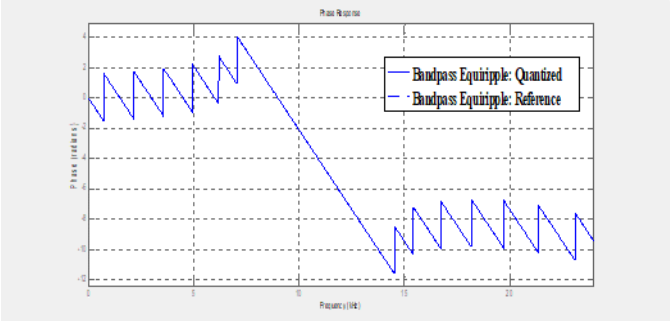

Fig. 6. Phase Response of Band Pass FIR Filter

The magnitude and phase response of a band-pass FIR filter in a single graph are shown in *Fig. 7*.

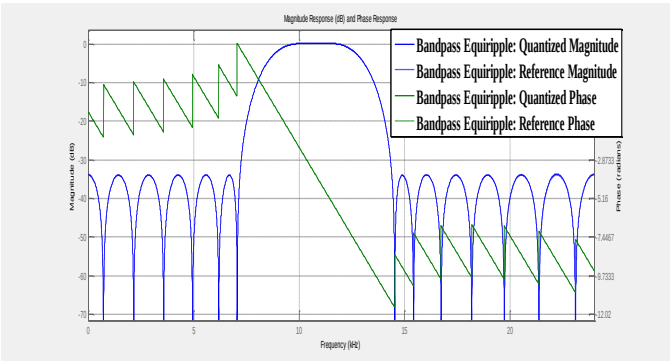

Fig. 7. Magnitude and Phase Response of the band pass FIR Filter

The pole–zero plot of FIR filter using the Equiripple method is shown in *Fig. 8*.

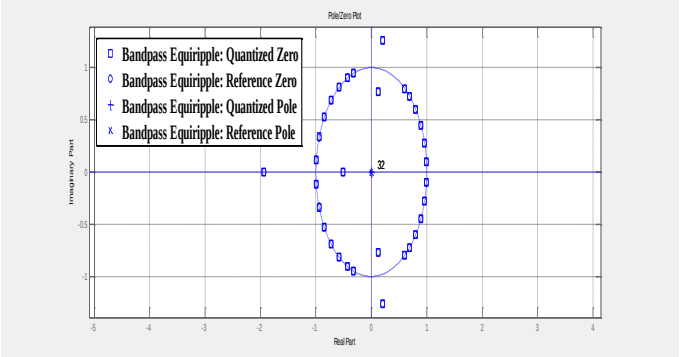

Fig. 8. Pole –Zero Plot of the Equiripple FIR Filter

4. FPGA SYNTHESIS

The multiplier less FIR filter has been designed in MATLAB. After that, it is further simulated on the FPGA using Spartan 3E, xc3s1200efg320-5, and Virtex 4, xc4vfx20ff672-10 chips. The simulation results of band pass FIR filter using Spartan 3E, and Virtex 4 are shown in Figs. 9, and 10 respectively. The filter responses on the FPGA are the same as on MATLAB.

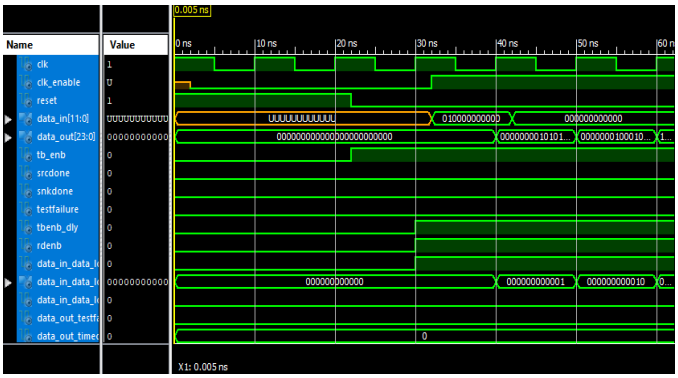

Fig. 9. FIR filter input-output using Xilinx Spartan 3E

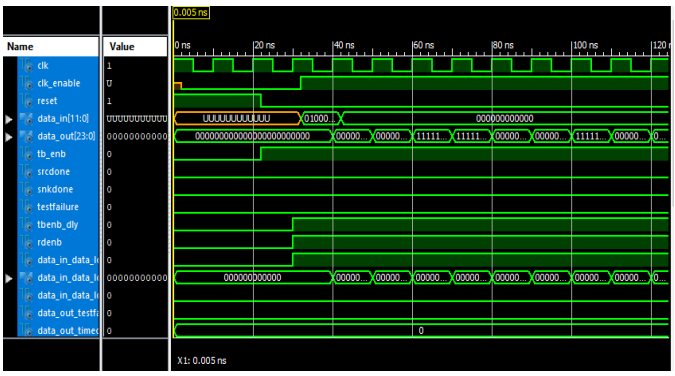

Fig. 10. FIR filter input-output using Xilinx Virtex 4

The characteristics table of Spartan 3E and Virtex 4 chips is given in Table II, and III respectively. These tables show the availability and utilization of logic by Spartan 3E and Virtex 4 chips.

Table 2. The Characteristics Table of Spartan 3E

Parameters	CSD	FCSD
Slices	1515	1463
Flip Flop	408	408
LUTs	2651	2547

Table 3. The Characteristics Table of Virtex 4

Parameters	CSD	FCSD
Slices	1505	1452
Flip Flop	402	402
LUTs	2621	2512

A 32 tap multiplier less band-pass FIR filter using CSD and FCSD has been implemented on Spartan3E and Virtex 4 chips.

VHDL is used for designing this filter. Table II, and III show that the band-pass FIR filter using CSD representation, consume 1515 slices, 408 flip flops, 2651 LUTs and FCSD representation consume 1463 slices, 408 flip flops, 2547 LUTs on Spartan 3E and on the other hand, on Virtex 4, the band-pass FIR filter using CSD representation, consume 1505 slices, 402 flip flops, 2621 LUTs and FCSD representation consume 1452 slices, 407 flip flops, 2512 LUTs. Therefore the FCSD representation consume less slices and LUTs as compare to CSD representation. Here, Spartan 3E and Virtex 4 have minimum periods of 66.696ns and 44.942ns, respectively. Hence, Spartan 3E has 48.40% more delay as compared to Virtex 4.

5. CONCLUSIONS

In this work, a design analysis of a 32 tap multiplier less digital band-pass FIR filter on an FPGA is presented. The transposed architecture is used to design this band pass FIR filter on MATLAB. After being implemented in MATLAB, the CSD and FCSD based FIR filter is further simulated on the FPGA using Spartan 3E and Virtex 4 chips. Here, FCSD representation is adopted to represent the filter coefficients as it reduces 3.55% slices and 4.08% LUTs on Spartan 3E and 3.65% slices and 4.33% LUTs on Virtex 4 as compare to CSD representation. Simulation results show that the multiplier less FIR filter using Spartan 3E and Virtex 4 is operated at a maximum frequency of 14.993MHz and 22.251MHz, respectively. Therefore, it is concluded that Virtex 4, xc4vfx20ff672-10, is 48.40% faster as compared to Spartan 3E, xc3s1200efg320-5.

REFERENCES

[1] Kumar, Bommalingaiah Nanjappa Mohan, and Rangaraju Hulivangala Gangappa. "Low area VLSI implementation of CSLA for FIR filter design." International Journal of Intelligent Engineering and Systems 12, no. 4 (2019): 90-99.

[2] Chen, Jiajia, Chip-Hong Chang, Jiatao Ding, Rui Qiao, and Mathias Faust. "Tap delay-and-accumulate cost aware coefficient synthesis algorithm for the design of area-power efficient FIR filters." IEEE Transactions on Circuits and Systems I: Regular Papers 65, no. 2 (2018): 712-722.

[3] Mittal, Anubhuti, Ashutosh Nandi, and Disha Yadav. "Comparative study of 16-order FIR filter design using different multiplication techniques." IET Circuits, Devices & Systems 11, no. 3 (2017): 196-200.

[4] Chen, Jiajia, Jinghong Tan, Chip-Hong Chang, and Feng Feng. "A new cost-aware sensitivity-driven algorithm for the design of FIR filters." Ieee transactions on circuits and systems i: regular papers 64, no. 6 (2017): 1588-1598.

[5] Rainy Chaplot and Anurag Paliwal, "Low Power Reconfigurable FIR Filter Based on Window Techniques for On Chip Network," IEEE conf., pp. 291-296, August 2013.

[6] Neha Goel and Ashutosh Nandi, "Design of Low-pass FIR Filter Using FCSD Representation" IEEE conf., on Computational Intelligence & Communication Technology, pp. 617-620, Jan 2015.

[7] Rajesh Mehra and Swapna Devi, "Area Efficient & Cost Effective Pulse Shaping Filter For Software Radios," International Journal of Ad hoc, Sensor & Ubiquitous Computing, pp. 85-91, Sep. 2010.

[8] Aye Than Mon, Zaw Min Naing, Chaw Myat New, and Hla Myo Tun, "Development OF Digital Band Pass Filter And Antenna Design For C Band Small Satellite Uplink Model" International Journal Of Scientific & Technology Research, pp. 216-222, July 2014.

[9] Bahram Rashidi, "High Performance and Low Power Finite Impulse Response Filter Based on Ring Topology with Modified Retiming Serial Multiplier on FPGA" IET Signal Processing, pp. 743-753, June 2013.

[10] Farzad Nekoei, Yousef S. Kavian, and Otto Strobel, "Some Schemes of Realization Digital FIR Filters on FPGA for Communication Applications," IEEE conf., on Microwave & Telecommunication Technology, pp. 616-619, Sept. 2010.

[11] Rajesh Mehra and Rashmi Arora, "FPGA Based Design of High-Speed CIC Decimator for Wireless Applications," International Journal of Advanced Computer Science and Applications, pp. 59-62, 2011.

[12] Lalit Maurya, Vijay Kr. Srivastava and Rajesh Mehra, "Simulink Based Design Simulations of Band Pass Filter" International Journal of Research in Engineering and Technology, pp-157-161, Feb. 2014.

[13] Emmanuel C. Ifeachor and Barrie W. Jervis, Digital Signal Processing, Pearson Education, 2nd Edition, 2008.

[14] Zhang xin-ran and Wang yu-duo, "Digital Filter Design and Analysis of BSF Based on the Best Approximation Method of Equiripple" IEEE Trans., pp. 113-116, Feb. 2014.

[15] Rachna Arya and Shiva Jaiswal, " Design of Low-pass FIR Filter Using Kaiser Window Function with variable parameter Beta (β)" Int. J. of Multidisciplinary and Current Research, pp. 220-224, March 2015.

[16] Tanveet Kaur, "Approach for Design of FIR Filter Using Kaiser Window" ISTP Journal of Research in Electrical and Electronics Engineering (ISTP-JREEE) 1st

International Conference on Research in Science, Engineering & Management, pp. 207-211, 2014.

[17] Tushar Kanti Roy and Monir Morshed, " Performance Analysis of Low-pass Filters Design Using Kaiser, Gaussian, Tukey Window Function Method" IEEE Conf., on Advances in Electrical Engineering, pp. 1-6, Dec. 2013.

[18] Kanu Priya and Rajesh Mehra, "Area Efficient Design Of Fir Filter Using Symmetric Structure," International Journal of Advanced Research in Computer and Communication Engineering, pp. 842-845, Dec. 2012

[19] S.K. Mitra, Digital Signal Processing, Tata McGraw-Hill Edition, 3rd Edition 2006.