ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 10(16), DOI: 10.17485/ijst/2017/v10i16/114305, April 2017* ISSN (Online) : 0974-5645

Reconfigurable Low Pass FIR Filter Design using Canonic Signed Digit for Audio Applications

Preethi M. Nair, Rajesh Mehra and Chandni*

National Institute of Techincal Teachers Training and Research, Chandigarh –160019, India;
preethimnair67@gmail.com, rajeshmehra@yahoo.com, chandni@baddiuniv.ac.in

Abstract

Objectives: This paper highlights the design of multiplier-less FIR filter. The binary coefficients are replaced by Canonic Signed Digit representation which reduces the complexity of the design. **Methods/Statistical Analysis:** In the current scenario more research is going on the optimization of Finite Impulse Response filters with less complex hardware design. The FIR filters performance depends on number of coefficient multipliers. The multipliers are expensive in terms delay area and power. In the CSD based filter, the number of non-zero bits is reduced. This proposed filter is designed in MATLAB, simulated in ISE environment and implemented on FPGA. **Findings:** The proposed filter is implemented on three FPGA devices, Xilinx's Spartan-3E, xc3s500e-4fg320, Virtex 2P, 2vp30ff1152-5 and Virtex 5P xc5v1x50t-3ff1136. **Improvements:** The designed structure uses reduced number of hardware components like slices, look up tables (LUTs) and flip-flops as compared to different structures and offers better performance.

Keywords: CSD, DSP, FIR, FPGA, IIR

1. Introduction

To meet the demands of digital signal processing, the emphasis is to develop the systems with minimum hardware. In this paper an efficient[1-3]. FIR filter for Audio Frequency application using CSD digit representation of filter coefficients is proposed in order to minimize hardware complexity and power consumption[2]. Digital filters have got more importance in digital signal processing. Linear Time Invariant filters are mostly used. LTI filters are classified as (i) Finite Impulse Response (FIR) filter (ii) Infinite Impulse Response (IIR) filter[1,4]. The FIR filters provide higher stability, better speed and show a linear phase characteristics. It also provides a flexibility in design.. Hence FIR applications cover the field of digital audio, data transmission and areas like image processing[5]**.** One disadvantage of FIR filter is that for same specifications the FIR filter requires higher order[5] than IIR filter. The number of multiplications and additions increases with filter order and where constraint on hardware is present, power consumption[4] and hardware cost of the filter is strongly challenged. This leads to the development of multiplier-less FIR architecture that emphasizes on the reduction of hardware complexity. In multiplier-less FIR filter, coefficients can be represented as CSD (Canonic Signed Digit) form[2,6]. This coding reduces the number of nonzero digits which permits reduced shift and add processes. This leads to the saving of hardware space complexity and power. The use of some advanced algorithms leads to the minimization of adders. This filter design ensure the performance with reduced multiplications[4,7]. It permits the subtraction and addition of shifted data. In addition to this, the redundant feature allows the implementation with less number of adders and subtractors. This leads to the implementation of the filter with high hardware compatibility. The frequency response shows errors while using frequency sampling technique at the points where it is not sampled. To reduce these errors, optimization technique must be involved. In this design least-square approach for FIR filter is used to reject the interference in the desired signal to have an optimum solution. In the least square design the transition bandwidth is small than equiripple design.

A digital filter consists of an ADC to sample the input signal followed by a processor and memory to store data and filter coefficients as shown in Figure 1. Finally

* *Author for correspondence*

DAC block is used to reproduce the output. In high performance applications FPGA, ASIC or DSP chips are used[3]. FIR filters are designed by using a finite number of samples of the impulse response. FIR filter response depends on present and past input samples.

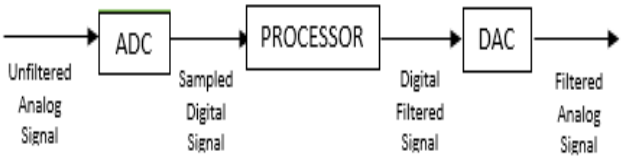

Figure 1. Block Diagram of a Digital Filter.

FIR filter of order N is given by the transfer function H(z),

$$H(z) = \sum_{k=0}^{N} h[k] z^{-k} \quad (1)$$

Also, another relation of FIR filter is given by:

$$y[n] = \sum_{k=0}^{N} h[k]\, x[n-k] \quad (2)$$

FIR filter is of order N with N+1coefficients, N+1 multipliers and number of (two input) adder is N. The equation for FIR filter in Direct Form structure is given by

$$y[n] = h[0]x[n] + h[1]x[n-1] + h[2]x[n-2] \quad (3)$$

Filter structure shown in Figure 2 is having a delay, adders and multipliers. Operands given to each multiplier are the FIR coefficients, referred to as tap weight[1,7]. In a digital filter with canonic structure the number of delays would be equal to that of the order of transfer function, otherwise the structure is non canonic. There are three methods used for FIR filter design:

- Using Window
- Using Frequency Sampling
- Optimal design

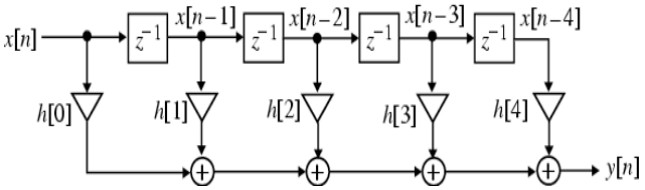

Figure 2. Direct Form Structure.

2. Design Methods

In this paper for optimal solution the filter design uses least square optimal method. By solving the linear system of equations the least square error can be found. Linear-phase filter design by least square method is non iterative and easy to design.

In lowpass filter the transfer function is given by

$$H(z) = \sum_{n=0}^{N-1} h(n) z^{-n} \quad (4)$$

The desired response is

$$D(w) = \begin{cases} 1 & 0 \le w \le w_p \\ 0 & w_s \le w \le \pi \\ d & w_p \le w \le w_s \end{cases} \quad (5)$$

Amplitude response of H(z) is

$$H(e^{jw}) = \sum_{n=0}^{N-1} h_n e^{-j(n-m)w} = \sum_{n=0}^{M} b_n \cos nw \quad (6)$$

where $M = \frac{N-1}{2}$ and

$$b_n = \begin{cases} 2h_{(N-1/2-n)} & n \neq 0 \\ h_{(N-1)/2)} & n = 0 \end{cases}$$

The leastsquare approach is used to formulate an objective function

$$E_{ls} = \int_R [D(w) - H_0(e^{jw})]^2 \frac{dw}{\pi} \quad (7)$$

R is the region $0 \le w \le \pi$ excluding transition band.

Digital signal processing requires large number of floating point multiplications[5,7]. As the number of multiplication increases it requires large number of multipliers. In applications where power, area and speed[3] are equally needed CSD makes it a great success.In order to attain high speed in multiplication the partial products in the design should keep to a minimum value. In Canonical signed digit representation it recodes a representaion with minimum number of non -zero digits[2,9]. As non–zero digit represents the number of partialproducts,reduced number of non-zero digits in CSD reduces number of partial product to be processed.Canonical-signed-digit (CSD)[6,10] is a ternary number system that uses the

sequence of one or more of the digits {−1, 0, 1} and each position has a weight of power of two that can be added or subtracted. The properties of Canonic Signed Digit (CSD) is that[2,6]

- In a CSD number no two consecutive bits are non-zero.
- In canonic signed digit coding the number of nonzero digits keeps to minimum and known as the name canonic.
- Canonic signed digit representation of a number is unique.

An n digit binary number given as $X = \{x_0, x_1 \ldots\ldots x_{n-1}\}$

$$X = \sum_{i=0}^{n-1} xi \; 2^i \quad x \quad (0,1) \qquad (8)$$

Example : Signed digit representation of 93
X_{10} = 64+16+13 = 01011101
X_{10} = 128-32-2-1 = 10-1000-1-1

Example: X= 231 * i^2
= 11100111 *i (binary form)
= 100-10100-1 *i (CSD)
= (256 – 32 + 8 - 1) *i
= (i<<8) - (i<<5) + (i<<3) – i (3 adders required)

Here only three adders are required for multiplications**.**This recoder makesit possible to save about 33% non-zeros than 2's complement form that helps to reduce complexity of hardware by replacing the multipliers by simple adder/subtractor. This paper proposes a CSD based multiplier-less reconfigurable FIR filter structure[9,10,11,12,13], as this design does not depend on number of taps. Due to the fact that number of taps and nonzero digits in each tap were arbitrarilyassigned to reduce the precision of the coefficients and there by filter complexity without degrading the performance of the filter.

3. MATLAB Design Simulation

The key objective of this paper is to implement a design on the basis of specifications. Steps to be involved in designing proposed lowpass filter are as per the filter specifications given in Table 1 using MATLAB[11] and then its synthesis on FPGA device:

Table 1. Filter Specification

Filter Type	Low pass
Structure	Direct Form 1
Design method	Least Square
Pass band frequency	4KHz
Stop band frequency	7KHz
Order (N)	25
Attenuation(pass band)	1 dB
Attenuation(stop band)	80 dB

- Define specifications of filter that include- Type of filter, in this design lowpass filter, cut off fequency, passband frequency, stopband frequency, sampling frequency passband and stop band gain, order etc.
- Detemine the required frequency response as per specification.
- Choose the efficient design method in order to minimise the cost. In this proposed design, CSD representation of filter coefficient leads to an efficient filter design.
- Add provision for optimisation,in this design least square optimisation method is involved.
- Synthesize on proper structure and proper implementation on FPGA device.
- Analyse the performance of the filter .

The required specification for the least square low pass filter are pass band frequency, stop band frequency, pass band attenuation, filter order and sampling frequency. The sampling rate is selected so as to satisfy the condition Fs > BW, here Fs= 18 KHz, and the optimal magnitude response attained by the order N= 25.The filter can be designed in two different models (i) Direct Form (ii) Transposed Form. This filter is designed in Direct Form. HDL Coder generates VHDL or Verilog code for fixed-point filter.Once the filter is designed one can view filter analyses in the display such as magnitude response, impulse response, pole zero plot, Phase response, step response etc. Figure 3 plots the magnitude response of filter as per specifications that it passes low frequencies with 1dB gain and high frequencies attenuated below 80dB. The proper selection of optimal method results in the desired magnitude response. Figure 4 plots the phase response of filter. FIR filters have the linear phase response.

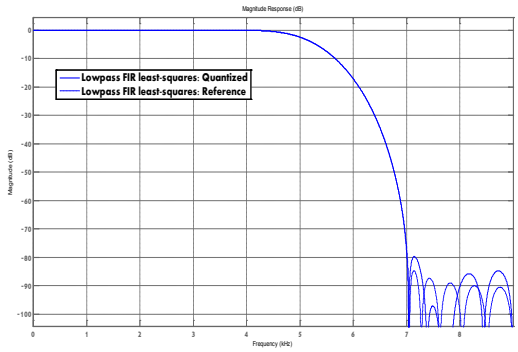

Figure 3. Magnitude Response.

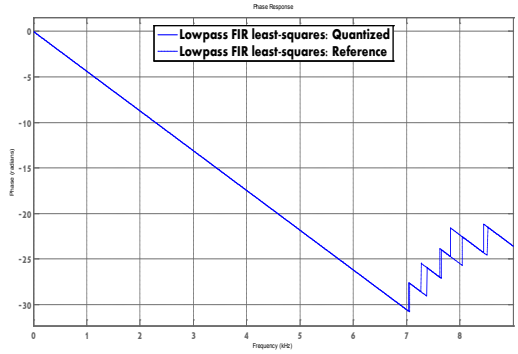

Figure 4. Phase Response.

The phase plot is linear except for discontinuities at the two frequencies where the magnitude goes to zero. The size of the discontinuities is π, representing a sign reversal. They do not affect the property of linear phase. Figure 5 shows the pole zero plot by which system stability can be observed.

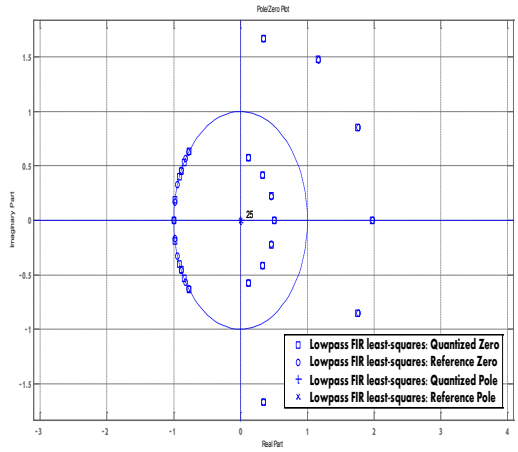

Figure 5. Pole Zero Plot.

4. Synthesis and Analysis

The proposed reconfigurable low pass FIR filter based on CSD has been simulated using Xilinx ISE simulator. The CSD based reconfigurable FIR filter architecture is intented to reduce the number of filter coefficients there by reducing the comlpexity without any degradation in performance. A 25-tap low pass filter is proposed,the filter hardware is described in VHDL and synthesized on Xilinx ISE. Figure 6 shows the proposed design simulation.

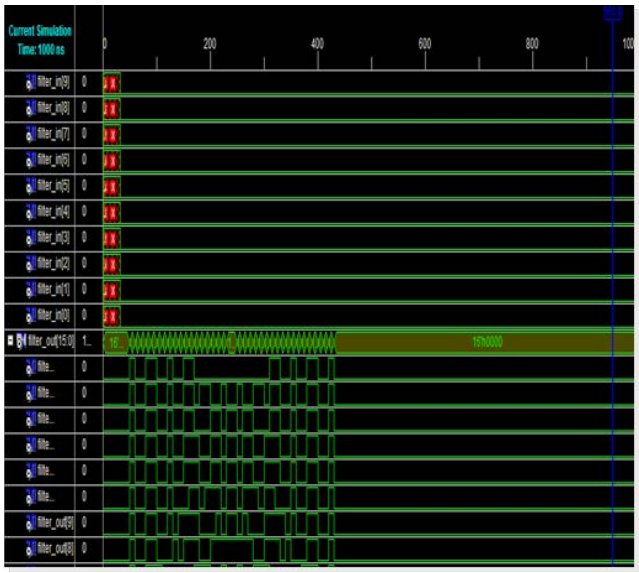

Figure 6. FIR Filter Input-Output using Xilinx.

5. FPGA Implimentation

This Filter is synthesized on XST synthesis tools Spartan 3E, Vertex2P and Virtex5. And more flexible than of conventional ASICs. When compare the performance of three tools Spartan 3E can provide better performance on short development cycle with more flexibility and less initial cost. There is no need of replacement of hardware for the up gradation in design as it provide improved programmability. The Spartan-3E structure include of programmable logic circuit known as Configurable Logic Blocks and Look –Up Tables, that include storage circuit and logic block like flip flop or registers. The data between input and output pins and internal circuits is controlled by IOBs or input output blocks. Each input output block enhances data to and from the device as well as 3-state mode of operation. The storage of data achieved by using

DDR registers (means Double Data-Rates) and RAM. In order to ensure higher performance at high density designs Virtex-2P tool is selected. Like Spartan tool this tool also includes IOBs and CLBs. The interconnection between input output pins and the internal logic circuits are provided by programmable I/O blocks. This tool allows hardware software debug at the same speed of the system and no need of verification during development. In Virtex-5, 4input LUT changed to 6 inputs LUT. Thus better handling of increasing complex functions with less number of LUTs per device. The performance analysis is done by comparing the device utilisation and speed which are summarised in Tables 2, 3 and 4. The result in Table 2 shows that the proposed FIR filter on Spartan 3E FPGA results in maximum speed of 34.401 MHz by consuming 30% Slices 4% of Flip-Flops 25% of LUTs.

Table 2. Resource Utilization and speed of Spartan 3E

Logic details	Used	Available	Utilization
Slices	1399	4656	30%
Flip Flops	432	9312	4%
LUTs	2368	9312	25%
IOBs	35	232	15%
Maximum Frequency	34.401MHz		

Table 3. ResourceUtilization and speed of Virtex2P

Logic details	Used	Available	Utilization
Slices	1399	13696	10%
Flipflops	432	27392	1%
LUTs	2368	27392	8%
IOBs	35	644	5%
Frequency	37.159 MHz		

Table 4. Resource Utilization and speed of Virtex 5

Logic details	Used	Available	Utilization
Slices:	45	2767	1%
Registers:	432	19200	2%
LUTs:	2380	19200	12%
IOBs	35	220	15%
Maximum Frequency	89.899MHz		

The proposed filter further implemented onVirtex2P. The result shows that the proposed FIR filter has achieved a speed of 37.159 MHz by using 10% Slices 1% of Flip-Flops 8% of LUTs.

The proposed filter further implemented onVirtex5. The result shows that the proposed FIR filter provides maximum speed of 89.899MHz by using 1% slices 2% of Flip-Flops 12% of LUTs. Figure 7 shows comparison of all the FPGAs in terms of resource utilization. The comparison of resource utilization is also presented in Table 5 and speed comparison in Table 6.

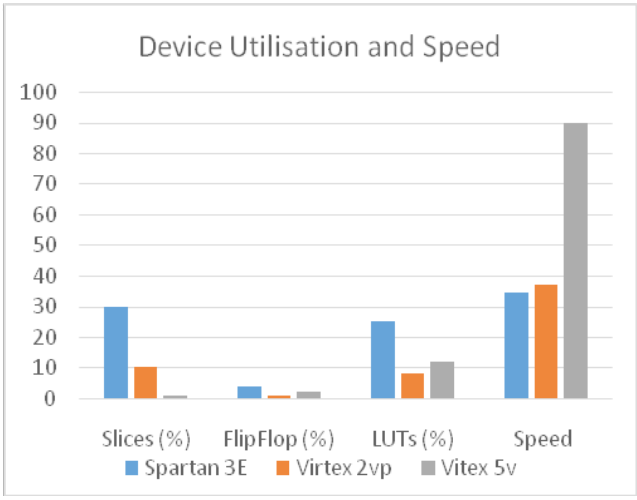

Figure 7. Resource Utilization and Speed of Spartan 3E, Virtex2P and Virtex5.

Table 5. Resource Utilization Comparison

Logic details	Spartan 3E	Virtex2P	Virtex5
Slices:	1399	1399	45
Flipflops	432	432	432
LUTs	2368	2368	2380
IOBs	35	35	35

Table 6. Speed Performance Comparison

Timing Summary	Spartan3E	Virtex2P	Virtex5
Maximum Frequency	34.401MHz	37.159MHz	89.899MHz

6. Conclusion

The design of reconfigurable low pass FIR Filter based on CSD for audio application is developed. The resource utilization table of these FPGA devices shows that Spartan 3E and Virtex2P have equal number of device utilization. Spartan 3E and Virtex2P utilize 1399 slices, 2368 LUTs and 432 Flip-Flops.Virtex2P is 8.01% faster than Spartan 3E whereas Virtex5 offers maximum speed of

89.899MHzwhich is 161.32% faster than Spartan 3E and 141.93% faster than Virtex2P by consuming45 number of Bit Slices, 432 Flip- Flops and 2380 LUTs. Hence it can be concluded that low pass FIR Filter based on CSD reduces the hardware complexity as it is multiplier-less design and implementation on Virtex5 offers better speed.

7. References

1. Priya K, Mehra R. Area efficient design of FIR filter using symmetric structure. International Journal of Advanced Research in Computer and Communication Engineering. 2012 Dec;, 1(10): 842–5.
2. Goel N, Nandi A. Design of FIR filter using FCSD representation. InComputational Intelligence & Communication Technology (CICT),.2015 IEEE International Conference.2015 Feb.p.617–20. Crossref
3. Mehra R, Devi S. Area efficient and cost effective pulse shaping filter for software radios. International Journal of Ad hoc Sensor and Ubiquitous Computing (IJASUC).2010 Sep; 1: 85–91.
4. Mehra R, Kaur R. Reconfigurable area and speed efficient interpolator using DALUT algorithm. InInternational Conference on Computer Science and Information Technology, 2011 Jan.p.117–25. Crossref
5. Mohanty BK, Meher PK. A high-performance FIR filter architecture for fixed and reconfigurable applications. IEEE transactions on very large scale integration (vlsi) systems. 2016 Feb; 24(2): 444–-52.
6. Priya K, Mehra R. FPGA based cost efficient FIR filter using Factored CSD technique.IJRTE. 2013 Jan; 1(6): 1–4.
7. Mahesh R, Vinod AP. New reconfigurable architectures for implementing FIR filters with low complexity.IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2010 Feb; 29(2): 275–88. Crossref
8. Vaidyanathan P, Nguyen T. EigenfiltersA new approach to least-squares FIR filter design and applications including Nyquist filters. IEEE Transactions on Circuits and Systems. 1987 Jan; 34(1): 11–23. Crossref
9. Vishwanath BR, Theerthesha TS. Multiplier using canonical signed digit code. International Journal for Research in Applied Science & Engineering Technology (IJRASET). 2015; 3(5): 415–20.
10. Rajolia A, Kaur M. Finite Impulse Response (FIR) filter design using Canonical Signed Digits (CSD). IJSR India Online.2013; 2 (7): 1–3.
11. Kolawole ES, Ali WH, Cofie P, Fuller J, Tolliver C, Obiomon P. Design and implementation of Low-Pass High-Pass and Band-Pass Finite Impulse Response (FIR) filters using FPGA. Circuits and Systems. 2015 Feb; 6(2): 30–48. Crossref
12. Kamble P, Deote N, Wanjari N, Gaikwad S. Implementation of FIR filter structure for audio application using XilinxSystem generator. International Journal of Advanced Research in Computer Science and Software Engineering. 2015; 5(1): 762–5.
13. Mehra R, Saini G, Singh S. FPGA based high speed BCH encoder for wireless communication applications. In Communication Systems and Network Technologies (CSNT) International Conference..2011 Jun.p.576–9. Crossref