ELECTRICAL & ELECTRONIC ENGINEERING | RESEARCH ARTICLE

Design and implementation of an efficient FIR digital filter

Sumbal Zahoor¹* and Shahzad Naseem²

Abstract: Digital signal processing (DSP) circuits are extremely important in computing and communications areas. One application of DSP is a Finite impulse response (FIR) filter. The principle objective of this exploration is to present a methodology for an upgraded framework of a FIR digital filter from software level to the hardware level. It includes the selection of design method, structure and cost effective hardware utilization. Theoretical and experimental results performed FIR band pass filter suggests that the window design method is relatively simple and easy to use because of the availability of well-defined equation. Comparison presented that Kaiser window gives the minimum main-lobe width and a sharp cut-off which means this window has less transition width and the study showed that the Direct-Form structure approach is simpler and offers a better performance than other common filter structures. It results in low cost, reduced area and more robust to withstand the quantization errors. For efficient hardware realization, the paper investigates the impacts of quantization on frequency response by progressively diminishing the quantity of bits in every coefficient using an iterative algorithm to a level where its frequency response matches to the novel requirements. Experimental study of coefficient quantization uncovers a connection between the quantity of bits, number of coefficients and the frequency response that consequence in reduced area and better speed. The synthesis results show that the discussed technique can substantial help in lessening the equipment assets. Computer aided design (CAD) tools are used to implement the design by adopting the behavioural level design method.

Subjects: Digital Signal Processing; Microelectronics; Circuits & Devices

Keywords: DSP; FIR; Kaiser window; direct form; quantization; CAD

ABOUT THE AUTHOR

Sumbal Zahoor is doing Ph D in Electrical and Electronic Engineering from the University of Lahore. After completing her B Sc in Electrical Engineering, her interest took her into university of the Punjab, where she completed her M Sc in Microelectronics Engineering by securing 2nd position. She has been working with engineering universities for more than five years till to date. Her research interests include signal processing, VLSI and image processing.

PUBLIC INTEREST STATEMENT

Digital filters are incredibly powerful, but easy to use. In fact, this is one of the main reasons that digital signal processing DSP has become so popular it offers number of advantages over analog signal processing (ASP). Well ideally the application is defined for the signal you are trying to process. It can be anything from audio, video, sensor output, data from the web, in short and simple words any sort of information. So, processing it means making the information understandable.
1. Digital filter

A digital filter is a system that performs mathematical operations on a sampled, digitized signal to reduce or enhance certain features of the processed signal. Digital filter scheme consists of a pre-filter or anti-aliasing filter to perform filtering of an input signal using a low pass filter. This is required to restrict the bandwidth of a signal to satisfy the sampling theorem. An interface is needed between the analog signal and the digital filter, this interface is known as analog-to-digital converter (ADC). After the process of sampling and converting, a digital signal is ready for further processing using an appropriate digital signal processor. The output signal that is digitized is usually changed back into analog form using digital-to-analog converter (DAC). The digital filtering process is shown in Figure 1 (Alam & Gustafsson, 2014).

Digital filter is a major topic in the field of digital signal processing (DSP). Over the past few years the field of DSP has become so popular both technologically and theoretically. The major reason for its success in the industry is due to the use of the low cost and development of software and hardware. Applications of DSP are mainly the algorithms that are implemented either in software using interactive software like MATLAB or a processor. In high-bandwidth applications FPGA, ASIC or a specialized digital signal processor are used for expediting operations of filtering. Digital filters are preferably used because they eliminate several problems associated with analog filters. There are two fundamental types of digital filters: Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) (Tian, Li, & Li, 2013).

2. FIR filters

FIR filters also known as non-recursive digital filters have a finite impulse response because after a finite time the response of FIR filter settles to zero. Block diagram of FIR filter is shown in Figure 2.

The basic structure of FIR filter consists of adders, multipliers and delay elements as shown in Figure 3.

The difference equation of nth order digital filter (FIR) can be represented as:
The transfer function $H(z)$ is given as:

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

The transfer function $H(z)$ is given as:

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{i=0}^{N} a_i z^{-i}$$

where $X(z)$ is the filter’s input and $Y(z)$ is the filter’s output. In realization, a given transfer function is used to convert into a suitable filter structure (Xu, Yin, Qin, & Zou, 2013).

The main advantages of the FIR filter design over their IIR equivalents are the following:

1. FIR filters with exactly linear phase can easily be designed.
2. There exist computationally efficient realizations for implementing FIR filters.
3. FIR filters realized non-recursively are inherently stable and free of limit cycle oscillations when implemented on a finite-word length digital system.
4. Excellent design methods are available for various kinds of FIR filters with arbitrary specifications.
5. Design and noise issues are less complex than IIR filter (Rabiner, Kaiser, Herrmann, & Dolan, 1974).

3. Design stages of digital filters

Design of a digital filter involves the following five steps:

1. Filter specification
2. Filter coefficient calculation
3. Realization
4. Analysis of finite word length effect and
5. Implementation
These five stages are interlinked as shown in Figure 4.

In first stage the required specifications of the FIR filter are defined. Whereas, in second stage window method is selected because it offers a simple and flexible way of calculating the FIR filter coefficient; due to its well-defined equations.

4. Filter designing

Filter designing and analysis tool (FDATool) is used for designing the digital filters. It is a powerful user interface for scheming and analyzing the filter’s behavior quickly in signal processing. It is used to realize quantized direct-form FIR filters Simulink model (Siauw & Bayen, 2014).

To analyze the behavior of FIR digital filter, different window functions are used by using the specifications as shown in Table 1.

Magnitude and phase responses of a 15th order digital band pass filter using Hanning, Hamming, Blackman and Kaiser window functions are observed and investigated as shown in Figures 5–8 (Jieshan & Shizhen, 2009).

Table 2 shows the brief comparison among different window functions that are used in designing.

Using the above comparison, it is analyzed that the Kaiser window is more reliable and result in high gain. Generally, the width of the main-lobe determines the transition bandwidth, while the relative heights of the side-lobes control the size of the ripples in the amplitude response. There is a bargain between main-lobe width and a height of side-lobe; in other words, both quantities cannot be reduced at the same time. Calculations show that the Kaiser window gives the minimum normalized transition width of main-lobe i.e. 0.11719 and a sharp cutoff which means this window has less transition width and introduces more ripple. This window gives simple and fast results (Patel, Kumar, Jaiswal, & Saxena, 2013).

5. Hardware realization

As the focus of this research work is to minimize the hardware implementation cost. For this, the effects of quantization by varying the number of quantization bits and analyzing the corresponding frequency responses are observed (Mehboob, Khan, & Qamar, 2009). A 15th-order band pass filter using Kaiser window is realized as double precision floating point implementation (see Figure 4) converted to a quantized filter. Float-to-fixed point conversion is required to target ASIC and fixed-point digital signal processor core which will lessen the truncation and calculation complexity.

| Table 1. Filter specifications |
|--------------------------------|
| **Parameters** | **Values** |
| Filter type | Band pass |
| Design method | FIR window ($\beta = 0.5$ for Kaiser window only) |
| Filter order | 15 |
| Lower cut off frequency 1 | 0.4 |
| Upper cut off frequency 2 | 0.5 |
Figure 5. FIR BPF using Hanning window.

Figure 6. FIR BPF using Hamming window.

Figure 7. FIR BPF using Blackman window.
The number of quantization bits are reduced from 16–3 using an iterative algorithm and keeping the frequency response exactly within limits. The frequency response of the filter without quantization is shown in Figure 9.

For low area requirement coefficients of a filter are converted to a fixed point numeric representation by quantizing at [16, 15], [8, 7], [6, 5] and [3, 2]. Where the numbers 16, 8, 6, and 3 shows the input word length or quantizing bits and the numbers 15, 7, 5, and 2 depicts the input fraction length. Quantized frequency response plots of FIR band pass filter are shown from Figures 10–13 (Beyrouthy & Fesquet, 2011).

| Window technique     | Filter order | Width of main lobe | No. of side lobes |
|----------------------|--------------|--------------------|-------------------|
| Hanning window       | 15           | 0.20313            | 4                 |
| Hamming window       | 15           | 0.17969            | 3                 |
| Blackman window      | 15           | 0.23438            | 1                 |
| Kaiser window (β = 0.5) | 15           | 0.11719            | 6                 |

Figure 8. FIR BPF using Kaiser window.

Figure 9. Frequency response of FIR band pass filter with no quantization.

Figure 10. Current filter information.
It is observed that even if the number of bits reduced to less than half of the original 16 bits, the resulting filter response has no significant change. By doing further analysis, more reduction in the quantization bits still produces no detectable deteriorating effect in the pass band, but the damage start begins in stop band attenuation and the transition band width of the filter’s frequency response (Table 3).
Table 3. Performance characteristics of filters

| Filter specification & hardware utilization | BPF 1 | BPF 2 | BPF 3 |
|--------------------------------------------|-------|-------|-------|
| No. of nets                                 | 8,812 | 5,387 | 4,674 |
| No. of instances                            | 8,793 | 6,151 | 4,433 |
| No. of gates                                | 14,482| 90,200| 8,504 |
The asset use regarding essential (Gates) components utilized is observed to be least for the proposed BPF3. The filter with the most favorable number of quantized bits is then converted to VHDL. The Simulink model of a designed filter is shown in Figure 14.

Coefficients of a direct-form band pass FIR filter are shown in Figure 15.

Designing of a filter is done using a Hardware Description Language (HDL) code. Following tools are used for the implementation of BPF (Navaid, Raaziyah, Rajesh, & Sandeep, 2013):

- ModelSim 6.5b for simulation.
- Leonardo Spectrum Level 3 for synthesis.
- Mentor’s Pyxis Custom IC Design Platform is used for design capture to schematic-driven layout.
The custom integrated circuit or ASIC design flow chart is shown in Figure 16.

HDLs offer advantages in designing the complex circuits using the top down approach and one can verify the functionality of the design early (Nekoei, Kavian, & Strobel, 2010). Behavioral level simulation of BPF is shown in Figure 17.

After verifying the behavior, the filter’s VHDL code is synthesized into a gate level netlist and represents a new structural VHDL schematic (Fuller, 2007). The Leonardo Spectrum Level 3 is used to generate the filter top module as shown in Figure 18.

The technology TSMC 0.35 $\mu m$ is selected to generate the RTL schematic that can be viewed as a gate-level schematic. The complete RTL schematic of a BPF is shown in Figure 19.
The Leonardo Spectrum is used to provide a Technology schematic that can be viewed as an architecture-specific schematic. This schematic is generated after targeting the technology in the synthesis process. It shows the design in terms of logic elements optimized to the target device or technology. The view technology schematic of the designed filter is shown in Figure 20.

The Mentor’s Pyxis IC platform simulation is performed and shown in Figure 21.

The Pyxis Layout provides a fast and flexible environment. An automated layout generated after performing the function of floor planning and the function of placing and routing is shown in Figure 22.

6. Conclusion
FIR filters are extensively used in wired, wireless communications, video, audio processing and handheld devices are preferred because of their stability and linear phase properties. This paper presents a novel design methodology for an optimized FIR digital filters from software level to the hardware level. The main goal is to encompass all the fields that are used in the efficient hardware realization of filters i.e. design method, selection of structure and the algorithm to reduce the arithmetic complexity of FIR filtering.

Theoretical and experimental result suggests that the Kaiser window gives the minimum mainlobe width i.e. 0.11719 and a sharp cutoff which means this window has less transition width, and the study showed that the direct-form structure approach is simpler, more robust to withstand the quantization errors, low cost and offers better performance than other common structures. Proposed optimized filter implementation using an appropriate quantization scheme results in reducing arithmetic complexity, area and hardware resources. Comparison revealed that the optimized filter implementation is requiring 42% less hardware resources than the normal filter implementation.
Supplemental data
Supplemental data for this article can be accessed at https://doi.org/10.1080/23311916.2017.1323373.

Acknowledgements
I would like to express gratitude to the National Institute of Electronics (NIE) and their team including Sir Shahid Ahmed Khan (Incharge IC Design Center) and special thanks to Sir Kamran Bhatti (Senior Research Officer) for providing unconditional support and technical knowledge that made possible to complete this research work in smooth way.

Funding
The authors received no direct funding for this research.

Author details
Sumbal Zahoor¹
E-mails: sumbal.zahoor@superior.edu.pk, sumbalzahoor@yahoo.com
Shahzad Naseem²
E-mail: Shahzad.Naseem@yahoo.com
¹ Department of Electrical Engineering, Superior University, Lahore, Pakistan.
² Microelectronics Research Centre, University of the Punjab, Lahore, Pakistan.

Citation information
Cite this article as: Design and implementation of an efficient FIR digital filter, Sumbal Zahoor & Shahzad Naseem, Cogent Engineering (2017), 4: 1323373.

References
Alam, S. A., & Gustafsson, O. (2014). Design of finite word length linear-phase FIR filters in the logarithmic number system domain. VLSI Design, 2014, 1–14. https://doi.org/10.1155/2014/217495
Beyrouthy, T., & Fesquet, L. (2011). An event-driven FIR filter: Design and implementation. In Proceeding 22nd IEEE International Symposium on Rapid System Prototyping (RSP) (pp. 59–65). https://doi.org/10.1109/RSP.2011.5929976
Fuller, Dr. L. (2007). Mentor graphics tools tutorial. Rochester: Rochester Institute of Technology Microelectronic Engineering.
Jieshan, L., & Shi Zheng, H. (2009). An design of the 16-order FIR digital filter based on FPGA. In Proceedings international conference on information science and engineering (ICISE), IEEE (pp. 489–492).
Mehboob, R., Khan, S. A., & Qamar, R. (2009). FIR filter design methodology for hardware optimized implementation. IEEE Transactions on Consumer Electronics, 55, 1669–1673. https://doi.org/10.1109/TCE.2009.5278041
Novaïd, Z., Raza, R., Rajesh, M., & Sandeep, S. (2013). High speed reconfigurable FPGA based digital filter. In Book: quality, reliability, security and robustness in heterogeneous networks (pp. 470–478).
Nekoei, F., Kavian, Y. S., & Strobel, O. (2010). Some schemes of realization digital FIR filter on FPGA for communication application. In Proceeding 20th international crimean conference on microwave and telecommunication technology (pp. 616–619).
Patel, R., Kumar, M., Jaiswal, A. K., & Saxena, R. (2013). Design technique of bandpass FIR filter using various window function. IOSR Journal of Electronics and Communication Engineering, 6, 52–57. https://doi.org/10.9790/2814
Rabiner, L. R., Kaiser, J. F., Herrmann, O., & Dolan, M. T. (1974). Some comparisons between FIR and IIR digital filters. Bell System Technical Journal, 53, 305–331. https://doi.org/10.1002/bltj.1974.53.issue-2
Siauw, T., & Bayen, A. (2014). An introduction to MATLAB programming and numerical methods for engineers. Cambridge, MA: Academic Press.
Tian, J., Li, G., & Li, Q. (2013). Hardware-efficient parallel structures for linear-phase FIR digital filter. In Proceedings of 56th IEEE international midwest symposium on circuits and systems (MWSCAS 2013) (pp. 995–998). https://doi.org/10.1109/MWSCAS.2013.6674819
Xu, C., Yin, S., Qin, Y., & Zou, H. (2013). A novel hardware efficient FIR filter for wireless sensor networks. In Proceedings of Fifth IEEE international conference on ubiquitous and future networks (ICUFN 2013) (pp. 197–201).