Design of Programmable Digital FIR/IIR Filter with Excellent Noise Cancellation

Pushpalatha Chandrasekar1 and Jee-Youl Ryu2
1Graduate Student, Department of Information and Communications Engineering, Pukyong National University, Busan 48513, South Korea.
2Associate Professor, Department of Information and Communications Engineering, Pukyong National University, Busan 48513, South Korea.
Corresponding author

Abstract
In this paper we propose FPGA implementation for programmable digital FIR/IIR filter with excellent noise cancellation. To realize programmable filters, sine wave generation and detecting frequency technique of the sine wave signals have been developed. Simulation results verify that the system has the ability to provide a better frequency response for various cut-off frequencies. The design has been implemented in Altera, cyclone II family EP2C70F89618 device using Quartus software. The proposed filters showed excellent noise cancellation for various noises.

Keywords: IIR filter, FIR filter, programmable, FPGA.

INTRODUCTION
The filter is used to filter out unwanted frequency components, and it extracts useful frequency components of an input signal. A digital filter is a system to perform mathematical operations on sampled values of an input signal. Programmable digital filters have prevalent applications in the various fields of communication, image processing and electronics, etc. It plays a vital role in every electronic appliances such as radios, AV receivers, cell phones, etc. There are two types of digital filters namely Finite Impulse Response Filter (FIR) and Infinite Impulse Response Filter (IIR). The FIR filter has finite impulse response. Since it doesn’t have any poles, the filter is always stable, and it satisfies the linear phase characteristics unlike an IIR filter. In contrast to FIR filter, the IIR filter has internal feedback loop which causes IIR filter to have poles. If poles exist on the unit circle or inside the unit circle, and then the filter is stable otherwise it is unstable. Thus stability condition always depends on the location of poles. Numerous publications have reported this field, and active study is still underway [1-12]. This paper presents programmable digital FIR/IIR filter with excellent noise cancellation. This circuit is implemented using FPGA (Altera, cyclone II family EP2C70F89618 device) with Quartus software. The sine wave generation and detecting frequency technique of the sine wave signals have been described to develop programmable filters.

SYSTEM MODELING AND METHODS
There are so many ways to design FIR/IIR filter. Some of them are listed below. To design FIR/IIR filters we can choose one of the methods according to our requirement and application [1].

Window method
The window method is the most common and simple method to design FIR filter. It performs mathematical function whose values are zeroes outside of chosen interval. When ideal response ($h_d(n)$) of the filter is multiplied by a window function ($w(n)$), the product also has zero valued outside the interval. Hence FIR filter’s impulse response is truncated. Ideal filter response varies with the type of filter such as low-pass, high-pass, band-pass and band-stop. As compared with other design methods, window method is so easy to design and implement.

Rectangular window
The rectangular window is a pretty simple window method among other methods. The rectangular window function is defined as in Equation (1).

$$w[n] = \begin{cases} 1, & 0 \leq n \leq N-1 \\ 0, & \text{otherwise} \end{cases}$$

(1)

Where N is filter order.

Unlike other window functions, narrow transition region is achieved. The problem with this method has more ripples (side lobes) at stop band frequency, since it has low stopband attenuation. Figure 1 shows magnitude and phase responses of a rectangular window.

Figure 1: Frequency response of rectangular window.
Frequency sampling method
Let’s imagine most direct technique when a desired frequency response has been specified. It consists simply of uniformly sampling the desired frequency response, and performing an Inverse Discrete Fourier Transform (IDFT) to get the corresponding impulse response.

Butterworth filter
This filter is designed to have a flat frequency response in the passband because of this characteristic known as maximally flat filter. It has no ripples in the passband and smoothly rolls off towards zero in the stop band.

Architecture
A filter must be realized by developing a signal flow diagram which portrays the filter in terms of operations on sample sequences. Every structure varies from other structures by its own required multipliers and complexity. Direct Form I uses 2N delay units to implement the transfer function with N, the order of the filter. This flow is suitable for filters with less order, but it is impractical for filters of higher orders. Whereas Direct Form II only needs N delay elements which are potentially half as Direct Form I. The signal flow graph for Direct Form II is shown in Figure 2, and here $Z^{-1}$ is the unit delay element. Similarly ladder and state-space, and cascaded structures are available.

![Figure 2: Direct Form II signal flow diagram.](image)

Proposed Model
Methodology includes design of programmable digital filter which is synthesized and optimized using VHDL programming language. The proposed programmable digital filter is implemented on Cyclone II based EP2C70F89618 target device. Figure 3 represents the proposed model to design digital FIR/IIR filter.

![Figure 3: Flow diagram of proposed model.](image)

Frequency Detection Algorithm
After generating the opportune sinusoidal signal, the proposed algorithm calculates the frequency of the generated signal to calculate filter coefficient for detected input signal frequency. To measure the frequency of the input signal any model may use Direct Digital Frequency Synthesis (DDFS). The DDFS has given the relation between the clock signal frequency and the generated signal frequency [6]. The given relation is described in Equation (2).

$$f_{out} = FCW \cdot \frac{f_{clk}}{2\pi}$$

Where FCW is frequency control word, $f_{clk}$ is reference clock frequency, and N is word length of phase accumulator.

We propose simple algorithm to find out the frequency of signal. The number of samples per cycle and the reference frequency play a vital role in finding frequency. Dividing the reference frequency by the counted number of samples per cycle yields the required frequency of the input sinusoidal signal. This can be written as a mathematical equation in (2). The cut-off frequency of the input signal has been found using the Equation (2).

$$F_{in} = \frac{(F_{ref})}{(N_s)}$$

Where $F_{in}$ is frequency of the input signal, $F_{ref}$ is frequency of the reference clock signal, and $N_s$ is number of sample values that completes a single cycle.

RESULTS
Sine wave generator is designed to provide sine wave using VHDL language [1]. Figure 4 depict the output of the sine wave generator. Variable sine wave generator is also made to provide four sine signals with different frequencies with the help of input pins. By varying the input combination at input pins we can generate different frequency sine signals.

![Figure 4: Generated sine wave input signal.](image)
high-pass filter by assigning the input at filter selection pin. By giving \([\text{filter selection}=00]\) it acts as a low-pass filter. Likewise by giving \([\text{filter selection}=01]\) it behaves as a high-pass filter. In the same way, we can give input at \([\text{filter selection}]\) either “10” or “11” to act as a band-pass filter or band-stop filter.

Figure 6 shows frequency response for four variable passband frequencies using Butterworth filter. The results were obtained using VHDL language for the filter length of \(L=5\) with length of the filter which is equal to number of coefficient. These results are also plotted using MATLAB software. As shown in Figure 6, magnitude and phase responses for various window function \((w(n))\) values showed good filtering properties. These results verify that the proposed filters provide programmable cut-off filtering properties for signals with various input frequencies.

**CONCLUSION**

This paper presented FPGA implementation for programmable digital FIR/IIR filter with excellent noise cancellation. The results were obtained using VHDL language for the filter length of \(L=5\) with length of the filter which is equal to number of coefficient. As the filter order increases, complexity, hardware requirements and cost also increase, but it provided frequency response which is close enough to ideal filter frequency response. The design has been implemented in Altera, cyclone II family EP2C70F89618 device using Quartus software. The proposed filters showed excellent noise cancellation for various noises.

**ACKNOWLEDGMENTS**

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2015R1D1A3A01015753).

**REFERENCES**


