Optimized Design of FIR Filter Based on FPGA

Qihui Fu, Guoqin Zhang, Xiushan Wu* and Shubin Yan
School of Electrical Engineering, Zhejiang University of Water Resources and Electric Power, Hangzhou 310018, China

*Corresponding author

Abstract. Based on the field programmable gate arrays (FPGA), a low area and high speed FIR filter is proposed and designed. The multiplication modules were replaced by adders and shift registers in the proposed architecture. This is possible because a coefficient approximation is performed, using the algorithm that computes the coefficients like a sum-of-powers-of-two (SOPOT). Compared with the traditional method, the area of the 7-tap 12 bit FIR filter occupied is 75\% smaller.

Keywords: FIR filter; FPGA; SOPOT.

1. Introduction
FPGA is now widely used in digital signal processing systems. The digital part of the digital to analog converter (ADC) uses for pulse width modulation (PWM) can be realized in ALTERA AFlex10KA device more economically, using this device can quickly measure the active power, reactive power and apparent power. The power analyzer can be realized on the FPGA, which can be used to measure the single-phase power, and the resources occupied about 60\% of the logic unit on the device [1, 2]. All operations are performed in parallel, so the response time is very short. In order to get the correct measurement, all signals must pass through a finite impulse response filter (FIR) firstly. Compare to the infinite impulse response filter (IIR) in the way of hardware, the FIR filter is easier to achieve and design that FIR filter holds a more simple and regular structure, but to reach the same filtering effect, the FIR filter often with a higher order than the IIR filter, so it must occupy a more hardware resources. Therefore, achieving the function of the FIR filter on the FPGA, the constrained condition of speed area and power consumption need to be taken into account. That is why proposed a FIR filter without multiplier modules [3].

2. The Design of FIR Filter
The outstanding characteristic of FIR filter with constant coefficients is its unit sample response \( f(n) \) is a finite sequence of \( n \) points long, \( 0 \leq n \leq N-1 \), which can be considered as a linear time invariant digital filter, the \( y(n) \) is the output of the filter, which is defined as the linear convolution of input \( x(n) \) and unit response \( f(n) \), then

\[
y(n) = \sum_{k=0}^{N-1} x(k) f(n-k) = x(n) f(n)
\]

For a linear time invariant system, the Z transform can be more easily expressed as:

\[
Y(z) = F(Z) X(z)
\]
\[ F(z) = \sum_{n=0}^{N-1} f(z)z^{-n} \]  

(3)

\( N \) order linear time invariant FIR filter can be shown in Figure 1, the implementation of a digital filter require the computing unit of the adder, multiplier and delay unit, etc.. Multiplier is a fixed coefficient multiplier, this coefficient is called the order [4].

The essence of digital signal processing is numerical operation, which can be implemented by software on the computer, and can also be implemented by hardware. Either way, in the process of its operation coefficient, some coefficients, signal sequence and the results are stored in binary form in a finite word length unit. In order to realize the FIR filter in the FPGA device, the multiplier coefficients were quantified if the multiplier’s factor is integer quadratic power, or can be expressed with two or three SOPOT terms, then the multiplier can replace with shift register. Because the error of the coefficient based two SOPOT terms is bigger, extended to three SOPOT terms, and calculate the error. Fig. 2 is the multiplier structure which use adder and shift register, the multiplier coefficients can be expressed as:

\[ f(k) = \sum_{i=0}^{3} a_{i,k} z^{b_{i}} \]  

(4)

here \( a_{i,k} \) range is \([-1,1]\); \( b_{i,k} \) range is \([-m,...,0,...,n]\), here the \( m \) and \( n \) are the dynamic range of the coefficient. When the \( m \) and \( n \) are larger, the coefficient is closer to the true value [5].

![Figure 1. N order linear time invariant FIR filter.](image1)

![Figure 2. The multiplier structure.](image2)

3. The Coefficient Calculation

The coefficients of the filter are all calculated by the software FIR editor of the ALTERA company or other special software, and once the coefficients are obtained can be converted by two SOPOT terms. The Fig.3 is the conversion coefficients algorithm, this algorithm will make the coefficients of the filter’s multiplier to convert into three SOPOT terms. To assume the algorithm in condition \( i>2 \) can replace \( i>1 \), it can be much easier to make the coefficients into a two SOPOT terms. In flowchart, \( a[i, k] \) is the coefficient \( a_{i,k} \), and the \( b[i, k] \) is the \( b_{i,k} \) in the formula (4), respectively. There are two cycle structures in the algorithm. The first is \( i \) cycle structure that is used for calculating the vector \( a[i, k] \) and \( b[i, k] \); the second is \( k \) cycle structure is used to calculate the coefficient of each multiplier in the filter. To calculate the coefficient \( f[k] \) of the algorithm when the logarithm is 2, the \( f[k] \) of the input port is the multiplier coefficient value. In the first step, the calculation depends on the difference of the coefficient \( b[0, k] \)'s secondary power, the value of \( a[i, k] \) is calculated by dividing the signed number \( f[k] \) by the absolute value of \( f[k] \).

When \( i \) is increasing, the coefficients \( a[i, k] \) and \( b[i, k] \) are calculated sequentially. If in a node or the input point, \( b[i, k] \) is equal to the word length of the multiplier factor \( n \), the filter is equivalent to multiplying the coefficient by 0, at this time, the order of the filter is 0. Filter is quite in \( n \)-bit shift operation, in this algorithm, the direction of the shift is not important, because whether shifts to the left or right shift calculation results are 0. When \( i>2 \), then \( k \) value increase, the coefficient of the next multiplier will be approximately converted, due to the complexity of the algorithm in FPGA.
implementation, these coefficients of the transformation will be fully carried out externally, and the calculated results will be configured in random access memory of the FPGA [6].

![Flowchart of conversion coefficients algorithm.](image)

**Figure 3.** The flowchart of conversion coefficients algorithm.

### 4. Results of Experiment

The proposed filter is simulated on the MATLAB software for high pass FIR filter, the Fig.4 is amplitude response when the order is 15. The coefficients of filter’s multiplier are quantified to 8-bit in the accuracy of the integer. The figure shows the amplitude response of unquantized coefficients and approximate coefficients of SOPOT 2 terms or three SOPOT 3 terms. In the Fig.4, the stop band which of the approximate coefficients of SOPOT 2 terms is different from unquantized coefficients.

![Amplitude response of 15 order high-pass FIR filter.](image)

**Figure 4.** Amplitude response of 15 order high-pass FIR filter.

But in some cases, the multiplier coefficients are approximated with two SOPOT terms may be more accurate than which with three SOPOT terms. To resources occupation on FPGA hardware, the coefficient approximate with three SOPOT terms is more conservation of resources.

The proposed filter is simulated on the MATLAB software for low pass FIR filter too, amplitude response is shown in Fig.5 when the order is 13. Only in the normalized frequency of 0.80-0.85, the
multiplier coefficients are approximated with SOPOT 2 terms is more accurate than which with SOPOT 3 terms.

Figure 5. Amplitude response of 13 order low pass FIR filter.

Fig. 6 shows the block diagram of HR filter based on three SOPOT terms. This structure is suitable for implementation on the FPGA, its coefficients can be approximated with two SOPOT terms by only a very small modification. In figure 6, the D is represented as a clock cycle delay unit, which is implemented with a D trigger. A left shift register is designed by the parameterized combination logic unit provided by the QUARTUS software. The shift register can left the input with three different digits, which is determined by the value of \( b_{i,k} \). The symbol of input is determined by the value of \( a_{i,k} \). If \( a_{i,k} = +1 \), then the symbol of input is logic 0; if \( a_{i,k} = -1 \), then the symbol of input is logic 1. In addition, the symbol "+" in the diagram represents the adder.

Figure 6. The block diagram of HR filter based on three SOPOT terms.

The structure describes in Fig. 6 implements in the Flex10k device of ALTERA. A 7-order 12-bits FIR filter implemented by this method, it occupies the number of logic unit is 1540. However, the realization of traditional method occupies the number of logic unit is 2067.
5. Conclusion
The FIR filter structure bases on three SOPOT terms is introduced in this paper, this filter has better performance and can achieve a faster speed than the traditional filter with a multiplier module. A 7-order 12-bits FIR filter implemented on FPGA, with the occupied area reducing nearly 75%. This new structure can allow designers to achieve more filters on a FPGA devices.

Acknowledgement
This research was supported by National Natural Science Foundation of China (NSFC: 61975189, 61675185). The National innovation and entrepreneurship training program for college students (201911481012, No.201911481013)

References
[1] S. Y. Park, P. K. Meher. Low-power, high-throughput, and low-area adaptive fir filter based on distributed arithmetic, IEEE T. CIRCUITS-II, 60(2013)346-350.
[2] B. Pandey, B. Das, A. Kaur, et al. Performance evaluation of fir filter after implementation on different FPGA and soc and its utilization in communication and network, WIRELESS PERS. COMMUN. (2017)1-15.
[3] L. Aksoy, P. Flores, Monteiro, José. A tutorial on multiplierless design of FIR filters: algorithms and architectures, CIRC. SYST. Signal PR. 33(2014):1689-1719.
[4] B. K. Mohanty, P. K. Meher. A high-performance energy-efficient architecture for fir adaptive filter based on new distributed arithmetic formulation of block lms algorithm. IEEE T. SIGNAL. PROCES. 61(2013) 921-932.
[5] A. Eghbali, H. Johansson. On efficient design of high-order filters with applications to filter banks and transmultiplexers with large number of channels. IEEE T. SIGNAL. PROCES. 62(2014)1198-1209.
[6] R. Jia, H. G. Yang, Y. Lin, R. Chen, Z. H. Guo. A computationally efficient reconfigurable fir filter architecture based on coefficient occurrence probability, IEEE T. COMPUT. AID. D. 35(2015)1-1.