Performance Analysis of High Speed and Area Efficient Finite Impulse Response Filters

Md. Zakir Hussain¹*, Kazi Nikhat Parvin²

¹Assistant Professor, ECED, Muffakham Jah College of Engineering and Technology, Hyderabad, India
²Assistant Professor, ECED, Bhoj Reddy Engineering College for Women, Hyderabad, India

Introduction

In today's world, we witness radios in numerous applications such as cell phones, door openers, televisions, computers, etc. Radio is a device that is capable of transmitting and receiving signals in the radio frequency. When some part of the radio's physical layers function is software-defined, then they are known to be SDR. The most common application of SDR is found in communication systems, which are cost-efficient and are flexible. Their application can also be found in intensive care units where a large number of wired and wireless electronic devices must coexist.¹ The basic block diagram of a digital communication system is shown in Figure 1. SDR is used to implement the digital part.² Many applications work at a high sampling rate; this rapidly changing sampling rates require fast multipliers and efficient filters. The performance of the system depends on filters, mixers, amplifiers modulators, and demodulators, etc. The goal must be to reduce area and increase speed performance. A higher-order FIR Filter used for various SDR applications is suggested³ using canonical signed digit (CSD) and common subexpression elimination method (CSE). A re-configurable FIR filter has been suggested using CSD.⁴ Various multipliers, and adders⁵ were used, and a comparison was drawn to reduce hardware cost, i.e., area utilization. We extensively studied the performance of the FIR filter for various computational (multipliers and adders) elements for the transposed FIR filter structure. Windowing techniques, like a rectangular, Hamming window, Hanning window were found to fit the given filter specifications. The filter was designed for a word length of 32 bits.

FIR Filter with Windowing Technique

Filters are classified into two types, digital filters, and analog filters. Digital filters contain FIR and infinite impulse

Corresponding Author: Md. Zakir Hussain, Assistant Professor, ECED, Muffakham Jah College of Engineering and Technology, Hyderabad, India, e-mail: zakirhussainsm@gmail.com

How to cite this article: Hussain, M. Z., & Parvin, K. N. (2020). Performance analysis of high speed and area efficient finite impulse response (FIR) filters. SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology, 12(1), 8-12.

Source of support: Nil

Conflict of interest: None

This paper is structured as follows. Section 2 is about FIR filters. The architecture used is discussed in section 3, followed by multipliers used in the proposed method. Simulation results are analyzed later, and later in section 5, the conclusion is drawn.

© The Author(s). 2020 Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and non-commercial reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The Creative Commons Public Domain Dedication waiver (http://creativecommons.org/publicdomain/zero/1.0/) applies to the data made available in this article, unless otherwise stated.
response (IIR). As FIR is defined for N number of values, a finite value, they are called finite impulse response. IIR filters cannot guarantee stability, and they are not linear phased following these reasons, they are undesirable.

\[ y[n] = \sum_{k=0}^{M-1} b[k] x[n-k] \]

The output here depends only upon current input and previous input, unlike IIR, which also depends on previous output values. FIR filter is shown below in Figure 2. M is the order of the filter. \( x[n] \) is the input sequence, \( b[k] \) is filter coefficients, which were calculated using the windowing technique.

The transposed direct form has been used to implement the filter. It is constructed from the direct form by following the steps given below:

- Exchanging the input and output.
- Inverting the direction of signal flow.

The main blocks of the filter are multiplier, adder, and a delay block. The multiplier multiplies input with the coefficients; adder adds the outputs from delay block and multipliers. The linear phase filters are positive or negative symmetry depending on M, as shown below equation.

\[ h(n) = \pm h(M - n - 1) \]

When M is even the number of multiplications can be reduced from M to M/2 and for odd value (M-1)/2, thereby reducing hardware utilization to half.

The filter length is 32, which is M + 1. The filter order can be calculated using Kaiser or Hermann, Rabiner developed formula. The following equations are used to calculate filter coefficients using the windowing technique.

\[ h_{\text{window}}(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{\text{window}}(w)e^{jwn} dw \]

\[ h(n) = h_{\text{window}}(n)w(n) \]

\[ H(w) = \sum_{n=0}^{M-1} h(n)e^{-jwn} \]

Direct truncation of \( h_{\text{window}}(n) \) to M terms to obtain \( h(n) \) leads to Gibbs effect due to which overshoot and ripples, before and after approximated discontinuity, occur. Rectangular, Hamming, and Hanning window are found most suitable for the given filter specifications. A comparison is drawn in these three windows. The generalized cosine windows \( W(N) \) is given below.

\[ W(n) = A - B \cos(2\pi(n+1)(N + 1) + C \cos(4\pi(n+1)(N + 1) \]

\[ n = 0, 1, 2, \ldots, n - 1 \]

The filter designed using Table 1 has symmetric coefficients. The filter coefficients were represented in fixed-point representation (Figure 3).

**Architecture**

The filter is implemented using a transposed form structure. The multiplication of filter coefficients is performed using multipliers, like CSD, booth multiplier, modified booth multiplier, Vedic multiplier. Adders, like carry save adder, carry look-ahead adder are used to add delayed outputs of multipliers.

**Vedic Multiplier**

Veda is a Sanskrit word meaning “storehouse of knowledge.” They are divided into four parts. Vedic mathematics contains 16 principles called “sutra.” “Urdhva Tiryabham” (UT) is the most popular multiplication technique. It is the fast and easiest technique to perform a multiplication operation. UT means vertically and clockwise.

The hardware architecture of the 32-bit Vedic multiplier is shown in Figure 4 and Table 2.

**Modified Booth**

This algorithm promises fast multiplication by reducing partial products. Here, multipliers LSB’s three bits are encoded to -2, -1, 0, 1, and 2. A zero is appended to the LSB, and grouping of three bits is done as shown in Figure 5.

After grouping, the respective operation is performed, as given below in Table 3.
Booth Multiplier

The booth algorithm performs multiplication.\textsuperscript{13} It examines adjacent bits in a pair of two bits. N/2 cycles are examined, and respective operation is performed as given in the Table 4. It uses a two’s complement notation of a signed binary number for multiplication.

The steps followed are:
- Choose multiplier and multiplicand.
- Only in the first step initialize accumulator with zero.
- Number of multiplicand bits is initialized in count register.
- Current LSB and previous LSB are used to decide the operation to be performed.
- After every arithmetic operation, a right shift is performed.
- Until count registers become “0,” continue the above steps in a cycle.

Canonical Signed Digit (CSD)

The CSD is used to encode a number in signed digit representation. Digits 1, 0, and -1 are used for coding.\textsuperscript{14-16} The following rules are needed to be taken care of while coding in CSD format\textsuperscript{17}.

- Two consecutive numbers must not be non-zero.
- Minimum non-zero digits must be used to represent the value.

**Results and Discussion**

Tables 5 and 6 compare the area consumed and delay taken by adders and multipliers. From the table, it is clear that the Vedic multiplier achieves minimum delay and also a minimum area. Among the adders, the ripple carry adder has minimum slice utilization and carry-save adder minimum delay. Here, MATLAB FDA toolbox is used to generate coefficients of the filter using windowing technique. The device package used in Xilinx ISE 14.7 is Virtex 6 XC6VLX760.\textsuperscript{18}

Figures 6 to 8 are simulation results of the proposed work. Table 7 shows the comparison of synthesis results; from


### Table 7: Comparison of all proposed filters based on maximum frequency

<table>
<thead>
<tr>
<th>Computational elements</th>
<th>Rectangular window</th>
<th>Maximum frequency (MHz)</th>
<th>No. of slice register</th>
<th>Hanning window</th>
<th>Maximum frequency (MHz)</th>
<th>No. of slice register</th>
<th>Hamming window</th>
<th>Maximum frequency (MHz)</th>
<th>No. of slice register</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adders</td>
<td>Multipliers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSA</td>
<td>Booth</td>
<td>61,532</td>
<td>266.38</td>
<td>57,689</td>
<td>289.6</td>
<td>61,642</td>
<td>284.368</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLA</td>
<td>CSD</td>
<td>95,113</td>
<td>284.79</td>
<td>94,555</td>
<td>284.79</td>
<td>99,624</td>
<td>284.799</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kogge</td>
<td>Modified booth</td>
<td>66,348</td>
<td>137.88</td>
<td>62,331</td>
<td>137.88</td>
<td>66,380</td>
<td>137.885</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSA</td>
<td>Vedic</td>
<td>19,215</td>
<td>289.6</td>
<td>18,816</td>
<td>289.6</td>
<td>19,884</td>
<td>280.485</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RCA</td>
<td>Booth</td>
<td>60,108</td>
<td>288.08</td>
<td>56,273</td>
<td>288.08</td>
<td>59,948</td>
<td>280.485</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLA</td>
<td>Modified booth</td>
<td>54,850</td>
<td>284.79</td>
<td>51,460</td>
<td>284.79</td>
<td>54,950</td>
<td>277.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kogge</td>
<td>CSD</td>
<td>106,777</td>
<td>140.18</td>
<td>106,076</td>
<td>140.18</td>
<td>111,548</td>
<td>140.187</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RCA</td>
<td>Vedic</td>
<td>17,830</td>
<td>279.39</td>
<td>17,408</td>
<td>279.39</td>
<td>18,830</td>
<td>280.485</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**References**


