RESEARCH ARTICLE

SAMRIDDHI Volume 12, Issue 1, 2020 Print ISSN: 2229-7111 Online ISSN: 2454-5767

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

Md. Zakir Hussain[1*], Kazi Nikhat Parvin[2]

[1]Assistant Professor, ECED, Muffakham Jah College of Engineering and Technology, Hyderabad, India

[2]Assistant Professor, ECED, Bhoj Reddy Engineering College for Women, Hyderabad, India

Abstract

This study represents the designing and implementation of a bandpass finite impulse response (FIR) filter of order 31, using windowing techniques. The frequency parameters used are of a typical GSM receiver,[19] which is one of the applications of software-defined radio (SDR). To minimize filter area, various multiplication techniques, like a canonical signed digit, *Vedic* multiplier, booth multiplier, and modified booth multiplier are used. Adders, like ripple carry adder, carry save adder, carry look ahead adder, and Kogge-Stone adder are used to add the product from the multiplier unit. A comparison between three different windows has been made. The FIR is designed in MATLAB using a windowing technique. Then, it is synthesized on Xilinx 14.7, Virtex 6 XC6VLX760, whose results are included in this paper.

Keywords: Digital filters, Finite impulse response (FIR), FPGA.

SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology (2020); DOI: 10.18090/samriddhi.v12i01.2

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 suggested[3] 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 adders[5] 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.

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.

FIR Filter with Windowing Technique

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

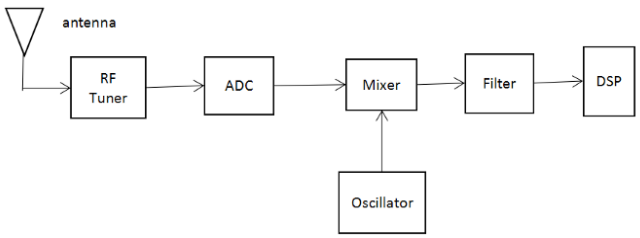

Figure 1: SDR receiver

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.[6]

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

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.[7,8]

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

The main blocks of the filter are multiplier, adder, and a delay block.[1] 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.[9,10]

The filter length is 32, which is M + 1. The filter order can be calculated using Kaiser or Hermann, Rabiner developed formula.[5] The following equations are used to calculate filter coefficients using the windowing technique.

$$h_d(n) = \frac{1}{2\Pi}\int_{-\Pi}^{\Pi} H_d(w)e^{jwn}dw$$

$$h(n) = h_d(n)w(n)$$

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

Direct truncation of $h_d(n)$ to M terms to obtain h(n) leads to Gibbs effect due to which overshoot and ripples, before and after approximated discontinuity, occur.[5,12] 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,..., n - 1

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

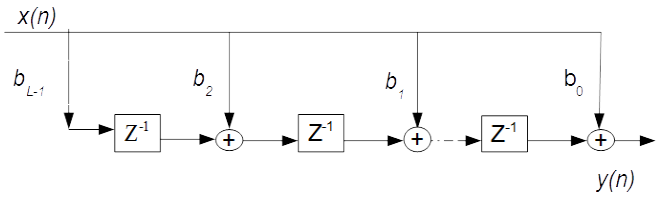

Figure 2: Transposed direct form FIR filter structure

Table 1: FIR filter design parameter

Filter parameter	*Set*
Sampling frequency	4,800 MHz
Passband frequency 1	925 MHz
Passband frequency 2	960 MHz
Order	31

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.[11] 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.[12] 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.

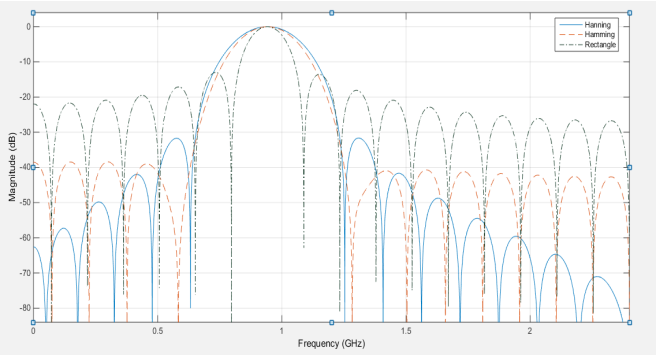

Figure 3: Magnitude response of finite impulse response (FIR) filter with given set of frequencies using different windows

Table 2: Value of coefficients for A, B, and C

Window	*A*	*B*	*C*
Rectangular	1	0	0
Hanning	0.5	0.5	0
Hamming	0.54	0.46	0

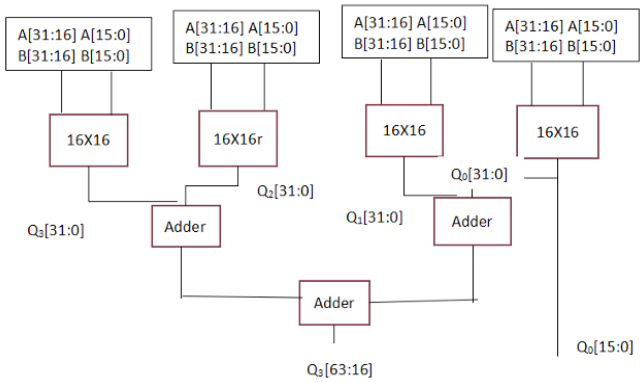

Figure 4: 32-bit *Vedic* multiplier

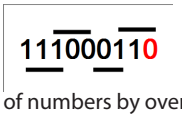

Figure 5: Grouping of numbers by overlapping technique

Booth Multiplier

The booth algorithm performs multiplication.[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.[14-16] The following rules are needed to be taken care of while coding in CSD format.[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.[18]

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

Table 3: Modified booth recoding table

Modified booth recording table multiplier bits			*Encoded multiplier value Σ*	*Partial products*	*Operation*
Qi + 1	*Qi*	*Qi - 1*			
0	0	0	0	0 M	No action
0	0	1	1	1 M	Add
0	1	0	1	1 M	Add
0	1	1	2	2 M	Shift left and add
1	0	0	-2	-2 M	Shift left and subtract
1	0	1	-1	-1 M	Subtract
1	1	0	-1	-1 M	Subtract
1	1	1	0	0 M	No action

Table 4: Operations to be performed

bi bi-1	Arithmetic operation
0 0	No arithmetic operation
1 1	No arithmetic operation
0 1	Add multiplicand to the current value of the accumulator product
1 0	Subtract multiplicand from the current value of the accumulator product

Table 5: Comparison of 32-bit multipliers

Attributes	*Vedic multiplier*	*Booth multiplier*	*Modified booth*	*Canonical signed digit multiplier*
No. of slices	1,364	2,001	2,203	2,122
No. of LUT's	2,486	4,001	4,139	3,896
Delay (ns)	32.146	128.091	89.884	63.897

Table 6: Comparison of 64-bit adders

Attributes	*Kogge-Stone*	*Ripple carry adder*	*Carry save adder*	*Carry look-ahead adder*
No. of slices	329	74	188	107
No. of LUT's	599	128	334	176
Delay (ns)	22.889	72.616	20.248	58.372

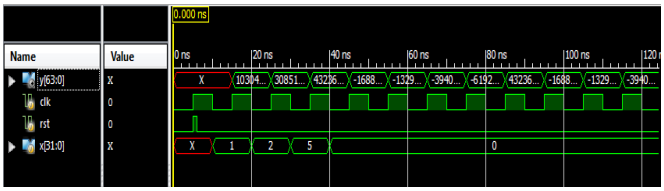

Figure 6: Ripple carry adder with booth multiplier for rectangular window

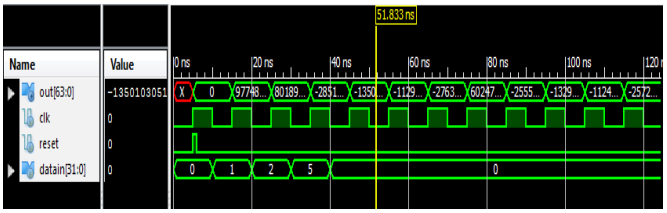

Figure 7: Ripple carry adder with booth multiplier for Hanning window

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

Computational elements		*Rectangular window*		*Hanning window*		*Hamming window*	
Adders	*Multipliers*	*No. of slice register*	*Maximum frequency (MHz)*	*No. of slice register*	*Maximum frequency (MHz)*	*No. of slice register*	*Maximum frequency (MHz)*
CSA	Booth	61,532	266.38	57,689	289.6	61,642	266.386
CLA	CSD	95,113	284.79	94,555	284.79	99,624	284.799
Kogge	Modified booth	66,348	137.88	62,331	137.88	66,380	137.885
CSA	*Vedic*	19,215	289.6	18,816	289.6	19,884	266.386
RCA	Booth	60,108	288.08	56,273	280.48	59,948	280.485
CLA	Modified booth	54,850	284.79	51,460	284.79	54,950	277.4
Kogge	CSD	106,777	140.18	106,076	140.18	111,548	140.187
RCA	*Vedic*	17,830	279.39	17,408	279.39	18,325	280.485

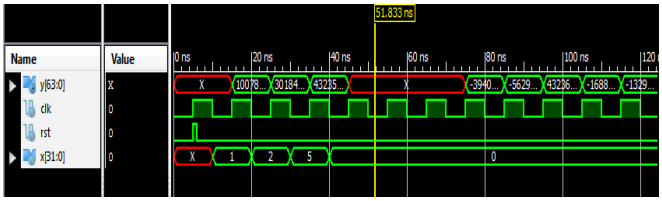

Figure 8: Ripple carry adder with booth multiplier for Hamming window

this table, it is observed that RCA with *Vedic* multiplier and CSA with *Vedic* multiplier, consumes less slices for hanning window. 111,548 slices are the maximum slice consumption found by Hamming window.[19]

17,408 is the minimum slice consumption, which is achieved by the combinations of RCA and *Vedic* multiplier. A 60% increase in slice utilization is observed from Hanning to a rectangular window. The designed filters have a wide transition width. This can be reduced by increasing the filter order. 137.885 MHz is the minimum, maximum frequency for the combination of Kogge-Stone adder and modified booth. 289.607 MHz is the maximum frequency that is less delay. These results can be considered while designing a receiver, etc.[20]

Conclusion

This work demonstrates important decisions to make while opting for the filter computational elements for an area-efficient filter or minimal delay filter. From the above results, which are shown in Table 7, it is evident that *Vedic* multiplier with CSA and RCA offers minimum slice utilization for the Hanning window. Among all the three windows, *viz*., rectangular window, Hanning window, and Hamming window. Hanning window utilizes fewer slices for almost all the combinations, and the rectangular window utilizes high. From Table 7 and Figure 8, *Vedic* multiplier with RCA has a minimum delay, whereas Kogge-Stone adder with modified booth multiplier or CSD multiplier has a maximum delay. For area constraints, filter with RCA and CSA combinations can be utilized similarly for delay filter designed using RCA, and *Vedic* multiplier can be used.

References

[1] Pathan, A., Memon, T. D., Keerio, S., & Kalwar, I. H. (2016, November). FPGA Based performance analysis of multiplier policies for FIR filter. In 2016 International Conference on Advances in Electrical, Electronic and Systems Engineering (ICAEES) (pp. 17-20). IEEE.

[2] Mittal, A., Nandi, A., & Yadav, D. (2017). Comparative study of 16-order FIR filter design using different multiplication techniques. IET Circuits, Devices & Systems, 11(3), 196-200.

[3] Cavanagh, J. (2017). Computer arithmetic and Verilog HDL fundamentals. CRC Press.

[4] Kant, A., & Sharma, S. (2015, September). Applications of Vedic multiplier designs-a review. In 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions) (pp. 1-6). IEEE.

[5] Ifeachor, E. C., & Jervis, B. W. (2002). Digital signal processing: a practical approach. Pearson Education.

[6] Kyritsis, E., & Pekmestzi, K. (2016, May). Hardware efficient fast FIR filter based on Karatsuba algorithm. In 2016 5th International Conference on Modern Circuits and Systems Technologies (MOCAST) (pp. 1-4). IEEE.

[7] Gnanasekaran, M., & Manikandan, M. (2014, July). Low delay-high compact FIR filter using reduced wallace multiplier. In Second International Conference on Current Trends In Engineering and Technology-ICCTET 2014 (pp. 400-404). IEEE.

[8] Paul, A., Khan, T. Z., Podder, P., Hasan, M. M., & Ahmed, T. (2015, February). Reconfigurable architecture design of FIR and IIR in FPGA. In 2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 958-963). IEEE.

[9] Nekoei, F., Kavian, Y. S., & Strobel, O. (2010, September). Some schemes of realization digital FIR filters on FPGA for communication applications. In 2010 20th International Crimean Conference" Microwave & Telecommunication Technology" (pp. 616-619). IEEE.

[10] Thingom, I., & Khundrakpam, P. (2016, March). FPGA implementation of FIR filter using RADIX-2 r. In 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET) (pp. 1524-1528). IEEE.

[11] B.Parhami,Computer Arithmetic,Algorithms and hardware Designs,Oxford University,Press,London,U.K ,2000.

[12] Maskell, D. L. (2007). Design of efficient multiplierless FIR filters. IET Circuits, Devices & Systems, 1(2), 175-180.

[13] Lyons, R. G. (2004). Understanding digital signal processing, 3/E. Pearson Education India.

[14] Jackson, L. B. (2013). Digital Filters and Signal Processing: With MATLAB® Exercises. Springer Science & Business Media.

[15] Jawahar, A., & Latha, P. P. (2016, October). Implementation of high-order FIR digital filtering for software defined radio receivers. In 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES) (pp. 1452-1456). IEEE.

[16] Gupta, R. R., & Mehri, R. (2015, December). Analysis & implementation of high cascaded integrated comb for software defined radios application. In 2015 Third International Conference on Image Information Processing (ICIIP) (pp. 97-101). IEEE.

[17] Chávez-Santiago, R., Mateska, A., Chomu, K., Gavrilovska, L., & Balasingham, I. (2013, July). Applications of software-defined radio (SDR) technology in hospital environments. In 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 1266-1269). IEEE.

[18] T. Hentschel and G. Fettweis, "Software radio receivers," in CDMA Techniques for Third Generation Mobile Systems. Dordrecht, The Netherlands: Kluwer, 1999,pp. 257–283.

[19] Hunter, I. (2001). Theory and design of microwave filters (No. 48). Iet.

[20] Khan, S. A. (2011). Digital design of signal processing systems: a practical approach. John Wiley & Sons.