

An New Implementation for Multi Ordered FIR Filters with Low Complexity

Mendem Naga Kumar ¹ | K Satyanarayana ²

¹PG Scholar (VLSI), Department of ECE, Sri Vani School of Engineering, Chevuturu, Andhra Pradesh, India.

²Assistant Professor, Department of ECE, Sri Vani School of Engineering, Chevuturu, Andhra Pradesh, India.

To Cite this Article

Mendem Naga Kumar and K Satyanarayana, "An New Implementation for Multi Ordered FIR Filters with Low Complexity", *International Journal for Modern Trends in Science and Technology*, Vol. 04, Issue 12, December 2018, pp.-18-23.

Article Info

Received on 30-Oct-2018, Revised on 22-Nov-2018, Accepted on 01-Dec-2018.

ABSTRACT

Low power and low complexities are the two key requirements of finite impulse response (FIR) filters employed in multi standard wireless communication systems. In this paper, two new reconfigurable architectures of low complexity FIR filters are proposed, namely constant shifts method and programmable shifts method. The proposed FIR filter architecture is capable of operating for different word length filter coefficients without any overhead in the hardware circuitry. We show that dynamically reconfigurable filters can be efficiently implemented by using common sub expression elimination algorithms. Design examples show that the proposed architectures offer good area and power reductions and speed improvement compared to the best existing reconfigurable FIR filter implementations in the project.

*Copyright © 2018 International Journal for Modern Trends in Science and Technology
All rights reserved.*

I. INTRODUCTION

FIR digital filters find extensive applications in mobile communication systems for applications such as channelization, channel equalization, matched filtering, and pulse shaping, due to their absolute stability and linear phase properties. The filters employed in mobile systems must be realized to consume less power and operate at high speed. Recently, with the advent of software defined radio (SDR) technology, finite impulse response (FIR) filter research has been focused on reconfigurable realizations. The fundamental idea of an SDR is to replace most of the analog signal processing in the transceivers with digital signal processing in order to provide the advantage of flexibility through reconfiguration. This will enable different air-interfaces to be implemented on a single generic

hardware platform to support multi standard wireless communications.

Wideband receivers in SDR must be realized to meet the stringent specifications of low power consumption and high speed. Re-configurability of the receiver to work with different wireless communication standards is another key requirement in an SDR. The most computationally intensive part of an SDR receiver is the channelizers since it operates at the highest sampling rate. It extracts multiple narrowband channels from a wideband signal using a bank of FIR filters, called channel filters. Using poly phase filter structure, decimation can be done prior to channel filtering so that the channel filters need to operate only at relatively low sampling rates. This can relax the speed of operation of the filters to a good extent. However due to the stringent adjacent channel attenuation specifications of wireless

communication standards, higher order filters are required for canalization and consequently the complexity and power consumption of the receiver will be high. As the ultimate aim of the future multi-standard wireless communication receiver is to realize its functionalities in mobile handsets, where its full utilization is possible, low power and low area implementation of FIR channel filters is inevitable. The filter multiplications are done via state machines in an iterative shift and add component and as a result of this there is huge savings in area. For lower order filters, the approach offers good trade-off between speed and area. But in general, the channel filters in wireless communication receivers need to be of high order to achieve sharp transition band and low adjacent channel attenuation requirements. For such applications, the approach in results in low speed of operation. The complexity of FIR filters is dominated by the complexity of coefficient multipliers. It is well known that the common sub expression elimination (CSE) methods based on canonical signed digit (CSD) coefficients produce low complexity FIR filter coefficient multipliers. The goal of CSE is to identify multiple occurrences of identical bit patterns that are present in the CSD representation of coefficients, and eliminate these redundant multiplications. A modification of the 2-bit CSE technique for identifying the proper patterns for elimination of redundant computations and to maximize the optimization impact was proposed, the technique in was modified to minimize the logic depth (LD) (LD is defined as the number of adder-steps in a maximal path of decomposed multiplications) and thus to improve the speed of operation.

The Proposed binary common sub expression elimination (BCSE) method which provided improved adder reductions and thus low complexity FIR filters compared to, a method based on the pseudo floating point method was used to encode the filter coefficients and thus to reduce the complexity of the filter. But the method in is limited to filter lengths less than 40.

II. FINITE IMPULSE RESPONSE (FIR) FILTERS

A finite impulse response (FIR) filter is a filter structure that can be used to implement almost any sort of frequency response digitally. An FIR filter is usually implemented by using a series of delays, multipliers, and adders to create the filter's output.

Figure 1 shows the basic block diagram for an FIR filter of length N. The delays result in operating on prior input samples. The h_k values are the coefficients used for multiplication, so that the output at time n is the summation of all the delayed samples multiplied by the appropriate coefficients.

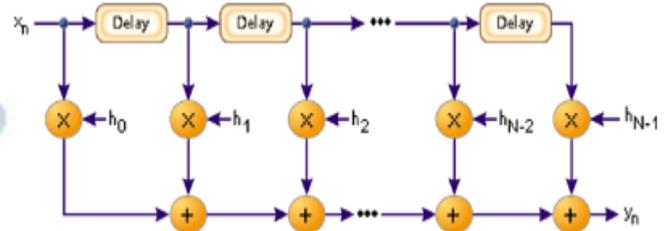


Figure 1: The logical structure of an FIR filter

The process of selecting the filter's length and coefficients is called filter design. The goal is to set those parameters such that certain desired stop band and pass band parameters will result from running the filter.

In signal processing, the function of a filter is to remove unwanted parts of the signal, such as random noise, or to extract useful parts of the signal, such as the components lying within a certain frequency range.

The following block diagram illustrates the basic idea.



There are two main kinds of filters, analog and digital. They are quite different in their physical makeup and in how they work.

A finite impulse response (FIR) filter is a type of a signal processing filter whose impulse response (or response to any finite length input) is of finite duration, because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which have internal feedback and may continue to respond indefinitely (usually decaying). The impulse response of an Nth-order discrete-time FIR filter lasts for N+1 samples, and then dies to zero.

FIR filters can be discrete-time or continuous-time, and digital or analog.

Equation (1) describes an FIR filter of length K:

$$y[n] = \sum_{k=0}^{K-1} a_k x_{[n-k]} \dots\dots\dots (1)$$

For a discrete-time FIR filter, the output is a weighted sum of the current and a finite number of previous values of the input. The operation is

described by the following equation, which defines the output sequence $y[n]$ in terms of its input sequence $x[n]$:

$$y[n] = b_0x[n] + b_1x[n - 1] + \dots + b_Nx[n - N]$$

$$y[n] = \sum_{i=0}^N b_i x[n - i]$$

N is the filter order; an N th-order filter has $(N + 1)$ terms on the right-hand side. The $x[n - i]$ in these terms are commonly referred to as taps, based on the structure of a tapped delay line that in many implementations or block diagrams provides the delayed inputs to the multiplication operations.

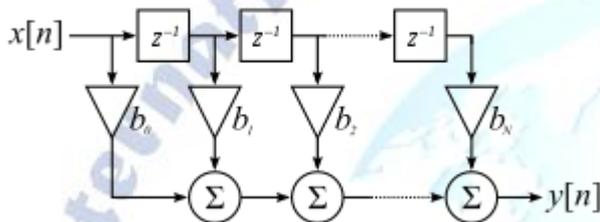


Figure:2 Basic Structure of an FIR Filter

III. DESIGN OF LOW COMPLEXITY CSM BASED FIR FILTER

By using the below three techniques proposed FIR filter with low complexity is constructed.

- Constant Shift Method(C.S.M)
- Programmable Shift Method(P.S.M)
- Common Sub Expression Algorithm

In this project, proposed the two architectures that integrate re-configurability and low complexity to realize FIR filters. The FIR filter architectures proposed are called constant shifts method (CSM) and programmable shifts method (PSM). In this project, an elaborate of the CSM and PSM architectures introduced by providing the detailed design. The design analysis of the architectures and their extension to high-level synthesis are presented. Also implemented two CSD based methods based on our CSM and PSM to compare the complexities of the CSD and binary based CSE techniques. The proposed architectures consider coefficients as constants (as they are stored in LUTs) and input signal as variable. The coefficient multiplication in such a case is known as multiple constant multiplications (MCM), i.e., multiplication of one variable (input signal) with multiple constants (filter coefficients).

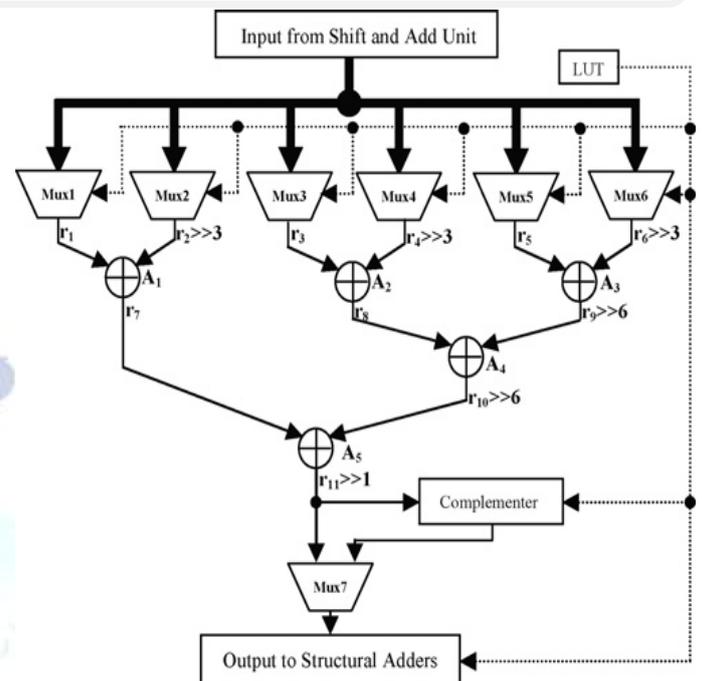


Fig. 3. Architecture of PE for CSM.

The architecture of PE for CSM is shown in Fig. 3. The coefficient word length is considered as 18 bits. The filter coefficients are stored in the LUT in sign-magnitude form with the MSB reserved for the sign bit. The first bit after the sign bit is used to represent the integer part of the coefficient and the remaining 16 bits are used to represent the fractional part of the coefficient. Thus, each 16-bit coefficient is stored as an 18-bit value in LUTs. Each row in LUT corresponds to one coefficient. Note that only half the number of coefficients needs to be stored as FIR filter coefficients are symmetric. The coefficient values corresponding to 20 to 2-14 are partitioned into groups of three bits and are used as select signals to multiplexers Mux1 to Mux5. i.e., the set (20, 2-1, 2-2) forms the select signal to Mux1 and so on. Since there are 3-bits, eight combinations are possible and hence Mux1 to Mux5 are 8:1 multiplexers. The value corresponding to 2-15 forms the select to a 2:1 multiplexer, Mux6. The output from the i th multiplexer is denoted as r_i . Note that even though we are taking coefficient with values up to a precision of 16 bits, the shifting of 2-1 is done finally as shown in (4) and (5) and hence the maximum shift will be 2-15. Mux7 determines whether the output needs to be complemented based on the sign bit of the filter coefficient and hence it is a 2:1 multiplexer. In FIR filters, coefficient values are always less than one. Hence, an integer bit is not employed. However if an integer digit is required, the proposed architectures do not impose any restrictions to accommodate it.

In Fig. 4, the shifts are obtained as follows. Let r_1 to r_6 denotes the outputs of Mux1 to Mux6, respectively. Then

$$y = 2^{-1}r_1 + 2^{-4}r_2 + 2^{-7}r_3 + 2^{-10}r_4 + 2^{-13}r_5 + 2^{-16}r_6 \dots (6)$$

The shifts are obtained by partitioning the 16-bit coefficient into groups of 3-bits.

IV. ARCHITECTURE OF PSM

The PSM architecture presented in this section incorporates re-configurability into BCSE. The PSM has a pre-analysis part in which the filter coefficients are analyzed using the BCSE algorithm. Thus, the redundant computations (additions) are eliminated using the BCSs and the resulting coefficients in a coded format are stored in the LUT. The coding format is explained in the latter part of this section. The shift and add unit is identical for both PSM and CSM. The number of multiplexer units required can be obtained from the filter coefficients after the application of BCSE. The number of multiplexers is selected after considering the number of non-zero operands (BCSs and unpaired bits) in each of the coefficients after the application of the BCSE algorithm.

The number of multiplexers will be corresponding to the number of non-zero operands for the worst-case coefficient (worst-case coefficient being defined as coefficient that has the maximum number of non-zero operands). The architecture of PE for PSM is shown in Fig. 4. The coefficient word length is fixed as 16 bits. The statistical analysis for various filters with coefficient precision of 16 bits and different filter lengths (20, 50, 80, 120, 200, 400, and 800 taps) is done and it was found that the maximum number of non-zero operands is 5 for any coefficient. The analysis was done for filters with different pass band (ω_p) and stop band (ω_s) frequency specifications given by 1) $\omega_p = 0.1\pi, \omega_s = 0.12\pi$; 2) $\omega_p = 0.15\pi, \omega_s = 0.25\pi$; 3) $\omega_p = 0.2\pi, \omega_s = 0.22\pi$; and 4) $\omega_p = 0.2\pi, \omega_s = 0.3\pi$, respectively. Based on our statistical analysis, The number of multiplexers are fixed as 5 (same as the number of non-zero operands).

The LUT consists of two rows of 18 bits for each coefficient of the form SDDDDXXDDDDXXMMMML and DDDDXDDDDXXDDDDXX, Where “S” represents the sign bit, “DDDD” represents the shift values from 20 to 2-15 and “XX” represents the input “x” or the BCSs obtained from the shift and add unit. In the coded format, XX = “01” represents “x,” “10” represents $x+2^{-1}x$, “11”

represents $x + 2^{-2}x$, and “00” represents $x + 2^{-1}x + 2^{-2}x$, respectively.

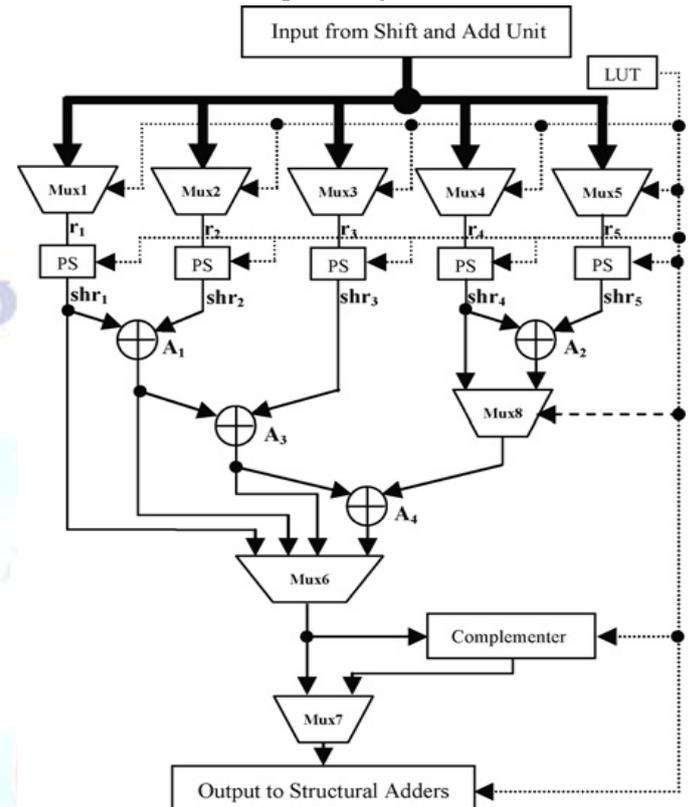


Figure. 4. Architecture of PE for PSM.

V. RESULTS

MODELSIM Software:

Mentor Graphics was the first to combine single kernel simulator (SKS) technology with a unified debug environment for Verilog, VHDL, and systemic

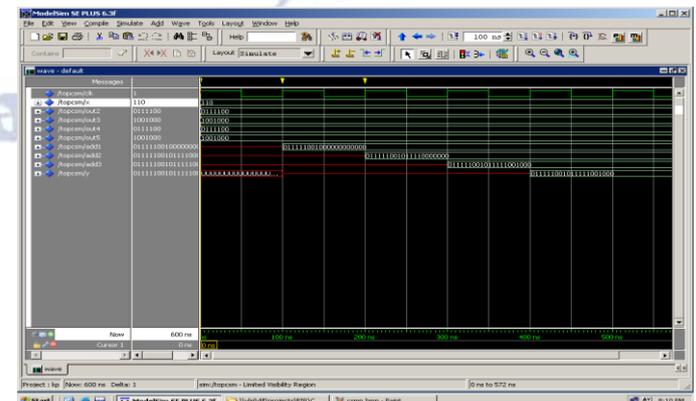
We are having many simulators in MODELSIM

1. MODELSIM SE EDITION
2. MODELSIM PE EDITION
3. MODEL SIM DE EDITION

In our project we are using MODEL SIM SE edition as a simulator.

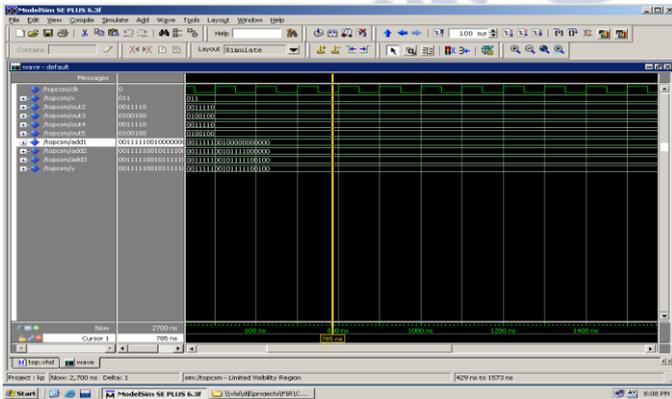
SIMULATION RESULTS

Simulation Result for CSM



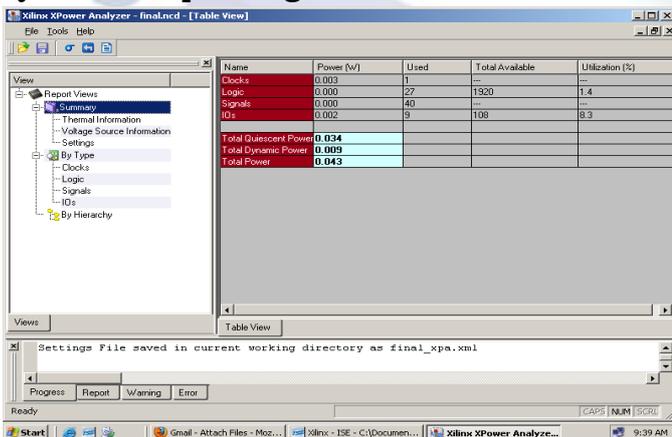
Here in the CSM technique h is taken as 18 bit filter coefficient and input $x=110$. As the multiplexers are minimized to 5 the complexity is reduced by minimum number of adders and shifting operations in order to reduce the complexity of the filter structure. After passing through several adders and several shift operations output bit will be obtained 'y' as 21 word length. Thus, the CSM architecture results in faster coefficient multiplication operation at the cost of few extra adders compared to PSM architecture

Simulation Result for PSM



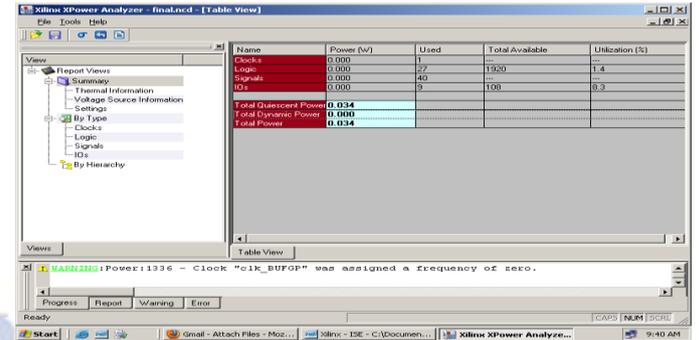
Here in the PSM technique h is taken as 18 bit filter coefficient and input $x=011$. And the output 'y' obtained as 00111100101111100100. Thus, by choosing the appropriate filter coefficient word length, it is possible to obtain reduced area and power as well as increased speed for the PSM architecture. In PSM, since the BCSE algorithm is employed, the number of additions to be performed will always be reduced compared to CSM.

Synthesis Report of general FIR Filters



Thus the above power report shows that for the implementation of general FIR Filter without CSM and PSM techniques power utilized is 0.043W.

Synthesis Report of CSM and PSM Filters



Thus the above power report shows that for the implementation of CSM and PSM based Fir Filters power utilized is 0.034W. Thus the CSM and PSM are used for implementing reconfigurable higher order filters with low complexity. The CSM architecture results in high speed filters and PSM architecture results in low area and thus low power filter implementations. The PSM also provides the flexibility of changing the filter coefficient word lengths dynamically.

VI. CONCLUSION

The two proposed new approaches namely, CSM and PSM, for implementing reconfigurable higher order filters with low complexity. The CSM architecture results in high speed filters and PSM architecture results in low area and thus low power filter implementations. The PSM also provides the flexibility of changing the filter coefficient word lengths dynamically. And implemented the architectures on Virtex-II 2v3000ff1152-4 FPGA and synthesized using 0.18 μm CMOS technology with a high coefficient precision of 16 bits and compared to numerous reconfigurable FIR filter architectures. The proposed reconfigurable architectures can be easily modified to employ any CSE (MCM) method. Thus, our method is a general approach for low complexity reconfigurable channel filters. In our project, If input length of FIR filter increases, the number of multiplexers also increases, implies loading effect increases. In order to remove loading effect, Instead of multiplexers we can use any alternating circuitry.

REFERENCES

[1] D. Saptono, V. Brost, F. Yang, and E. Prasetyo, "Design Space Exploration for a custom VLIW Architecture : Direct photo printer hardware setting using VEX Compiler," In Proc of the 4th International Conference of SITIS 2008, November 2008.
 [2] Gelato.org, "Impact Advanced Compiler Technology", <http://gelato.uiuc.edu>, 2009.

- [3] J.A. Fihser, P. Faraboschi, and C. Young, “ Embedded Computing : A VLIW Approach to Architecture, Compilers and Tools, Elsevier Morgan Kauffman, New York, 2005.
- [4] J.C Gyllenhaal, W. Mei Hwu, and B.R Rao,“HMDES Version 2.0 Specification Technical Report IMPACT-96-3, 1996.
- [5] M. Rupp, A. Burg, and E. Beck, “ Rapid prototyping for wireless design,” the five-ones approaches, Signal Processing, Vol.83, pp 1427-1444. 2003.
- [6] D. Pramanik, H. Kamberian, C. Proglar, M.Sanie, and D. Pinto, “ Cost effective strategies for asic masks”, In Cost and Performance in Integrated Circuit Creation, 2003.
- [7] Weng Fook Lee, “VLIW Microprocessor Hardware Design For ASIC and FPGA”, Mc Graw Hill Professional,2008.
- [8] V. Brost, F. Yang, M. Pandaivone, and N. Farrugia ,” Multiple modular VLIW processors based on FPGA”, Journal of Electronic Imaging, SPIE, 16(2):110, April-June 2007.

