ANIMATEDLY RECONFIGURABLE FIR FILTER DESIGN BY CONSTANT SHIFT ELIMINATION METHOD

RAMKUMAR SUNDARAM, DEBMALYA BHATTACHARYA, VIGNESH PRASANA & TATA UMAMAGESWARAN
School of Electrical Engineering, Vel Tech University, Avadi, Tamil Nadu, India

ABSTRACT

Software Defined Radio (SDR) is a recent technology evolving in the area of wireless communications. Finite Impulse Response filters play an important role in designing SDR. To achieve the flexibility dynamically reconfigurable filters are needed to be designed. In order to achieve that, this paper provides one method called constant shift elimination method based on the binary common sub expression algorithm. The goal of the constant shift elimination method is to identify multiple occurrences of identical bit patterns and eliminate the redundant multiplication. That will reduce the number of adders used to implement the coefficient multiplier used for designing dynamically reconfigurable FIR filters. Design examples show that the proposed method offers good power reductions and speed improvement compared to the best existing reconfigurable FIR filter implementations.

KEYWORDS: Software Defined Radio (SDR), Re-Configurability

INTRODUCTION

Digital filters are frequently used in digital signal processing by virtue of stability and easy implementation. Although programmable filters based on digital signal processing cores can take an advantage of Vel tech Dr. R.R. & S.R. Technical University. Flexibility, they are not suitable for recent consumer applications demanding high throughput and low-power consumption. In such an application, therefore, application specific digital filters are frequently adopted to meet the constraints of performance and power consumption.

Software radios can significantly reduce the cost and complexity of today’s cellular radio base stations. Software radios architectures centre on the use of wide band (WB) A/D converters and D/A converters as close to the antenna as possible, with as much radio functionality as possible implemented in the digital domain.

In general, cellular radio base station contains 20 or more narrowband (NB) receivers, and each of these receivers has its own set of analog mixers, local oscillators, analog filters, analog to digital converters (ADC), and baseband processing unit. The cost of today’s radio base station receivers is a linear function of the number of received channels, and this cost is dominated by the cost of the analog components. In a radio base station with a software radio receiver, a single analog front end can be used to receive all channels; hence, the cost of the analog part of the resulting receiver is a constant function of the number of received channels. The analog front end of a software radio receiver contains a wideband ADC for digitizing the entire frequency band allocated to the cellular radio base station. The output of this ADC goes to the Intermediate Frequency (IF) processing block which contains a bank of digital band pass filters for extracting individual radio channels from the output of the ADC.

Practical cellular radio base stations require that wideband (WB) receivers be efficient in their use of resources such as power, hardware cost, and computational resources. In a wideband radio receiver, the IF processing block is by far...
the most computationally demanding block; since, this block operates at the highest sampling rate. Computationally efficient algorithm for the implementation of the IF processing. Blocks are essential for wide spread deployment of software radios. Digital down-converter chips dedicated to extraction of one radio channel are currently available.

Complexity of this structure, the filter bank channelizer is a constant function of the number of received channels. Specifically, the complexity of this structure is approximately twice the complexity of implementing a digital band pass filter that extracts only one radio channel from the output of the ADC. If the radio base station conforms to a regular frequency reuse pattern, the complexity of the filter bank channelizer can be further reduced (by a factor of 2) resulting in what we call the sub sampled filter bank channelizer.

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. Reconfigurability of the receiver to work with different wireless communication standards is another key requirement in an SDR.

The complexity of the FIR filter is mainly based on the complexity of the coefficient multiplication. The Common Sub expression Elimination (CSE) methods based on Canonical Signed Digit (CSD) coefficients produce low complexity FIR filter coefficient multiplier. The goal of CSE is to identify multiple occurrences of identical bit patterns, and eliminate the redundant multiplication. In Binary Common Sub expression Elimination (BCSE) method was proposed, which provided less number of adder and low complexity FIR filter. In the pseudo floating point method was used to encode the filter coefficients and thus reduce the complexity. This method is limited to filter length less than 40. These methods are not suitable for reconfigurable filters, because the filter coefficients are fixed. Reconfigurability and low complexity are two key requirements of FIR filters employed in multi-standard wireless communication system. Two new reconfigurable architectures of low complexity FIR filters are Constant Shift (CSM) FIR filter. The proposed FIR filter architecture is capable of operating for different word length filter coefficients without any overhead in the hardware circuitry. This reconfigurable architecture filters can be efficiently implemented by using BCSE algorithm.

In CSM approaches, the filter coefficients are used as a selector line of the multiplexer to select the multiplied input value, so the complexity and area is reduced. Then the multiplexer output, the multiplied value was shifted by constant time based on the BCSE approaches and the resultant output value was summed together with proper shift. These operations were performed for all filter coefficients. This CSM approaches was applied for both 3-bit BCSE and 4-bit BCSE, also applied for various filters with coefficient precision 16 bit and different filter lengths.

The main objective of the proposed Constant Shift Method (CSM) Finite Impulse Response (FIR) filter architecture is reduction in area, power dissipation and increase the speed of operation compared to the existing reconfigurable FIR filter architecture. This architecture is applicable for SDR receiver channel filter and higher order filter.

**CONSTANT SHIFT ELIMINATION METHODS**

Various methods, utilize the CSs that occur in the CSD representation of filter coefficients. Recently, a CSE method based on Binary representation of coefficients, which produces better reduction of Logic Operators (LO), compared to the CSD based CSE methods. The reduction of LOs was obtained by combining three techniques, the 3 bit binary Horizontal CSE (HCSE), Vertical CSE (VCSE) and hardwiring of final stage adders. The HCSE method offers better reduction of adders for filters whose coefficients are coded using relatively larger word lengths (≥ 16 bits), whereas
in applications that require only shorter word lengths, the VCSE method is found to be better.

The complexity of Finite Impulse Response (FIR) filters is dominated by the number of adders (sub tractors) used to implement the coefficient multipliers. It is well known that Common Sub expression Elimination (CSE) method based on Canonic Signed Digit (CSD) representation considerably reduces the number of adders in coefficient multipliers. Recently, a binary based CSE (BSE) technique was proposed, which produced better reduction of adders compared to the CSD based CSE.

**BINARY COMMON SUBEXPRESSION ELIMINATION METHOD**

BCSE method deals with the elimination of redundant binary common sub expression that occurs within the coefficients. The BCSE technique focuses on eliminating redundant computation in coefficient multipliers using binary bit patterns present in coefficients.

**3-BIT BCSE METHOD**

3 bit binary representation can form for four BCSs, which are [0 1 1], [1 0 1], [1 1 0] and [1 1 1]. These BCSs can be expressed as follows:

\[ y_3 = [0 1 1] \times X = (2^{-1} \times X) + (2^{-2} \times X) \]
\[ = 2^{-1} (x + 2^{-1} \times X) = 2^{-1} y_6 \]
\[ Y_2 = [1 1 1] \times X = X + 2^{-1} \times X + 2^{-2} \times X \]
\[ = y_2 + y_6 \]
\[ y_5 = [1 0 1] \times X = X + 2^{-1} \times X \]
\[ Y_6 = [1 1 0] \times X = X + 2^{-1} \times X \]

The other BCSs such as [0 0 1], [0 1 0] and [1 0 0] do not require any adder for implementation as they have only one nonzero bit. In a straight forward realization of conventional method require five adders. In case BCSE method only three adders are needed. In Figure 2.1, “\(x \gg k\)” represents the input \(x\) shifted right by \(k\) units.

**PROPOSED FILTER ARCHITECTURE**

In this architectures that incorporate Reconfigurability into the BCSE-based low complexity filter architecture. Proposed architecture is based on the transposed direct form FIR filter structure as shown in Figure 1.

![Figure 1: Transposed Direct form of an FIR Filter](image-url)
The dotted portion in figure 1 represents the MB. In Figure 1, PE-i represents the processing element corresponding to the ith coefficient. PE performs the coefficient multiplication operation with the help of a shift and add unit which will be explained in the latter part of this section. In the CSM, the filter coefficients are partitioned into fixed groups and hence the PE architecture involves constant shifters. The FIR filter architecture can be realized in a serial way in which the same PE is used for generation of all partial products by convolving the coefficients with the input signal \( h * x[n] \) or in a parallel way, where parallel PE architectures are employed. The first option is used when power consumption and area are of prime concern.

**PROCESSING ELEMENT ARCHITECTURE**

![Figure 2: Architecture of PE for 4-Bit BCSE CSM](image)

The main blocks of the processing element are look up table (LUT), shift and add unit, multiplexer and adder units.

**Look up Table Unit**

In LUT the filter coefficients are stored. In the CSM, the coefficients are directly stored in LUTs without any modification.

The coefficient word length is considered as 16 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 \( 2^0 \) 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 \( \{2^0, 2^{-1}, 2^{-2}, 2^{-3}\} \) forms the select signal to Mux1 and so on. Since there are 3-bits, eight combinations are possible and hence Mux1 to Mux4 are 16:1 multiplexers. The value corresponding to \( 2^{-15} \) forms the select to a 4:1 multiplexer, Mux5. The output from the ith 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 and hence the maximum shift will be \( 2^{-15} \). Mux5 determines whether the output needs to be complemented based on the sign bit of the filter coefficient and hence it is a 4:1 multiplexer.

In FIR filters, coefficient values are always less than one. [Parks–McClellan algorithm to design filters (using “firpm” command in MATLAB)]. Hence, we have not employed the integer bit. However if an integer digit is required, the proposed architectures do not impose any restrictions to accommodate it.
Animatedly Reconfigurable FIR Filter Design by Constant Shift Elimination Method

Shift and Add Unit

In figure 1, the shifts are obtained as follows. Let \( r_1 \) to \( r_4 \) denote the outputs of Mux1 to Mux4, respectively. Then

\[
Y = 2^{-4}r_1 + 2^{-8}r_2 + 2^{-12}r_3 + 2^{-16}r_4
\]  
(1)

The shifts are obtained by partitioning the 16-bit coefficient into groups of 4-bits.

By partitioning the above equation

\[
Y = 2^{-4}(r_1 + 2^{-4}r_2) + 2^{-12}(r_3 + 2^{-4}r_4)
\]  
(2)

Substituting \( (r_1 + 2^{-4}r_2) \) and \( (r_3 + 2^{-4}r_4) \) by \( r_5 \) and \( r_6 \) respectively, we get

\[
Y = 2^{-4}r_5 + 2^{-12}r_6
\]

By substituting \( r_5 + 2^{-8}r_6 \) by \( r_5 \)

\[
Y = 2^{-4}r_5
\]  
(3)

The main advantage of the CSM architecture is that all the shifts are constants irrespective of the coefficients and hence can be hardwired resulting in high speed operation of the filter.

Multiplexer and Adder Unit

In the proposed CSM architecture, \([W/3]\)number of 8:1 multiplexers ([W/3] 8:1 multiplexers and remaining 2:1 or 4:1 multiplexers in some cases) of bit-width \((x + 2)\)are required, where \(W\) is the coefficient word length and \(x\) is the input data word length. For example, if \(W = 16\) (16-bit coefficient), the proposed 3-bit BCSs-based approach requires five 8:1 multiplexers and one 2:1 multiplexer. On the other hand, if 4-bit BCSs were used instead of 3-bit BCSs, four 16:1 multiplexers are required.

Assuming an 8:1 multiplexer is equivalent to four 2:1 multiplexers and a 16:1 multiplexer is equivalent to eight 2:1 multiplexers, then the 3-bit BCSs based PE requires 21 2:1 multiplexers and 4-bit BCSs-based PE requires thirty two 2:1 multiplexers, respectively.

Thus, the multiplexer complexity would increase when 4-bit BCSs are used. To be more precise, for each PE with 16-bit filter coefficients, the multiplexer complexity of 4-bit BCSs-based PE is increased by eleven 2:1 multiplexers when compared to 3-bit BCSs based shift and add unit. But it can be noted that the total number of adders required for 3-bit BCS-based filter with \(n\) coefficients is \(3 + 5n\) (three adders for shift and add unit and five adders for each PE) and that for 4-bit BCS-based PE is \(7 + 3n\) (seven adders for shift and unit and three adders for each PE).

Hence, two adders are saved for 4-bit BCSs based filter for each PE. From the above discussion, it can be concluded that if 4-bit BCSs were used instead of 3-bit BCSs, the complexity of shift and add unit and multiplexer unit of PE would have increased, whereas complexity of final adder unit would decrease.

RESULTS AND COMPARISON

In this section the synthesis results of the proposed CSM architectures are presented. The comparison of 3-bit BCSE approaches and 4 bit BCSE approaches also presented.
Table 1: Logic Utilization and Area Comparison

<table>
<thead>
<tr>
<th>Area</th>
<th>3-Bit BCSE CSM FIR Filter</th>
<th>4-Bit BCSE CSM FIR Filter</th>
<th>8-Bit BCSE CSM FIR Filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shift and Add Unit (Number of adders)</td>
<td>Conventional method</td>
<td>05</td>
<td>07</td>
</tr>
<tr>
<td></td>
<td>BCSE method</td>
<td>03</td>
<td>04</td>
</tr>
<tr>
<td>Multiplexer Unit (Number of MUXs)</td>
<td>16:1 MUX</td>
<td>--</td>
<td>04</td>
</tr>
<tr>
<td></td>
<td>8:1 MUX</td>
<td>05</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td>4:1 MUX</td>
<td>02</td>
<td>01</td>
</tr>
<tr>
<td>Final Adder Unit</td>
<td>Number of adders</td>
<td>05</td>
<td>03</td>
</tr>
<tr>
<td>Logic Utilization</td>
<td>2%</td>
<td>1%</td>
<td>15%</td>
</tr>
<tr>
<td>Combinational ALUTs</td>
<td>547</td>
<td>358</td>
<td>2731</td>
</tr>
</tbody>
</table>

Table 1 shows the adder requirement, multiplexer requirement and logic utilization (area) comparison for 3-bit BCSE and 4-bit BCSE CSM FIR filter. When compare with 3-bit BCSE CSM FIR filter 4-bit BCSE CSM FIR filter have low complexity and less number of adders.

Table 2: Power Dissipation Comparison

<table>
<thead>
<tr>
<th>Power Dissipation</th>
<th>3 Bit BCSE CSM Filter</th>
<th>4 Bit BCSE CSM Filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total thermal power dissipation(mW)</td>
<td>425.65</td>
<td>425.43</td>
</tr>
<tr>
<td>Core static thermal power dissipation(mW)</td>
<td>397.99</td>
<td>397.76</td>
</tr>
<tr>
<td>I/O Thermal power dissipation(mW)</td>
<td>27.67</td>
<td>27.67</td>
</tr>
</tbody>
</table>

Table 2 shows the thermal power dissipation comparison between 3-bit BCSE and 4-bit BCSE CSM FIR filter. The power dissipation is low in 4-bit BCSE CSM FIR filter when compared to 3-bit BCSE CSM FIR filter. From the timing analysis, both the filter has same delay. Hence the speed is same for both 3-bit and 4-bit BCSE CSM FIR filter.

From these result analysis 4 bit BCSE CSM FIR filter having low complexity, minimum number of adders and low power dissipation compared to the 3-bit BCSE CSM FIR filter. The time delay and speed is same for both filters. Even though the proposed architectures are reconfigurable, the usage of adders and shifters is dependent on the filter coefficient values. It can be noted that as the precision of the coefficient is made high, the area consumption is increased and the speed of operation is reduced. 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 4 bit BCSE CSM FIR filter.

CONCLUSIONS

New reconfigurable architecture of low complexity FIR filters namely, CSM for implementing reconfigurable higher order filters with low complexity. The CSM with 4-Bit BCSE architecture results in high speed filters and low area filter implementations. The CSM also provides the flexibility of changing the filter coefficient word lengths dynamically. This reconfigurable Finite Impulse Response Filter is going to be designed by using HDL 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 filter.

FUTURE ENHANCEMENT

The above new reconfigurable architecture designed for low area and high speed but did not concern in power consumption. In future going to design with low power consumption as well as more reduces the area and improving the
Animatedly Reconfigurable FIR Filter Design by Constant Shift Elimination Method

speed of the Fir filter. And also designing for 32-bit FIR filter will implementing in SDR receiver.

REFERENCES


4. R. Mahesh and A. P. Vinod, “A new common subexpression elimination algorithm for realizing low complexity higher order filters.”


AUTHOR’S DETAILS

S.Ramkumar- pursuing PhD in Annamalai University, Chidambaram. Currently involved in the research work in biosensor (osteoporosis bone cell modeling) and Working as Assistant Professor in Veltech University, AVADI, Chennai.
Tata Umamaheswara Rao completed M.tech in VLSI from Sathyabama University, Chennai. Currently working as an Assistant Professor in Veltech University, AVADI, Chennai.

N Vignesh Prasanna completed M.E in Embedded Systems from Anna University, Chennai. Currently working as an Assistant Professor in Veltech University, AVADI, Chennai.

Dr. Debmalya Bhattacharya is working as Associate Professor in Vel Tech Technical University Chennai. He has published several papers on DSP & Wireless communication in National and International Journals. His current research interests are DSP, Antenna Design & Wireless Communications.