International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 3, November 2014

Certain Investigation on High Performance Low Complexity Fir Filter Architecture Used In Advanced Multipliers

R.Chandru[1], N.Deepak[2]
Assistant Professor,
Department Of Electronics and Communication Engineering
Sri Ramakrishna Engineering College, Coimbatore, Tamilnadu,
India[1,2]

Abstract: In mobile communication systems and multimedia applications, need for efficient reconfigurable digital finite impulse response (FIR) filters has been increasing tremendously because of the advantage of less area, low cost, low power and high speed of operation. This project presents a near optimum low- complexity, reconfigurable digital FIR filter architecture based on computation sharing multipliers (CSHM), constant shift method (CSM) and modified binary-based common sub expression elimination (BCSE) method for different word-length filter coefficients. The CSHM identifies common computation steps and reuses them for different multiplications. The proposed reconfigurable FIR filter architecture reduces the adders cost and operates at high speed for low-complexity reconfigurable filtering applications such as channelization, channel equalization, matched filtering, pulse shaping, video convolution functions, signal preconditioning, and various other communication applications.

I. Introduction

With the increasing level of device integration and the growth in complexity of micro-electronic circuits, reduction of power efficiency has come to fore as a primary design goal while power efficiency has always been desirable in electronic circuits. The finite impulse response (FIR) digital filter is the fundamental element of digital signal processing (DSP) systems. The disadvantage of using digital FIR filters is that it involves lot of computations to process a signal. The implementation cost and power consumption are also high because of computational complexity. The complexity of the FIR filter is dictated by the complexity of the coefficient multipliers. The multipliers are the most expensive blocks in terms of area, delay, and power in a FIR filter structure. As shifts are less expensive in terms of hardware implementation, the design problem can be defined as the minimization of the number of addition/subtraction operations to implement the coefficient multiplications. The complexity of the multiplier block (MB) in a FIR filter is reduced, if implemented as shift-adders and sharing common sub-expressions. In order to reduce the complexity of the filter, the filter coefficients are encoded using the pseudo random floating point method; however it is limited to filter lengths less than 40. The methods are only suitable for fixed logic filters where the coefficients are fixed.

Several reconfigurable FIR filters have been proposed by researchers and are discussed in detail [1–3, 9, 13, and 14]. These architectures are appropriate only for relatively lower-order filters and not suitable for channel filters in communication receivers. The idea is to pre-compute the values such as 0*x*, 1*x*, 2*x*, 3*x*, 4*x*, 5*x*, 6*x* and 7*x*, where *x* is the input signal and then reuse these pre-computations efficiently using multiplexers.

The BCSE method proposed in [7] provides improved adder reductions leading to low complexity FIR filters. Re configurability of the FIR filters is not considered in [7] though. The concept of reconfigurable multiplier blocks (ReMB) is proposed in [3]. The ReMB generates all the coefficient products and a multiplexer selects the required ones depending on the input. The proposed reconfigurable FIR filter architecture, (processing element architecture), is shown in Fig. 1. The idea is to pre-compute the values such as 0*x*, 2*x*, 4*x*, 6*x*, 8*x*, 10*x*, 12*x*, 14*x*, where *x* is the input signal, then reuse these pre computations efficiently using multiplexers. This computation sharing multipliers (CSHM) can be used to realize efficient, low complexity FIR filter design.

II. Modified BCSE Method

This section presents the modified BCSE algorithm. This is a technique which eliminates redundant binary common sub-expressions (BCSs) that occur within the coefficients. Computational complexity is reduced by reusing the most common binary bit patterns present in the coefficients. In general, an n-bit binary number can represent $2^n - (n + 1)$ BCSs [9]. For example, a 4-bit binary representation can form 11 BCSs, which are [0 0 1 1], [0 1 0 1], [0 1 1 0], [0 1 1 1], [1 0 0 1], [1 0 1 0], [1 0 1 1], [1 1 0 0], [1 1 0 1], [1 1 1 0] and [1 1 1 1]. Note that other BCSs such as [0 0 0 1], [0 0 1 0], [0 1 0 0], and [1 0 0 0] do not require any adders for implementation. These BCSs can be expressed as

$[0\ 0\ 1\ 1] = x1 = x + 2^1x$ (1)
$[0\ 1\ 0\ 1] = x2 = x + 2^2x$ (2)
$[0\ 1\ 1\ 0] = x3 = 2^1x + 2^2x$ (3)
$[0\ 1\ 1\ 1] = x4 = x + 2^1x + 2^2x$ (4)
$[1\ 0\ 0\ 1] = x5 = x + 2^3x$ (5)
$[1\ 0\ 1\ 0] = x6 = 2^1x + 2^3x$ (6)
$[1\ 0\ 1\ 1] = x7 = x + 2^1x + 2^3x$ (7)
$[1\ 1\ 0\ 0] = x8 = 2^2x + 2^3x$ (8)
$[1\ 1\ 0\ 1] = x9 = x + 2^2x + 2^3x$ (9)
$[1\ 1\ 1\ 0] = x10 = 2^1x + 2^2x + 2^3x$ (10)
$[1\ 1\ 1\ 1] = x11 = x + 2^1x + 2^2x + 2^3x$ (11)

where x is the input signal.

A straightforward realization of above BCSs would require seventeen adders. However x3 can be obtained from x1 by a left shift operation (without using any extra adders).similarly x8 can be obtained from x1 by double left shift operation (without using any extra adders).x6 can be obtained from x2 by a left shift operation.x10 can be obtained from x4 by a left shift operation.

Binary horizontal common sub-expression elimination (BHCSE) is used in the proposed architecture. In general, an n-bit binary number can represent $2^{(n-1)} - 1$ BHCSs.4- bit binary representation can form seven BHCSs, which are [0 0 1 1], [0 1 0 1], [0 1 1 1], [1 1 1 1], [1 1 0 1], [1 0 1 1], and [1 0 0 1]. It can be noted that the common sub-expressions (CSs) [0 0 1 1], [1 1 0 0] and [0 1 1 0] can be implemented using [1 1]; the CSs [0 1 0 1] and [1 0 1 0] can be implemented using [1 0 1] and the CSs [0 1 1 1] and [1 1 1 0] can be obtained using [1 1 1] with simple shift operations.

Also, x4 can be obtained from x2 using an adder.x7 can be obtained from x5 using an adder.x9 can be obtained from x5 using an adder.x11 can be obtained from x4 using an adder. Thus only seven adders are needed to realize all the 4-bit BCSs rather than seventeen adders. Thus the binary representation based BCSE offers better reduction of adders/subtractors needed to implement the coefficient multipliers.

III. Architecture Of Reconfigurable Fir Filter

The proposed reconfigurable FIR filter is based on reconfigurable processing element architecture as shown in Fig.1. The processing element (PE) is capable of performing three computing functions: addition (including carry), multiplication and negation with two inputs and one output.

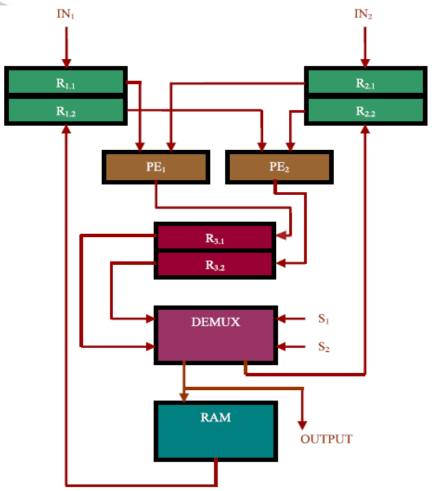


Figure1 Proposed reconfigurable FIR filter architecture module

IV. Low-Complexity Reconfigurable PE1 Architecture

The functions of different blocks of the PE1 (MB) are explained below.

A. *Shift-And-Add Unit*

One of the efficient ways to reduce complexity of multiplication operation is to realize multiplication using Shift-and-Add operations. In contrast to conventional Shift-and-Add unit used in previously proposed reconfigurable filter architectures we use modified BCSE based Shift-and-Add unit in our proposed CSM architecture. In the Shift-and-Add unit, all the even 4-bit BCSs are realized. The proposed architecture of Shift-and-Add unit is shown in Fig. 4.3. This architecture of Shift-and-Add unit is used to realize 4-bit even BCSs of input signal in the range [0 0 0 0], [0 0 1

0], [0 1 0 0], [0 1 1 0], [1 0 0 0], [1 0 1 0], [1 1 0 0], [1 1 1 0].

In the fig., “x << k” represents the input x shifted left by k units. A straightforward realization of these BCSs would require five adders. where x is the input signal.
Note that other BCSs such as [0 0 1 0], [0 1 0 0], and [1 0 0 0] do not require any adders for implementation. However, x8 can be obtained from x3 without using any adders. Also x10 can be obtained from x8.

Thus 4-bit even BCSs 0x, 2x, 4x, 6x, 8x, 10x, 12x, 14x of a 4-bit number are generated using only three adders. The easiest ways to obtain the BCSs is to know the amount of shifts in advance. All these eight BCSs (0x, 2x, 4x, 6x, 8x, 10x, 12x, 14x) are then fed to 8:1 multiplexers. In contrast to conventional Shift-and-Add unit, the number of adders/subtractors needed to implement the proposed Shift-and-Add unit is significantly reduced.

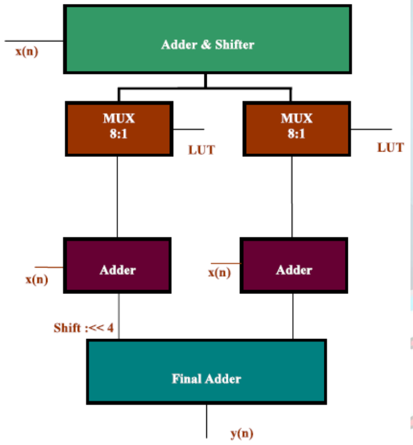


Figure 2 Architecture of the proposed CSM PE1 (MB) for 8-bit coefficients

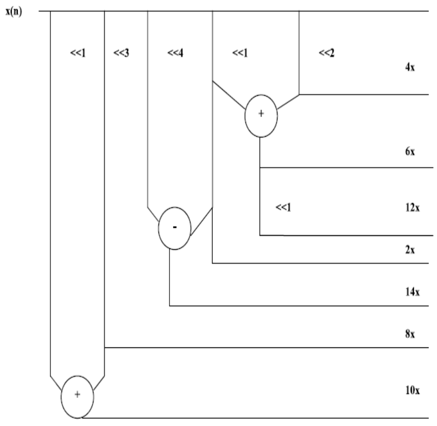


Figure 3 Architecture of Shift-and-Add unit

V. Experimental Results

In this section, the synthesis and design results of the proposed FIR filter architecture using reconfigurable MB with low complexity are presented. Altera Quartus II is used for synthesis.

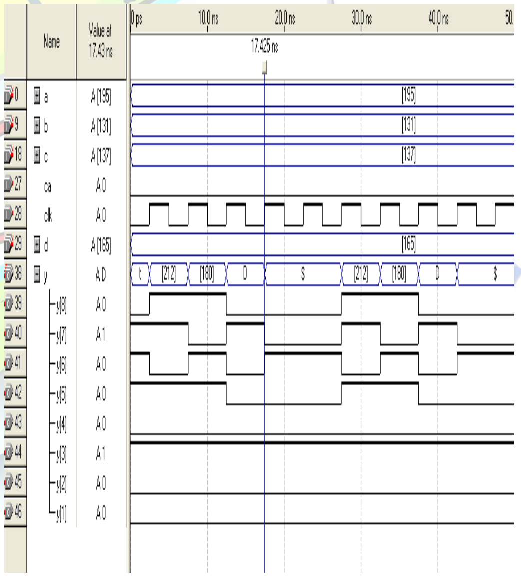
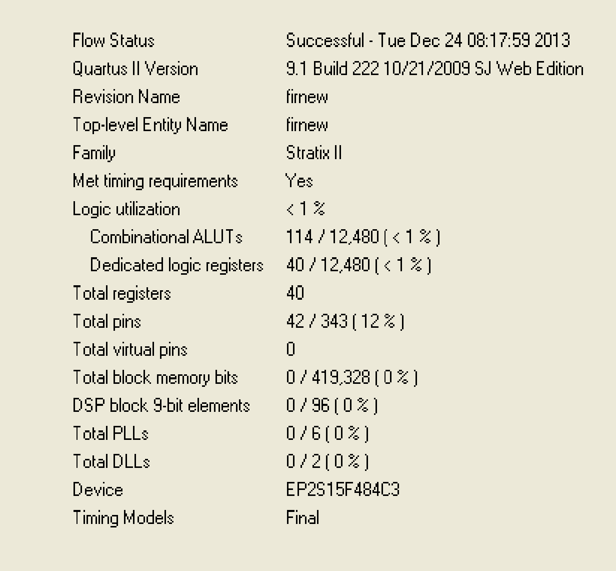


Figure 4 The output waveform of 8 Tap FIR Filter Using Wallace Tree Multiplier



```
Flow Status                     Successful - Tue Dec 24 08:17:59 2013
Quartus II Version              9.1 Build 222 10/21/2009 SJ Web Edition
Revision Name                   firnew
Top-level Entity Name           firnew
Family                          Stratix II
Met timing requirements         Yes
Logic utilization               < 1 %
    Combinational ALUTs         114 / 12,480 ( < 1 % )
    Dedicated logic registers   40 / 12,480 ( < 1 % )
Total registers                 40
Total pins                      42 / 343 ( 12 % )
Total virtual pins              0
Total block memory bits         0 / 419,328 ( 0 % )
DSP block 9-bit elements        0 / 96 ( 0 % )
Total PLLs                      0 / 6 ( 0 % )
Total DLLs                      0 / 2 ( 0 % )
Device                          EP2S15F484C3
Timing Models                   Final
```

Figure 5 The Synthesis Area Report of 8 Tap FIR Filter Using BCSE Multiplier

TABLE 4.1 Comparison of performance for the proposed implementation and the existing reconfigurable implementation.

TECHNIQUES	NO OF COMBINATIONAL ALUTS	TOTAL PINS (UTILIZATION)
USING WALLACE TREE MULTIPLIER	292	14%
USING BCSE MULTIPLIER	114	12%

The main difference of the proposed reconfigurable FIR filter architecture from the existing architectures is the use of reconfigurable FPGA based hardware accelerator with balanced shared memory architecture for FIR filtering. The proposed reconfigurable architecture uses BCS based Shift-and-Add unit and hardwired shifts. In the proposed reconfigurable CSM MB (PE1), all the shifts are constants and hence can be hardwired. This result in better speed of operation compared to other methods.

VI. Conclusion

This brief has proposed a reconfigurable FIR filter architecture using a low-complexity multiplier based on CSHM, CSM, and modified BCSE is proposed. The CSHM algorithm reduces the redundant computation in FIR filtering operation. The CSM ensures constant shifts and hence faster shifting. The modified BCSE method reduces the computational complexity by reusing the most common binary bit patterns present in the coefficients. The results show a significant reduction in either arithmetic operations or hardware necessary to implement those operations combined with satisfactory runtimes. Comparison with related work based on the available data shows that the proposed method yields comparatively better results in FIR filter optimization.

References

[1]. K.H. Chen, T.D. Chiueh, A low-power digit-based reconfigurable FIR filter. IEEE Trans. Circuits Syst. II, Express Briefs **53**(8), 617–621 (2006)

[2]. X. Chenghuan, C. He, Z. Shunan,W. Hua, Design and implementation of a high-speed programmable poly phase FIR filter, in Proc. 5th Int. Conf. Applicat.-Specific Integr. Circuit, vol. 2 (2003), pp. 783–787

[3]. S.S. Demirsoy, I. Kale, A.G. Dempster, Efficient implementation of digital filters using novel reconfigurable multiplier blocks, in Proceedings of Conference on Signals, Systems and Computers, vol. 1 (2004), pp. 461–464

[4]. R.I. Hartley, Sub-expression sharing in filters using canonic signed digit multipliers. IEEE Trans. Circuits Syst. II, Express Briefs **43**(10), 677–688 (1996)

[5]. W. Jeong, K. Roy, C. Koh, High-performance low-power carry-select adder using dual transition skewed logic, in Proc. ESSCIRC (2001), pp. 172–175

[6]. J.-J. Lee, S.G. Yong, Implementation of a bit-level super-systolic FIR filter, in IEEE Asia-Pacific Conference on Advanced System Integrated Circuits (AP-ASIC2004)I, Aug. 4–5, 2004 (2004), pp. 206–209

[7]. R. Mahesh, A.P. Vinod, A new common sub-expression elimination algorithm for implementing low complexity FIR filters in software defined radio receivers, in Proc. IEEE Int Symp. on Circuits and Systems, Greece (2006), pp. 4515–4518 682 Circuits Syst Signal Process (2013) 32:663–682

[8]. R. Mahesh, A.P. Vinod, A new common sub-expression elimination algorithm for realizing low complexity higher order digital filters. IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **27**(2), 217– 219 (2008)

[9]. R. Mahesh, A.P. Vinod, New reconfigurable architectures for implementing FIR filters with low complexity. IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **29**(2), 275–288 (2010)

About The Authors

Mr. R. Chandru is the Assistant Professor in the department of Electronics and Communication Engineering. He is having teaching experience of more than 3 years. He did his under graduate in Electronics and Communication Engineering from Sri Ramakrishna Institute of Technology, Coimbatore and his post graduate in VLSI Design from Anna University Academic campus Coimbatore, Anna University Chennai. His area of interest is Analog Electronics and Mixed signal VLSI.

Mr. N. Deepak is the Assistant Professor in the department of electronics and Communication Engineering. He is having teaching experience of more than 2 years and an Industry experience of 2 years. He did his under graduate in Electronics and Communication Engineering from Sapthagiri College of Engineering, Bangalore and his post graduate in VLSI Design from SRM University, Chennai. Currently he is pursuing his Ph.D, under the Anna University, Chennai in the area of Low Power VLSI Design. His area of interest is Low Power VLSI, Embedded Systems, and MEMS.