An Competent Multiplier Architecture Design for Fir Filter

Siva. N[1],
[1] M.E., Applied Electronics, ECE,
Selvam College of Technology,
Namakkal, Tamil Nadu, India.

Manoharan. A[2]
[2] Assistant Professor, ECE,
Selvam College of Technology,
Namakkal, Tamil Nadu, India.

***Abstract* – This paper presents design of competent hardware architecture for fixed point FIR filter has been considered. In FIR filter, the multiplication operation is performed between one particular variable and many constants and known as the multiple constant multiplications (MCM). The algorithms proposed earlier to implement this MCM for an efficient FIR filter design can be categorized in two main groups:1) graph based algorithms and 2) common sub-expression elimination algorithms (CSE). A CSE algorithm using binary representation of coefficients for the implementation of higher order FIR filter with a fewer number of adders than Canonic Signed Digit (CSD)-based CSE methods is used In this system we used vertical-horizontal binary common sub-expression elimination (VHBCSE) algorithm states that to manipulate the 16 bit input by layered operation. A 2-bit vertical BCSE has been applied first on the adjacent coefficient, followed by 4-bit and 8-bit horizontal binary common sub expressions elimination to detect and eliminate which are present within each of the coefficient. This technique is capable of reducing the average probability of use or the switching activity of the multiplier. Thus there will be a power consumption by minimum switching activity. The partial products generated by VHBCSE method and controlled additions are used by using any efficient adder to produce output efficiently. Further elimination of the common sub expressions has been performed through finding the common sub expressions present within the coefficients by applying BCSE algorithm of different lengths horizontally to different layers of the shift and add based constant multiplier architecture reconfigurable fir filter designed using VHBCSE algorithm based constant multiplier establishes the suitability of the proposed algorithm for efficient fixed point reconfigurable FIR filter synthesis .**

Key Words: Common Sub Expression sharing, MCM, FIR Filter, Shift and add technique.

I. INTRODUCTION

Multipliers are key components of many high performance systems such as FIR filters, microprocessors, digital signal processors, etc A system's performance is generally determined by the performance of the multiplier because the multiplier is generally the slowest element in the system. FIR filter has wide application as the key component in any digital signal processing, image and video processing, wireless communication, and biomedical signal processing systems. Moreover, systems like Software Defined Radio (SDR) and multi-standard video codec need a reconfigurable FIR filter with dynamically programmable filter coefficients, interpolation factors and lengths which may vary according to the specification of different standards in a portable computing platform. Significant applicability of an efficient reconfigurable FIR filter motivates the system designer to develop the chip with low cost, power, and area along with the capability to operate at very high speed Binary common sub-expression elimination (BCSE) algorithm is one of those techniques, which introduces the concept of eliminating the common sub-expression in binary form for designing an efficient constant multiplier, and is thus applicable for reconfigurable FIR filters with low complexity. However, the choice of the length of the binary common sub-expressions (BCSs) in makes the design inefficient by increasing the adder step and the hardware cost. The efficiency in terms of speed, power, and area of the constant multiplier has been increased by VHBCSE algorithm.

II. EXISTING SYSTEM

The existing system based on 2 and 3 bit BCSE algorithm that expresses multiplying the input (X) and the coefficient (H) partial products are generated and each bit is allowed through 4:1 multiplexers and addition shift operation takes place. The several adders are used to add the generated multiplexers output. At final a 2:1 multiplexer used to generates the 16 bit output. Choice of the BCS of fixed length (3-bit or 2-bit) in the earlier proposed BCSE algorithm based reconfigurable FIR filter designs leaves a scope to optimize the designed filter by considering the BCS across the adjacent coefficients as well as within a single coefficient. 2-bit BCSE algorithm is a method of assigning the values that existing in the sequence by leaving the upcoming two bits of each bit. 3-bit BCSE algorithm is a method of assigning the values that existing in the sequence by leaving the upcoming three bits of each bit. The convention considered for representing the input and the coefficient of the earlier designed FIR filter has signed magnitude format also gives a scope to modify the data representation to signed decimal number for wider applicability of the proposed FIR filter in any systems. On studying the above-mentioned literatures, it has been realized that the development of an efficient reconfigurable constant multiplier is very much needed for its applicability in any reconfigurable system.

III. LITERATURE REVIEW

A. Memory-Based Realization of FIR Digital Filter by Look-Up-Table Optimization.

Finite impulse response (FIR) digital filter is widely used in signal processing and image processing applications. Distributed arithmetic (DA)-based computation is popular for its potential for efficient memory-based implementation of finite impulse response (FIR) filter where the filter outputs are computed as inner-product of input-sample vectors and filter-coefficient vector. In this paper, however ,we show that the look-up-table(LUT)-multiplier-based approach, where the memory elements store all the possible values of products of the filter coefficients could be an area-efficient alternative to DA-based design of FIR filter with the same throughput of implementation. Finite-impulse-response (FIR) filters are basic processing elements in applications such as video signal processing and audio signal processing. The order of an FIR filter primarily determines the width of the transition-band, such that the higher the filter order, the sharper is the transition between a pass-band and adjacent stop-band. Many applications in digital communication

B. Design and Analysis of Multiplier less Finite Impulse Response Filter.

It is well known that if the individual multiplier coefficients in a digital filter can be expressed as a sum of powers-of-two, then the digital filter can be implemented in hardware without any need for actual digital multipliers. Since the multiplier is the circuit module occupying the largest silicon area, and is also the slowest, filters without multipliers are not only economical in silicon area but also fast. The improvements in speed and savings in silicon area are, however, achieved at the expense of deterioration in the frequency response characteristics. The extent to which the frequency response deteriorates depends on the number of power -of-two terms used in approximating each coefficient value, the architecture of the filter, and the discrete space optimization technique used to derive the coefficient values. There are several methods available for the optimization of FIR filters with powers-of-two multiplier coefficients by using local search algorithms. Among these methods, the Mixed Integer Linear Programming (MILP) optimization technique has become extremely popular. This optimization technique guarantees global optimality in the Min/Max sense, but is limited in scope to the design of linear phase FIR filters with the number of multiplier coefficients less than about 40. But the local search techniques have been found to perform.

C. A Low Complexity Reconfigurable Non-uniform Filter Bank for Channelization in Multi-standard Wireless Communication Receivers.

In a typical multi-standard wireless communication receiver, the channelizer must have the capability of extracting multiple channels (frequency bands) of distinct bandwidths corresponding to different communication standards. The channelizer operates at the highest sampling rate in the digital front end of receiver and hence power efficient low complex architecture is required for cost effective implementation of channelizer. Re configurability is another key requirement in the channelizer to support different communication standards. In this paper, we propose a low complexity reconfigurable filter bank (FB) channelizer based on coefficient decimation, interpolation and frequency masking techniques. The proposed FB architecture is capable of extracting channels of distinct (non-uniform) bandwidths from the wideband input signal. Design example shows that the proposed FB offers multiplier complexity reduction of 83% over Per-Channel (PC) approach and 60% over Modulated Perfect Reconstruction.

IV. PROPOSED SYSTEM

Vertical and horizontal BCSEs are the two types of BCSE used for eliminating the BCSs present across the adjacent coefficients and within the coefficients respectively in any BCSE method. Vertical BCSE produces more effective BCS elimination than the horizontal BCSE. However, this paper proposes one new BCSE algorithm which is a combination of vertical and horizontal BCSE for designing an efficient reconfigurable FIR filter. By using this proposed algorithm number of multiplexer used will be less. multiplier switching activities get reduced in our proposed algorithm, a 2-bit vertical BCSE has been applied first on the adjacent coefficient, followed by 4-bit and 8-bit horizontal BCSEs to detect and eliminate as many BCSs as possible which are present within each of the coefficient.

V. VHBCSE ALGORITHM

A. Multiplier

A binary multiplier is an electronic circuit used in digital electronics, such as a computer, to multiply two binary numbers. It is built using binary adders. A variety of computer arithmetic techniques can be used to implement a digital multiplier. Most techniques involve computing a set of partial products, and then summing the partial products together. This process is similar to the method taught to primary schoolchildren for conducting long multiplication on base-10 integers, but has been modified here for application to a base-2 (binary) numeral system. Older multiplier architectures employed a shifter and accumulator to sum each partial product, often one partial product per cycle, trading off speed for die area. Modern multiplier architectures use the Baugh–Wooley algorithm, Wallace trees, or Dadda multipliers to add the partial products together in a single cycle. Here this multiplier architecture uses VHBCSE algorithm. The performance of the Wallace tree implementation is sometimes improved by modified Booth encoding one of the two multiplicands, which reduces the number of partial products that must be summed.

B. Data flow diagram

Thedata flow diagram of the proposed vertical horizontal BCSE algorithm based constant multiplier (CM) design is

shown in Fig .1. The designed multiplier considers the length of the input (Xin) and coefficient (H) as 16-bit and 17-bit respectively while the output is assumed to be 16-bit long.

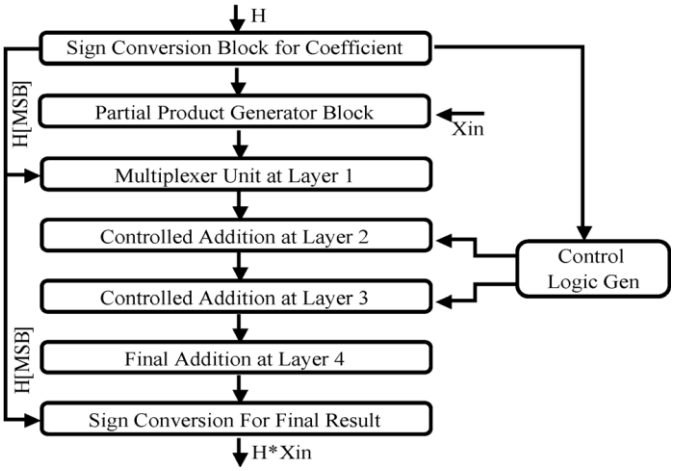

Fig.1. Data flow diagram of multiplier using VHBCSE algorithm

Herein, the sampled inputs are stored in the register first and then the coefficients are stored directly in the LUT. Functionality along with hardware architecture of different blocks of the designed VHBCSE based multipliers are explained below in details.

1) Sign Conversion Block: Sign conversion block is needed to support the signed decimal format data representation for both the input and the coefficient. The architecture of the sign conversion block is shown in Fig. 2. There is one 1's complement circuit to generate the inverted version of the 16-bit (excluding MSB) coefficient. One 16-bit 2:1 multiplexer produces the multiplexed coefficients depending on the value of the most significant bit (MSB) of the coefficient. For negative value of the original coefficient, the multiplexed coefficient will be in the inverted form; otherwise it will be as it is.

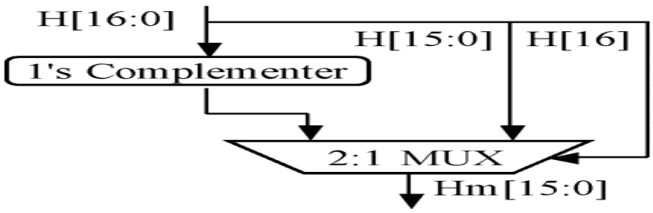

Fig.2. Architecture of the Sign Conversion Block

2) Multiplexers Unit: The multiplexer unit is used to select the appropriate data generated from the PPG unit depending on the coefficient's binary value. At layer-1, eight 4:1 multiplexers are required to produce the partial products according to the 2-bit BCSE algorithm applied vertically on the Multiplier Adder Tree (MAT). The widths of these 8 multiplexers are 17, 15, 13, 11, 9, 7, 5, and 3-bit each instead of 16-bit for all, which would reduce the hardware and power consumption.

3) Control Logic (CL) Generator: Control logic generator block takes the multiplexed coefficient (Hm[15:0]) as its input and groups it into one of 4-bit each (Hm[15:12], Hm[11:8], Hm[7:4], and Hm[3:0]) and another of 8-bit each (Hm [15:8], Hm[7:0]).The CL generator block will produce 7 control signals depending on the equality check for 7 different cases. The architecture for the control signal generator block is shown in Fig. 3. The control signal for 8-bit equality check is seen to be produced through the control signals generated from the 4-bit equality check.

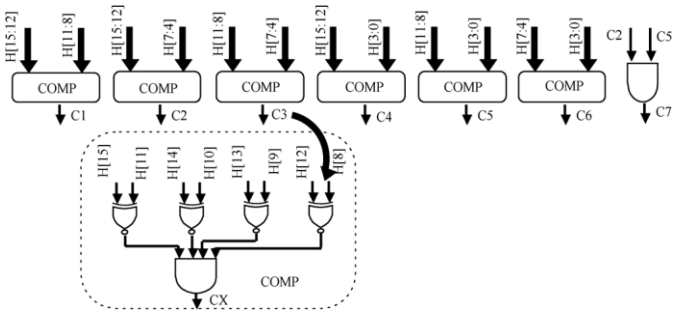

Fig.3. Control logic generator unit.

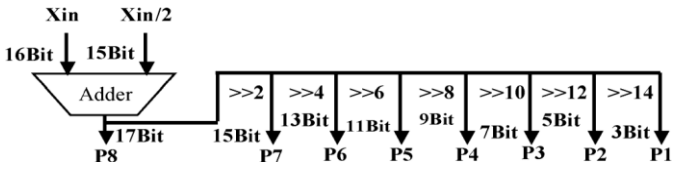

Fig.4. (Module 1) Partial product generator.

4) Partial Product Generator (PPG): In BCSE method, shift and add based technique has been used to generate the partial product which will be summed up in the following steps/layers for producing the final multiplication result. Choice of the size of the BCS defines the number of partial products. In the proposed algorithm in the layer-1, 2-bit binary common sub-expressions (BCSs) ranging from "00" to "11" have been considered, which will produce 4 partial products. But, within four of these BCSs, a single adder (A0) will be required to generate the partial product only for the pattern "11"; the rest will be generated by hardwired shifting. For the coefficient of 16-bit length,8 partial products of 17, 15, 13, 11, 9, 7, 5, and 3 bits (P8-P1) will be generated by right shifting the first partial product (P8) by 0, 2, 4, 6, 8, 10, 12, and 14 bits respectively. This technique helps in reducing the multiplexer's size which is used next to select the proper partial product depending on the coefficient's binary value.

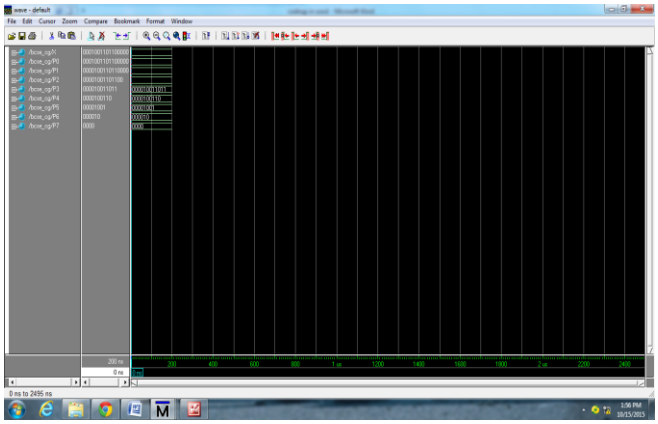

Fig.5. Output of the module partial product generator

Here, for the coefficient of 16-bit length,8 partial products of 17, 15, 13, 11, 9, 7, 5, and 3 bits (P8-P1) is generated by right shifting the first partial product (P8) by 0, 2, 4, 6, 8, 10, 12, and 14 bits.

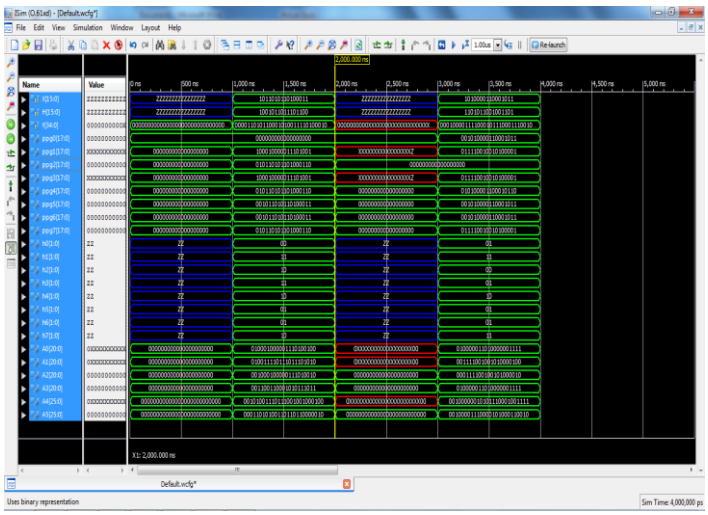

Fig.6. output of the multiplier architecture

The output waveform of the multiplier architecture for FIR filter is obtained by using XILINX ISE tool with the help of VERILOG hardware description language.

VI. CONCLUSION:

A view to implementing an efficient FIR filter, new vertical-horizontal BCSE algorithm is used, which removes the initial common sub-expressions(CSs) by applying 2-bit BCSE vertically. Further elimination of the CSs has been performed through finding the CSs present within the coefficients by applying BCSEs of different lengths horizontally to different layers of the shift and add based constant multiplier architecture. By using this algorithm, there will be maximization in efficiency of the multiplier.

REFERENCES:

[1] S. J. Darak, S. K. P. Gopi, V. A. Prasad, and E. Lai, "Low-complexity reconfigurable fast filter bank for multi-standard wireless receivers," *IEEE Trans*, vol. 22, no. 5, pp.1202–1206, May 2014.

[2] C.Y. Yao, W. C. Hsia, and Y. H. Ho, "Designing hardware-efficient fixed-point FIR filters in an expanding sub expression space," *IEEE Trans*, vol. 61, no. 1, pp. 202–212, Jan. 2014.

[3] B. Rashidi, "High performance and low-power finite impulse response filter based on ring topology with modified retiming serial multiplier on FPGA," *IET Signal Process.*, vol. 7, no. 8, pp. 743–753, Oct. 2013.

[4] P. K. Meher, "New approach to look-up-table design and memory based realization of FIR digital filter," *IEEE Trans*, vol. 57, no. 3, pp. 592–603, Mar. 2010.

[5] M. Aktan, A. Yurdakul and G. Dundar, "An algorithm for the design of low- power hardware-efficient FIR filters," IEEE Trans. Circuits Syst., Reg. Papers, vol. 55, pp. 1536–1545, Jul. 2010.

[6] J. H. Choi, N. Banerjee and K. Roy, "Variation-aware low-power synthesis methodology for fixed point FIR filters," IEEE Trans. Comput.Aided Design Integr. Circuits Syst., vol. 28, no. 1, pp. 87–97, Jan.2009.

[7] J. L. Nunez-Yanez, T. Spiteri, and G. Vafiadis, "Multi-standard reconfigurable motion estimation processor for hybrid video codecs," IET Comput. Digit. Tech., vol. 5, no. 2, pp. 73–85, Mar. 2011.