

Efficient VLSI Design Of constant Multiplier Architecture Based on Vertical-Horizontal Binary Common Sub-Expression Elimination Algorithm For reconfigurable FIR Filter Synthesis

¹M. Santhi and ²R. Vaishnavi

¹Head of the Department, Department of ECE, Saranathan College of Engineering, Trichy, India
²PG Scholar, Department of ECE, Saranathan College of Engineering, Trichy, India

Abstract: In Finite Impulse Response (FIR) filter, the multiplication operation is performed between a particular input variable and many coefficient constants. This is known as the Multiple Constant Multiplication (MCM) concept. The prior algorithms to implement this MCM for an efficient FIR filter are divided into two groups: Graphical algorithms and Common Sub-expression Elimination (CSE) algorithms. Graph based algorithms didn't consider about the delay which is the most important parameter in high performance systems. Canonic Signed Digit (CSD) based CSE methods requires more memory to store the signed bits. Both graph based and CSE algorithms run on a fixed set of coefficients which is not suitable for the application like SDR system. CSE algorithm using binary representation of coefficients known as Binary Common Subexpression Elimination (BCSE) for the implementation of higher order reconfigurable FIR filters resulted in less complexity when compared to CSD based CSE. In BCSE, number of unpaired bits is less when compared to CSD based CSE. These 2-bit and 3-bit BCSE based multiplier architectures lack optimization of adder tree. Vertical-Horizontal Binary Common Sub-expression Elimination (VHBCSE) algorithm has been applied to a multiplier of reconfigurable FIR filter whose coefficients change in real time. The proposed VHBCSE algorithm incorporates 2-bit BCSE algorithm being applied vertically across adjacent coefficients on the coefficient matrix initially, followed by applying variable-bit BCSE (4-bit and 8-bit) algorithm horizontally within each coefficient. Experimental results shows efficient power and area consumption.

Key words: Binary Common Sub-expression Elimination (BCSE) • Canonic Signed Digit (CSD) • Vertical Horizontal Binary Common Sub-expression Elimination (VHBCSE) • Control Logic Generator (CL) • Partial Product Generator (PPG)

INTRODUCTION

The multiplier is the major constraint in FIR filter which determines the performance of the desired filter. The multiplication operation complexity depends on number of adders, delay due to adders and computation cost. Most digital signal processing methods use nonlinear functions such as Discrete Cosine Transform (DCT) or Discrete Wavelet Transform (DWT). They are basically accomplished by repetitive application of multiplication and addition. So speed becomes a major factor which determines the performance of the entire calculation. Since the adder in multiplier requires the

longest delay among the basic operational blocks in digital system, the critical path is determined more by the multiplier. Furthermore, multiplier consumes much area and dissipates more power because of switching activities of adder. Adders play a key role in arithmetic and logic units (ALUs) [1]. The speed and power consumption of adders highly influence the delay and power consumption of processors. Multistandard systems like Software defined Radio (SDR), multistandard video codec are in need of reconfigurable FIR filter incorporating dynamically programmable filter coefficients, interpolation factors and lengths which of these depends on the specification of different standards. Hence the multiplier

should be designed in such a way that it should offer high speed, low power consumption, less area. Binary common sub-expression elimination (BCSE) algorithm is one of the techniques that eliminates the common sub-expressions in binary form for designing an efficient multiplier for reconfigurable FIR filter. The adder step and hardware cost of a multiplier depends on the choice of length of binary common sub-expressions (BCSs). 2-bit BCSE [1] resulted in less number of adder steps when compared to 3-bit BCSE. The BCSs can be considered across the adjacent coefficients as well as within a single coefficient. These 2-bit and 3-bit BCSE based multiplier architectures lack optimization of adder tree. An efficient multiplier in terms of area and power has been increased by VHBCSE algorithm.

The multiplications are implemented by series of shift and addition/subtraction operations. This realization of multiplication operation reduces the area. The shifts can be realized by using hard wired shifters and hence they are free. Registers are used to store the intermediate results which are partial products. Canonic Signed Digit(CSD) multiplier deals with each coefficient individually. If the multiplier is represented in CSD format, then the numbers of additions/subtractions are minimized. The advantage of CSD format is no value has more than $(N+1)/2$ nonzero bits. The number of adders/subtractors required to realize a CSD multiplier is equal to the number of nonzero digits in CSD code minus one. On the average, the CSD representation can reduce 33% of nonzero digits compared with the binary representation. A graphical based algorithm doesn't deal with each coefficient individually. Instead, all coefficients are considered as a whole. The hardware block called the multiplier block in FIR filter structure is used to implement the coefficient multiplications. Bull Horrocks (BH) [2] and Reduced Adder Graph (RAG-n) [3] algorithms didn't consider about delay which is caused due to more number of adder steps in generating partial products in serial manner. CRA [4] achieves optimization through an admissibility graph for each coefficient. An approach in [5] says when multiplication is implemented using shifts and adds, the adder width can be minimized by limiting the shifts of the operands to shorter lengths. Three methods are combined namely the Coefficient Partitioning (CP) algorithm, an efficient coefficient coding scheme known as Pseudo Floating-Point (PFP) representation and CSE to minimize the number of full adders (FAs). The "partitioned

and scaled" versions of the PFP coefficients obtained can be added using fewer numbers of FAs since their ranges are reduced.

An approach in deals with two types of reconfigurable architectures are proposed for multistandard wireless communication reconfigurability and low complexity. Constant Shifts Method (CSM) for fixed coefficients and Programmable Shifts Method (PSM) for reconfigurable coefficients. CSM is used for application specific filters whereas PSM is used for reconfigurable filters. CSM doesn't guarantee minimum number of additions since coefficients are splitted into fixed groups whereas in PSM guarantees minimum number of additions since BCSE is employed. So PSM can be used for higher order filters. In CSM, shifts are constant since filter coefficients are splitted into fixed groups. So constant shifters are used and also results in faster multiplication. In PSM, final shifting is based on values from LUT. So programmable shifters are used. proposed a two-step optimization technique for designing a reconfigurable VLSI architecture of an interpolation filter for multistandard Digital Up Converter (DUC) to reduce area and power consumption. Multistandard corresponds to the three standards: universal mobile telecommunication standard, wideband cdma and digital video broadcasting. These three standards adopted Root Raised Cosine (RRC) filter as pulse shaping filter since it decreases the bit error rate by preventing timing jitter at sampling instant. 2-bit BCSE and 3-bit BCSE encounter some problems. These two architectures consider the signed magnitude number format for inputs and coefficients. But in most of the systems signed decimal format data representation is followed. So changes have to be made in these architectures to support signed decimal data representation for its wide application. BCSE algorithm is applied only in coefficient matrix vertically. The adder tree is designed to sum up the partial products that are generated from multiplication of input with coefficient. This results in high power consumption because of the high probability of switching activities of these adders. The consumption of the hardware and power increases for filter coefficients which consists of small decimal values with negative sign and also for high positive numbers.

Proposed Solutions: The filter coefficient is to be multiplexed between its original and complemented values depending on the MSB of the coefficient to support the

signed decimal data representation. This technique reduces the hardware complexity when the coefficients consist of small negative decimal numbers and high positive decimal numbers. The 2-bit BCSE is to be applied vertically across the coefficients during the selection of partial products followed by conditional 4-bit and 8-bit BCSEs horizontally in binary tree structured adder respectively to find out the common sub-expressions (CSs) present within the coefficients. This solves the additional hardware consumption problem by eliminating more CSs. Applying BCSE horizontally reduces the probability of use of switching activities of the multiplier block (MB) adders that ultimately reduces the power consumption. Using above techniques, multiplier for reconfigurable filter can be designed where filter coefficient sets, interpolation factors, lengths change dynamically according to the specifications of wireless standards. So less cost multiplier operating at high speed is obtained. Adder that adds 2 n-bit numbers require (n+1) full adders to compute the sum. Instead of using ripple carry adder which consumes more time, carry skip adder can be used which reduces the delay by minimizing the time consumption of carry propagation.

Proposed Vertical Horizontal Bcse (Vhbcse) Algorithm: The proposed VHBCSE algorithm has a data flow diagram shown in Fig.1.

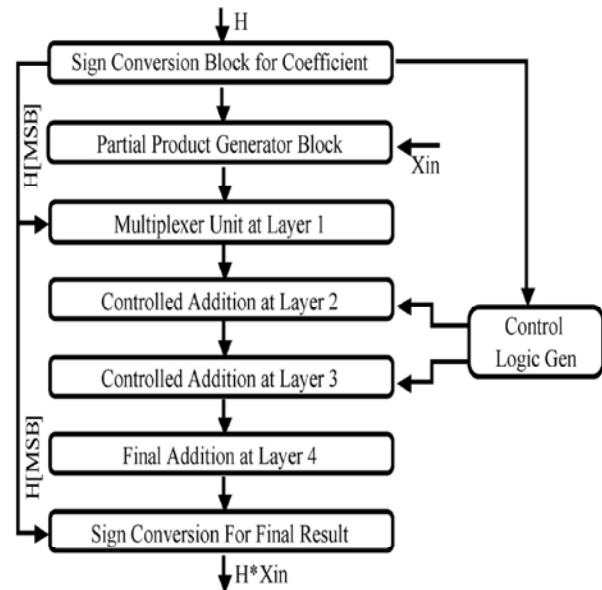


Fig. 1: Data flow diagram of the CM using VHBCSE algorithm.

Here the input (X_{in}) and the coefficient (H) are considered to be 16-bit and 17-bit respectively. The output is 16-bit in length. The sampled inputs are stored in register and the coefficients are stored in LUT.

The Functions of the Different Blocks of the Data Flow Diagram

Sign Conversion Block: Sign conversion block supports the signed decimal format data representation for both the input and the coefficient. The architecture of the signconversion block is shown in Fig. 2.

One 1's complemeter circuit generates inverted version of the 16-bit (excluding MSB) coefficient. One 16-bit 2:1 multiplexer generates the multiplexed coefficients according to the most significant bit (MSB) of the coefficient. The multiplexed coefficient will be in the inverted form for the negative coefficient. Otherwise it will be as it is.

Partial Product Generator: The shift and add based technique is used to generate the partial products and later these partial products are summed up by the following addition layers. The number of partial products depends on the choice of size of binary common subexpression (BCS). In layer-1, 2-bit BCSs are considered where extra adder is required for the patter '11' whereas the rest are generated by hardwired shifting. For 16-bit coefficient, eight partial products (P8-P1) are obtained by right shifting the first partial product P8. This reduces the

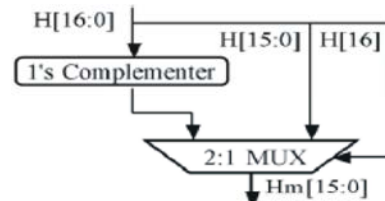


Fig. 2: Hardware architecture of the Sign Conversion Block.



Fig. 3: Block diagram of the Partial Product Generator Unit.

mux size that follows to select the proper partial product depending on the coefficient's binary value. The architecture of the block is shown in Fig.3.

Control Logic Generator (CL): CL block takes multiplexed coefficient (Hm[15:0]) as input and divides it into groups where each group is of 4-bit each (Hm[15:12], Hm[11:8],Hm[7:4],and Hm[3:0]) and another groups of 8-bit each (Hm[15:8],Hm[7:0]). CL generator produce 7 control signals for seven equality checks for 7 different cases. The architecture of the block is shown in Fig.4.

The control signal for 8-bit equality check is produced using the control signals generated from the 4-bit equality check.

Multiplexer Unit: The multiplexer unit is used select the proper partial product from PPG unit depending on the coefficient's binary value. Eight 4:1 multiplexers are required to produce the partial products in layer-1. The width of the mux depends on the width of partial products. This reduces the hardware and the power consumption.

Controlled Addition at Layer-2: The partial products (Pps) that are generated from eight groups of 2-bit BCSs are summed up for the final multiplication results which are performed in three layers. Layer-2 requires four addition (A1-A4) operations for addition of the eight PPs. The control signals (C1-C6) that were generated based on 4-bit

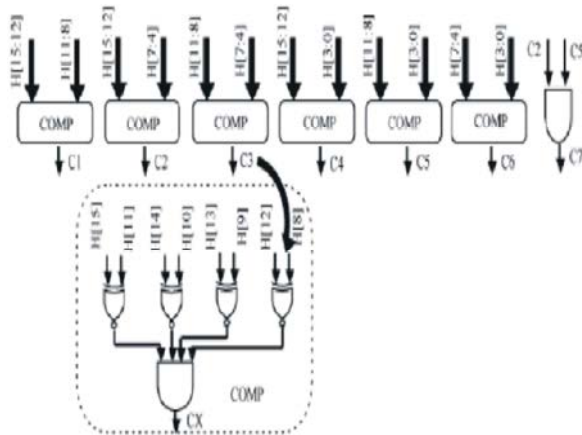


Fig. 4: Block diagram of the control logic generator unit.

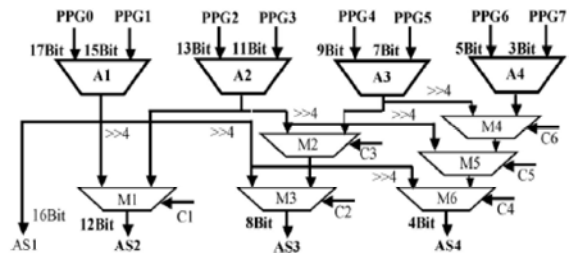


Fig. 5: Architectural details of the controlled addition at layer-2 block.

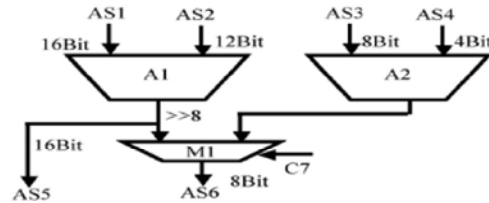


Fig. 6: Hardware architecture of the controlled addition at layer-3

BCSE from CL Block: The architecture of this block is shown in Fig.5

Controlled Addition at Layer-3: In layer-3, the four multiplexed sums (AS1, AS2, AS3 and AS4) that are generated from layer-2 are summed up in layer-3. This controlled addition A6 is controlled by the control signal (C7) that has been generated based on 8-bit BCSE from the CL generator block. The architecture of the block is shown in Fig.6.

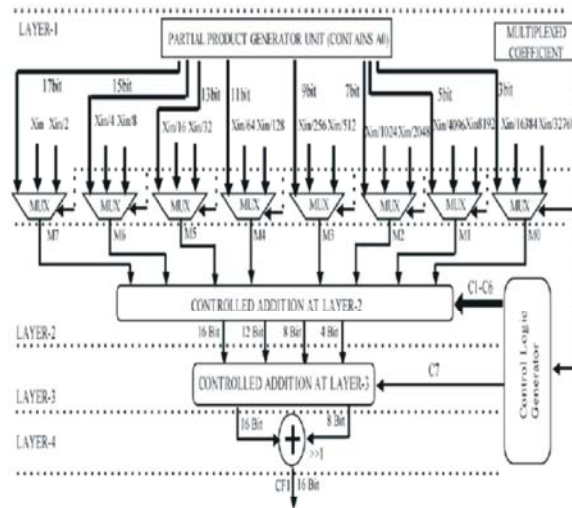


Fig. 7: Proposed Reconfigurable constant multiplier architecture

Final Addition on Layer: The sums (AS5-AS6) are added in layer-3 to produce the final multiplication result between input and coefficient. The overall architecture is shown in Fig.7.

RESULTS AND DISCUSSION

The VHBCSE algorithm based constant multiplier architecture has been coded using Verilog hardware description language and synthesized on Xilinx xc3s1400an-5fgg676FPGA device using Xilinx ISE 9.2i synthesis tool. Constant multiplier for 8-tap FIR filter has been designed using VHBCSE algorithm.

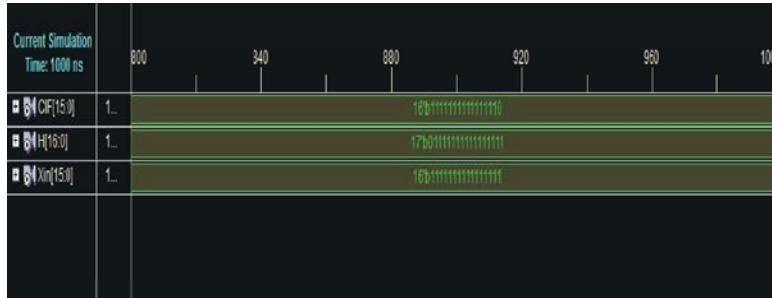


Fig. 8: Simulation result of VHBCSE multiplier

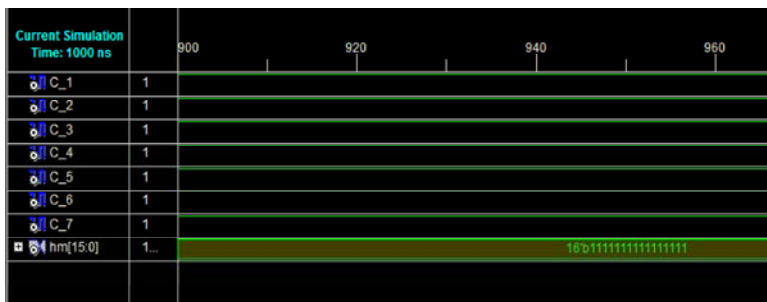


Fig. 9: Simulation result of module CL block

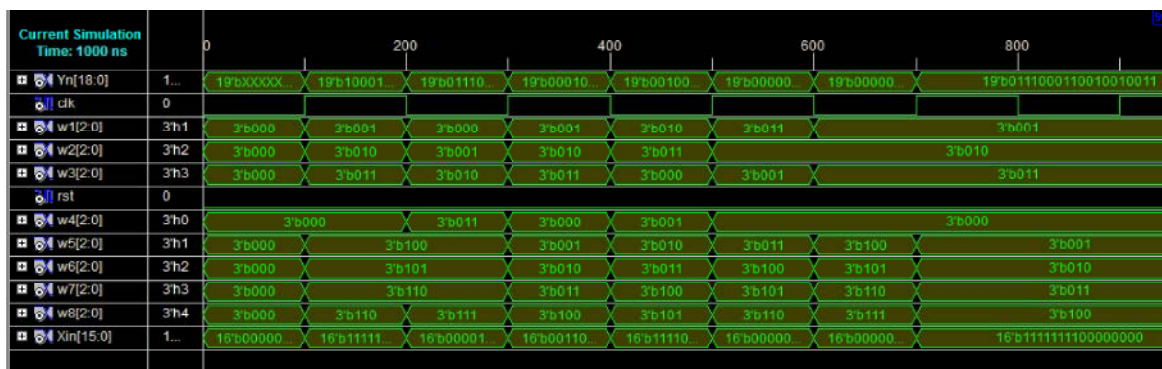


Fig. 10: Simulation result of 8-tap FIR filter designed using VHBCSE algorithm.

CONCLUSION

New VHBCSE algorithm is proposed that removes the initial common sub-expressions (CSs) using 2-bitBCSE technique. Common sub-expressions

within the coefficient BCSE technique. Common sub-expressions within the coefficient can be further eliminated by applying variable bit BCSE horizontally within the coefficients to different layers of multiplier architecture.

REFERENCES

1. Banerjee S., I. Chakrabarti and I. Hatai, 2014. An efficient VLSI architecture of a reconfigurable pulse-shaping FIR interpolation filter for multi-standard DUC, IEEE Trans. Very Large Scale Integr. (VLSI) Syst.
2. Bull, D.R. and D.H. Horrocks, 1991. Primitive operator digital filters, IEE PROCEEDINGS-G, Vol. 138, Nu. 3.
3. Chen, H.H., C.J. Chien, C.T. Hsu, T.F. Lin and C.Y. Yao, 2004. A novel common subexpression elimination method for synthesizing fixed-point FIR filters, IEEE Trans. Circuits Syst. I, Reg. Papers, 51(11): 2215-2221.
4. Ching-Chuen Jong, Chip-Hong Chang and FeiXu 2005. Contention Resolution Algorithm for Common Subexpression Elimination in Digital Filter Design, IEEE transactions on circuits and systems-2:Express briefs, vol. 52, no.10.
5. Dempster, A.G. and M.D. Macloed, 1995. Use of minimum-adder multiplier blocks in FIR digital filters, IEEE Trans.Circuits Syst.II, Analog Digit. Signal Process., 42(9): 569-577.