Implementation Report for Fir Filter Design based upon Rounded Truncated Constant Accumulation

Joshi Ashwini Vijay[1] Prof.R.S.Khule[2]
[1,2]Department of Electronics & Telecommunication Engineering
[1,2]MCERC College Nasik

Abstract— In proposed system Low-cost finite impulse response (FIR) designs are presented using the concept of faithfully rounded truncated multipliers. The optimization of bit width and hardware resources without sacrificing the frequency response and output signal accuracy. Multiple constant multiplication/accumulation in a direct FIR structure is implemented in direct form structure using a developed version of truncated multipliers which helps to reduce bit width. Dadda algorithm is adapted to reduced tree of partial product bits. Comparisons with previous FIR design approaches show that the proposed design gives more area and power efficient results.

Key words: Digital Signal Processing (DSP), Faithful Rounding, Finite Impulse Response (FIR) Filter, Truncated Multipliers, VLSI Design

I. INTRODUCTION

Finite impulse response (FIR) digital filter is very important a part of digital signal process (DSP) and communication systems. From a mathematical read, a digital filter computes the convolution of the sampled input and also the weight operates of the filter. Multiplication is one among the foremost space intense arithmetic operations in superior circuits. As a result several analysis works affect low power style of high speed multipliers. Multiplication involves 2 basic operations, the generation of the partial merchandise and their total. Multiplication of 2 bits produces associate output that is doubles that of the first bit.

Finite impulse response (FIR) digital filter is vital a part of digital signal process (DSP) and communication there are unit 2 basic FIR structures, direct kind and converse kind, as shown in Figure one for a linear-phase even-order FIR filter. Within the direct kind in Figure 1(a), the multiple constant multiplication (MCM)/accumulation (MCMA) module performs the coinciding multiplications of individual delayed signals and several filter coefficients, followed by accumulation of the entire product. Thus, the operands of the multipliers in MCMA area unit delayed signs x [n −i] and within the converse kind the operands of the multipliers within the MCM module area unit the present input signal x[n] and coefficients.

The results of individual constant multiplications undergo structure adders (SAs) and delay components. From a mathematical read, a digital filter computes the convolution of the sampled input and therefore the coefficient operates of the filter. Multiplication is one in every of the foremost space overwhelming arithmetic operations in superior circuits. As a result several analysis works contend with low power style of high speed multipliers. Multiplication involves 2 basic operations, the generation of the partial product and their add. Multiplication of 2 bits produces associate degree output that is doubly that of the initial bit.

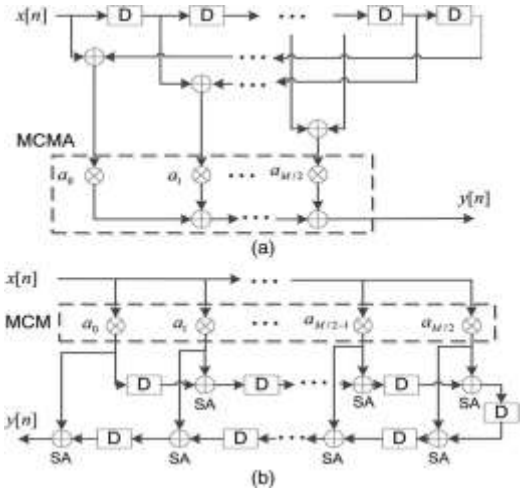

Fig. 1: Structures of linear-phase even-order FIR filters: (a) Direct form and (b) transposed form.

II. PROBLEM STATEMENT

Design of area- and power-efficient high-speed data path logic systems are one of the most considerable areas of research in VLSI system design. Multiplication is one among the foremost space overwhelming arithmetic operations in superior circuits. As a consequence several analysis works subsume low power style of high speed multipliers. Cheap FIR filter styles by conjointly considering the improvement of constant bit dimension and hardware resources in implementations.

As the Filter order will increase the amount of partial product within the multiplication will increase. A vital style issue of FIR filter implementation is that the improvements of the bit widths for filter coefficients, that have direct impact on the world value of arithmetic units and registers. Moreover, since the bit widths when multiplications grow, several DSP applications don't would like full-precision outputs. Instead, it's fascinating to come up with reliably rounded outputs wherever the full error introduced in division and misreckoning isn't any quite one unit of the last place (ulp).

III. SYSTEM ARCHITECTURE

A. System Description:

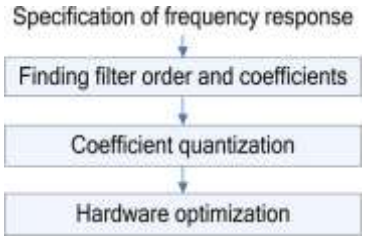

Fig. 2: Three stages in digital FIR filter design and implementation.

A standard flow of FIR filter style and implementation is divided into 3 stages: finding filter order and coefficients, constant quantization, and hardware improvement, as shown in Figure a pair of. Within the 1st stage, the filter order and also the corresponding coefficients of infinite exactness area unit determined to satisfy the specification of the frequency response victimization mat workplace intrinsically perform remez (). Here improved version of truncation number is employed to delete inessential PP bits. Finally, various optimization approaches such as Dadda algorithm is used to minimize the area cost of hardware implementations. In this brief, we adopt the direct FIR structure with MCMA because the area cost of the flip-flops in the delay elements is smaller compared with that of the transposed form.

B. PP Truncation and Compression:

The FIR filter design in this brief adopts the direct form in Figure 1(a) where the MCMA module sums up all the products$\hat{}ai \times x[n-i]$. Instead of

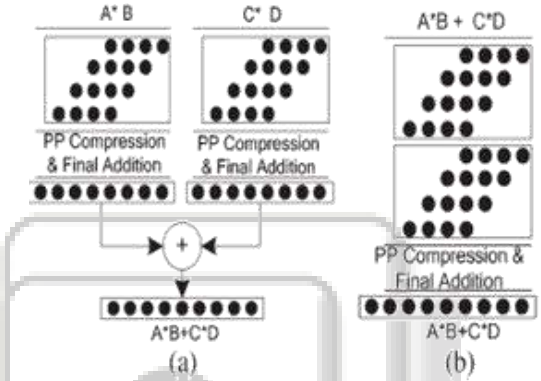

Fig. 3: Multiplication/accumulation using (a) individual PP compression and (b) combined PP compression.

C. Dadda Algorithm for Tree Reduction:

It is a hardware multiplier design invented by computer scientist Luigi Dadda in 1965. It is similar to the Wallace multiplier, but it is slightly faster (for all operand sizes) and requires fewer gates.

The reduction rules for the Dadda tree are as follows:

For three PP bits with a similar weights input them into a full adder. The result are associate output little bit of a similar weight and a carry output bit with a better weight for every 3 input wires. If there are 2 PP bits of a similar weight left, and also the current range of output bit therewith weight is adequate to two (modulo 3), input them into a [*fr1] adder. Otherwise, pass them through to ensuing layer. If there's only 1 wire left, connect it to ensuing layer.

IV. METHODOLOGY

The methodology for implementation of proposed work is thorough analysis and validation by using following software tools. The following tools are used for the designing of FIR filter:

- XILINX ISE web pack 9.1 for design, synthesis and implementation.
- MODELSIM 6.4 C for simulation.

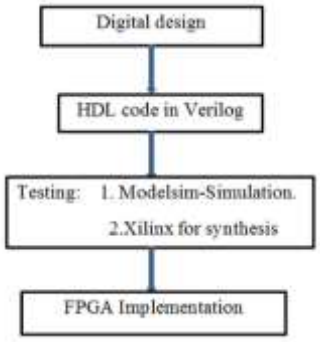

Fig. 4: Design Flow

After calculation of filter order and filter coefficients, FIR filter is realized in direct as well as transposed form. For hardware optimization the rounded truncation multiplier is used in which deletion, truncation and rounding is used to reduce no. of pp bits. Further pp its tree reduction dada algorithm is used. Design flow for proposed work is as given in figure 5.

V. FPGA IMPLEMENTATION

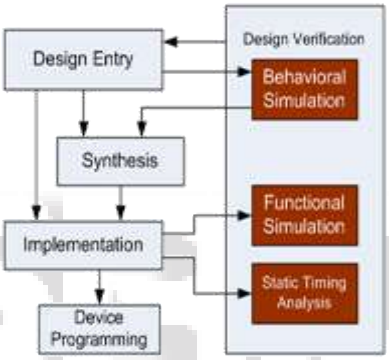

Fig. 5: FPGA design flow.

A simplified design flow for FPGA is given in the above diagram Here the HDL primarily based style entry is used. For this Verilog coding is used. Next step is synthesis which includes conversion of Verilog code into netlist. Behavioral simulation is done to check functionality of design.

Implementation process consists of a sequence of three steps

- Translate
- Map
- Place and Route

In translate step all netlist and constrains are combine into logic design file, saved as NGD file. Map process fits the logic defined by the NGD file into the targeted FPGA elements i.e. CLBs.

Running implementations with Xilinx Implementation Tools, performs timing simulation, and configures and download design to the Spartan-3 FPGA board.

VI. RESULT AND COMPARISON

After the design is successfully defined, behavioral simulation is performed, run implementation with the Xilinx Implementation Tools to perform timing simulation and power analysis. We get following results for this.

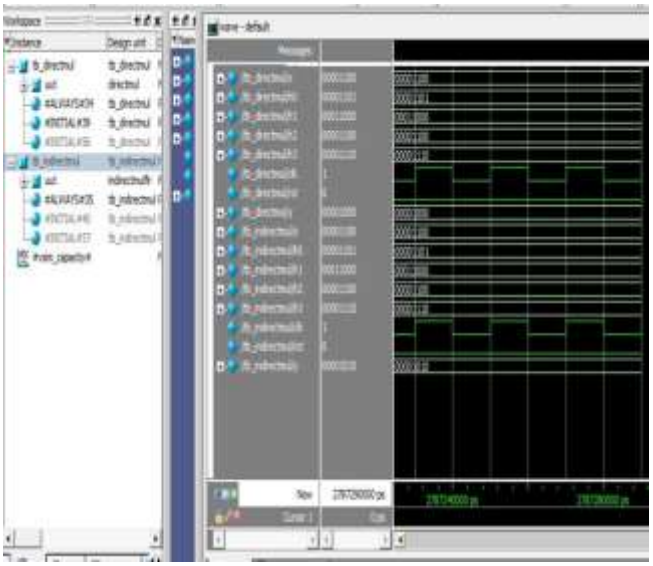

Fig. 6: Simulation Diagrams from Modelsim

A. Time, Area and Power Analysis:

Below table contain the exact values getting after doing synthesis and implementation. This is useful for comparative analysis of existing system MCAT_I and proposed system MCAT_II.

	Proposed Multiplier		Existing Multiplier
Parameter	Direct forms	Transpose forms	Transpose forms
Maximum delay	29.59	19.77	29.59
No. of flip flop's used	24	32	32
No. of LUT's used	359	391	375
Gate utilization	23%	25%	24%
POWER	25mW	25mW	26Mw

Table 1: Existing multiplier and Proposed multiplier Comparison

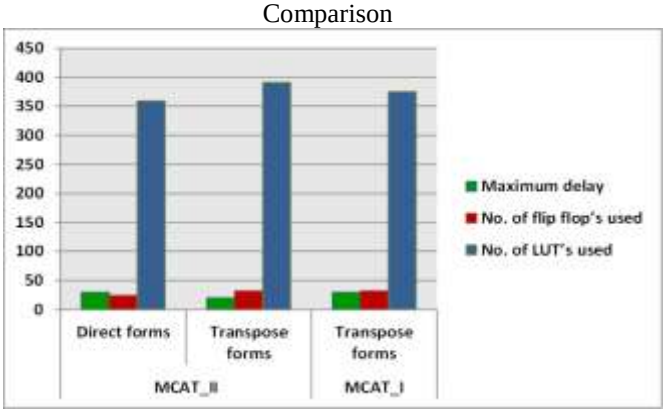

Fig. 7: comparison of the proposed design and existing design for FIR filter

From above comparison direct form requires less number of flip-flops (24) than transposed form (32), a significant increase is because the area of DFFs in the transposed forms is larger because of the range expansion of the results after MCM.Thus area reduction rates for direct form MCMAT_II are higher.

Although the area costs of the proposed designs are significantly reduced, the critical path delay is increased because all the operations in the MCMA are executed within one clock cycle.

VII. Acknowledgement

We especially thanks to our department and our Institute for the great support regarding paper and their all views. I really thankful to my all staffs and my guide Prof.R.S.Khule, who showed me the way of successful journey of publishing paper and project work.

VIII. Conclusion

This temporary has bestowed cheap FIR filter styles by together considering the optimization of constant bit breadth and hardware resources in implementations. Though most previous styles are supported the backward type, we have a tendency to observe that the direct FIR structure with dependably rounded MCMAT results in the tiniest space price and power consumption.

References

[1] M. M. Peiro, E. I. Boemo, and L. Wanhammar, "Design of high-speed multiplierless filters using a nonrecursive signed common subexpression algorithm," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process. vol. 49, no. 3, pp. 196–203, Mar. 2002.

[2] C.-H. Chang, J. Chen, and A. P. Vinod, "Information theoretic approach to complexity reduction of FIR filter design," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 8, pp. 2310–2321, Sep. 2008.

[3] F. Xu, C. H. Chang, and C. C. Jong, "Contention resolution—A new approach to versatile subexpressions sharing in multiple constant multiplications," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 2, pp. 559–571, Mar. 2008.

[4] F. Xu, C. H. Chang, and C. C. Jong, "Contention resolution algorithms for common subexpression elimination in digital filter design," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 52, no. 10, pp. 695–700, Oct. 2005.

[5] I.-C. Park and H.-J. Kang, "Digital filter synthesis based on an algorithm to generate all minimal signed digit representations," IEEE Trans.Comput.-Aided Design Integer. Circuits Syst., vol. 21, no. 12, pp. 1525–1529, Dec. 2002.

[6] C.-Y. Yao, H.-H.Chen, T.-F.Lin, C.-J.J. Chien, and X.-T. Hsu, "A novelcommon-subexpression-elimination method for synthesizing fixed-point FIR filters," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 51, no. 11,pp. 2215–2221, Sep. 2004.

[7] O. Gustafsson, "Lower bounds for constant multiplication problems, "IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 54, no. 11, pp. 974–978,Nov. 2007.

[8] Y. Voronenko and M. Puschel, "Multiplierless multiple constant multiplication," ACM Trans. Algorithms, vol. 3, no. 2, pp. 1–38, May 2007.

[9] H.-J. Ko and S.-F. Hsiao, "Design and application of faithfully rounded and truncated multipliers with combined deletion, reduction, truncation, and rounding," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 58, no. 5, pp. 304–308, May 2011.

