RESEARCH ARTICLE OPEN ACCESS

An Empirical Scheme of Different Algorithm in Fir Filter Designs Based On Faithfully Rounded Truncated MCMA

Satheesh.R[1], Rajesh Babu.G[2]
Research Scholar[1], Assistant Professor[2]
ECE Department (ME *VLSI DESIGN*),
SVS College of Engineering, and Coimbatore
Tamil Nadu - India

ABSTRACT

This paper describes the design of Finite Impulse Response (FIR) using the rounded truncated multiplier which offers diminution in area, delay, and power. FIR filter design allows hardware optimization that cannot affect the filter operation and output signal. According to this adder, registers and multipliers are the recourses to be optimized. LSB and MSB is the output form of this multiplier. Deletion, reduction, truncation, rounding and final addition are the operations performed to compress the LSB part. MCMAT (Multiple Constant Multiplication Accumulation and truncation) in direct FIR structure is proposed for FIR filter design. It utilizes recourse efficient multipliers namely truncated multipliers for performance improvement strategy. When this scheme is followed the truncation error does not exceeds 1 ulp (unit of least position). So it does not require any error compensation circuits, and the final output will be accurate.

Keywords:- Digital signal processing (DSP), faithful rounding, finite impulse response (FIR) filter, truncated multipliers, VLSI design.

I. INTRODUCTION

Linear phase finite impulse response (FIR) filters are widely used in digital signal applications such as speech coding, image processing, sampling of multiple systems, etc. Although the stability and linear phase is guaranteed, the complexity and power consumption of the linear phase FIR filter are usually much higher than that of the infinite impulse response (IIR) filter which meets the same magnitude response specifications. Multiplication is one of the most area consuming arithmetic operations in high-performance circuits. As a consequence many research works deal with low power design of high speed multipliers. Multiplication involves two basic operations, the generation of the partial products and their sum, performed using two kinds of multiplication algorithms, serial and parallel. Serial multiplication algorithms use sequential circuits with feedbacks; inner products are sequentially produced and computed. Parallel multiplication algorithms often use combinational circuits and do not contain feedback structures. Multiplication of two bits produces an output which is twice that of the original bit. It is usually needed to truncate the partial product bits to the required precision to reduce area cost.

Fixed-width multipliers, a subset of truncated multipliers, compute only n most significant bits (MSBs) of the 2n-bit product for n ×n multiplication and use extra correction/compensation circuits to reduce truncation errors. In preceding interrelated papers, to reduce the truncation error by adding error compensation circuits. So that the output will be precised. In this approach jointly considers the tree reduction, truncation, and rounding of the PP bits during the design of fast parallel truncated multipliers so that the final truncated product satisfies the precision requirement. The truncation error is not more than 1ulp (unit of least position), so error compensation circuit is not required, and the final output will be accurate.

The results of individual constant multiplications go through structure adders (SAs) and delay elements. The digital FIR filters can be divided into two categories: multiplier less based and memory based technique. Multiplierless-based designs recognize MCM with shift-and add operations and share the common sub-operations using canonical signed digit (CSD) recoding and common subexpression elimination (CSE) to minimize the added cost of MCM [1]. In [2] and [3], more area savings are achieved by jointly considering the optimization of coefficient quantization and CSE. Most multiplierless MCM-based FIR filter designs use the transposed structure to allow for cross-coefficient sharing and tend to be more rapid, if order of the filter is large. However, the area occupied delay elements is larger compared with that of the direct form due to the range expansion of the constant multiplications and the subsequent additions in the SAs. Memory-based FIR designs consist of two types of approaches: lookup table (LUT) methods and distributed arithmetic (DA) methods [4]–[6]. The LUT-based design stores in ROMs odd multiples of the input signal to realize the constant multiplications in MCM [4]. The DA-based approaches recursively accumulate the bit-level partial results for the inner product computation in FIR filtering [5], [6]. An important design issue of FIR filter

`

implementation is the optimization of the bit widths for filter coefficients, which has direct impact on the area cost of arithmetic units and registers. Moreover, since the bit widths after multiplications grow, many DSP applications do not need full-precision outputs. Instead, it is desirable to generate faithfully rounded outputs where the total error introduced in quantization and rounding is no more than one unit of the last place (ulp) defined as the weighting of the least significant bit (LSB) of the outputs. In this brief, we present different types of algorithms are discussed and also area, delay and power etc can be analyzed.

II. REVIEW OF MULTIPLIERLESS ALGORITHMS

A. Unsigned Fixed-Width Multipliers

The multiplier is based on multiple-input error-compensation architecture like[7]. It is worth noting that in only signed multipliers are considered. Moreover, the implementation of the error-compensation function is based on ripple architecture, like [8], with poor delay and power performances.

A novel error-compensation function is developed, that can be optimized in order to minimize either the maximum absolute error or the mean-square error. The dual tree error compensation function gives better accuracy. This error-compensation function can be implemented by using only a few gates, with tree architecture. As a consequence, dual tree error compensation approach is ideally suited for fast tree-based multipliers. A unsigned fixed-width multipliers has improve accuracy, silicon area, timing performances and power dissipation with respect to previously techniques. Simulation results for a 0.35- m technology show a decrease of the propagation delay up to 20%, with more than 10% power dissipation reduction.

B. Common Subexpression Elimination Method

In [9], we have presented an improved horizontal and vertical CSE which is able to reduce the number of adders and registers, but we have only considered a Multiplierless FIR filter.

CSE has stated as the problem of minimizing the numbers of the delay and adders/subtractor blocks which are needed to perform all of the multiplications. And hardware reductions achieved using the horizontal, vertical and oblique CSEs in realizing constant multipliers. The MCM area of the FIR filters has been reduced by an average of 20%. we have found that horizontal, combining horizontal and vertical, and improved horizontal and vertical CSEs have good performance in the matrix range of 800 and over.

C. Mixed Integer Linear Programming (MILP) Techniques

One of the most successful strategies is to optimize the filter coefficients in signed power-of-two (SPT) [10]–[12] space, where each coefficient is represented as a sum of a limited number of SPT terms. Thus, coefficient multiplications can be replaced by shifters and adders, so that the implementation of the filter is essentially Multiplierless. the FIR filter is designed in a discrete space such as finite word length space or SPT space to meet a given specification. In the next optimization technique is applied on the discrete coefficients to find and eliminate the common sub-expressions. Many of the optimization techniques in the second stage are based on a heuristic search or combined exhaustive search. There is an obvious disadvantage in such a two-stage optimization technique.

Mixed integer linear programming (MILP) is the technique employed to optimize the filter coefficients to meet the given specifications. In this technique, the frequency response ripple is minimized subject to a given number of adders. The obtained results may not be the optimum in terms of the number of adders, but the saving in the number of adders achieved using the proposed technique is significant compared with those obtained using other techniques that can be used to design filters with respectable length and bit width. The computation time to optimize a filter with order 60 and coefficient bit width 12 is typically within a few minutes to a few hours depending on specific examples.

D. Non-recursive Signed Common Subexpression Elimination (NRSCSE) Method

The basic method to multiply by constants without using multiplier blocks is based on the power-of-two representation of these constants. A canonic signed-digit (CSD) representation can be employed to implement the basic shift-and-add algorithm, obtaining a reduction in the number of nonzero bits compared with the binary code.
The ITM algorithm [13] describes another common subexpression- based technique. The method finds the maximum number of coincidences between two signed-digit (SD).

Dempster and Macleod demonstrated that the Bull-Horrocks modified algorithm (BHM) could reach the minimum number of logical operators. One of the advantages of the BHM is the use of integers with a larger magnitude than the coefficient, in order to reduce the number of operators. BHM allows the designer a 26% average reduction in the number of logical operators, compared with the simplest shift-and-add algorithm from the CSD representation. Although BHM leads to the lowest number of adders, the algorithm does not consider the logic depth. While the graph synthesis algorithms reduce the area, its dependence graph increases the logic depth.

BHM algorithm uses the lowest number of LO, but presents the highest LD. Hartley algorithm requires few additional logic operators respect to BHM alternative, having at the same time a minimum LD. In addition, the Hartley algorithm leads to a straightforward layout, thus simplifying hardware synthesis. Others algorithms to implement multiplierless structures like [15] and [14] (compared in [13]) are based on exhaustive search methods, that are time consuming and more difficult to translate into hardware. a new array splitting algorithm that combines the advantages of previous methods: it reduces the logic depth obtained from Hartley algorithm, using approximately the same number of logic operators than BHM. The original-array starts with a CSD representation of the coefficients, obtaining a layout similar to the Hartley description. The resulting structure can also be easily synthesized into hardware.

Compared with the most popular multiplierless algorithms, NR-SCSE offers the best relation area—frequency operation, or number of adders—logical depth along with optimal runtime operation due to its simplicity. Moreover, the structure of the filter obtained from the algorithm can easily be described using a HDL. Finally several interesting properties in both parameters LD and LO have been studied by using groups of 50 and 100 random filters.

E. Multi-root Binary Partition Graph (MBPG) Method

The integer linear programming (ILP) model, optimal subexpression sharing algorithms based on CSD coefficients have been developed [16], [17]. These algorithms have high computational complexity and are not suitable for large problem.

A new paradigm of design methodology to reduce the complexity of application-specific finite- impulse response (FIR) digital filters. The data structure of new adder graph known as the multi-root binary partition graph (MBPG) is proposed for the formulation of the multiple constant multiplication problem of FIR filter design. This method is applicable to any positional representation of coefficients. The correlation of operand lengths and adder complexity is also illustrated with two's complement ripple carry adder, from which the number of FAs required to realize each vertex of MBPG is analytically determined. This makes it possible to exploit the bit level information associated with each vertex to reduce the size of some arithmetic operators by partially modifying the graph topology. The concept of entropy and conditional entropy has been applied for the first time in this research to maximize common subexpression sharing's by set partitioning. The synthesis of a minimal MBPG provides a meaningful insight into judicious resource sharing's in MCM problem based on the rigor of proven probabilistic measures. Finally, as the widths of operands and the number of shifts are annotated on the vertices and edges of the resultant MBPG, adder complexity due to different widths and relative shifts of operands can be readily evaluated by graph traversal. Fine grain reduction of logic complexity and logic depth can be made by reordering the vertices without perturbing the size of MBPG. Experimental results on benchmark filters reported in the literature and design examples of communication filters based on D-AMPS and PDC cellular standards show that the proposed algorithm is capable of designing FIR filters with an average FA cost reduction of 40.38% over the baseline CSD implementation. The critical path delay has also been significantly reduced by 25.77% on average.

III. REVIEW OF MEMORY BASED ALGORITHMS

A. Memory-Based Realization

Offset binary coding (OBC) [18], [19], and group distributed technique [20] is used to reduce the memory-space in DA-based architectures. A decomposition scheme is suggested in [20] for reducing the memory-size of DA-based implementation of FIR filter. But, it is observed that the reduction of memory-size achieved by such decompositions, is accompanied by increase in latency as well as the number of adders and latches.

Distributed arithmetic (DA)-based computation is popular for its potential for efficient memory-based implementation of finite impulse response (FIR) filter. There are two new approaches for designing the LUT for LUT-multiplier-based implementation, where the memory-size is reduced to nearly half of the conventional approach. It is observed that transposed form realization of FIR filter is more efficient than the direct-form realization for the LUT-multiplier based implementation. In the transposed form, a single segmented-memory core could be used instead of separate memory modules for individual multiplications in order to avoid the use of individual decoders for each of those separate modules.

IV. FAITHFULLY ROUNDED AND TRUNCATED MULTIPLIERS WITH COMBINED DELETION,REDUCTION, TRUNCATION, AND ROUNDING

A faithfully rounded truncated multiplier design describes the maximum absolute error is guaranteed to be no more than 1 unit of least position, this method mutually considers the deletion, reduction, truncation, and rounding of partial product bits in order to minimize the number of full adders and half adders during tree reduction. A parallel tree multiplier design typically consists of three major steps, i.e., PP generation, PP reduction, and final carry propagate addition. PP generation produces PP bits from the multiplicand and the multiplier. The goal of PP reduction is

to compress the number of PPs to two, which is to be summed up by the final addition. The two most famous reduction methods are Wallace tree [21] and Dadda tree [22] reductions. Wallace tree reduction manages to compress the PPs as early as possible, whereas Dadda reduction only performs compression whenever necessary without increasing the number of carry-save addition (CSA) levels.

In [23], a new reduced area (RA) reduction method was described by bit width of the final carry propagate adder (CPA) is minimized. To allow for more flexible column-by-column reduction to be used in the truncated multiplier design.

V. FIR FILTER DESIGNS BASED ON FAITHFULLY ROUNDED TRUNCATED MCMA

This architecture of MCMA with truncation (MCMAT) that removes unnecessary PPBs. The white circles in the L-shape block represent the undeletable PPBs. The deletion of the PPBs is represented by black circles.

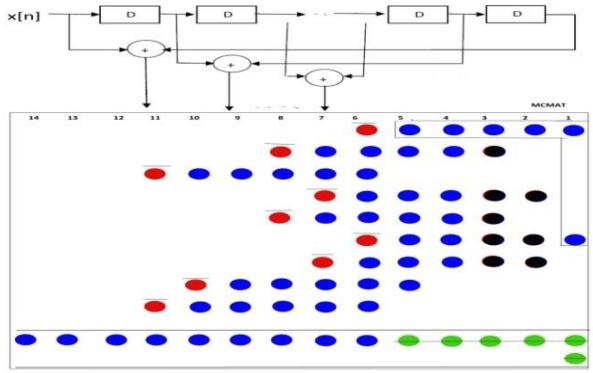

Fig 1. Overall FIR filter architecture using multiple constant multipliers/accumulators with faithfully rounded truncation (MCMAT).

After PP compression, the rounding of the resultant bits is denoted by crossed circles. The last row of the PPB matrix represents all the offset and bias constants required including the sign bit (red circles) modifications.

VI. EXPERIMENTAL RESULTS

Table-1
Comparison Of 28 Tap Fir Filter Using Various Methods

FILTER	MCM/SA (um^2)	DFFs (um^2)	Area (um^2)	Delay (ns)	TP(M data/s)	Power(mw)	Structure
NRSCSE	3151/9737	9655	22543	4.75	209	1.10	Transposed
MBPG	3141/9737	9655	22533	4.78	209	1.14	Transposed
LUT	16155/9737	9655	35547	4.77	210	1.66	Transposed
1D-DA	3146	9655	8672	4.85	17	0.36	Direct
MCMA	10161	5030	15191	6.58	152	0.90	Direct
MCMAT-1	7716	5030	12746	6.35	158	0.71	Direct
MCMAT-2	7460	5030	12490	6.35	158	0.70	Direct

VII. CONCLUSION

Here we have considered different multiplier algorithms and we conclude that, the design of FIR Filter structure using Multiple Constant Multiplication Accumulation and Truncation (MCMAT-2) leads to smallest area and low power consumption. The simulation and synthesis result show that, by using MCMAT-2 in direct form, the Performance factor such as area and power are reduced compared to MCM , MCMA and MCMA I. The costs of the MCMA designs are also significantly reduced. One of the applications of FIR filter is to design adaptive filter with radix-8/10 multiplier. This work can be extended for Adaptive filter design with radix multiplier.

REFERENCES

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

[2] F. Xu, C. H. Chang, and C. C. Jong, "Design of low-complexity FIR filters based on signed-powers-of-two coefficients with reusable common subexpressions," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 26, no. 10, pp. 1898–1907, Oct. 2007.

[3] Y. J. Yu and Y. C. Lim, "Design of linear phase FIR filters in subexpression space using mixed integer linear programming," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 54, no. 10, pp. 2330–2338, Oct. 2007.

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

[5] P. K. Meher, S. Candrasekaran, and A. Amira, "FPGA realization of FIR filters by efficient and flexible systolization using distributed arithmetic," IEEE Trans. Signal Process., vol. 56, no. 7, pp. 3009–3017, Jul. 2008.

[6] S. Hwang, G. Han, S. Kang, and J.-S. Kim, "New distributed arithmetic algorithm for low-power FIR filter implementation," IEEE Signal Process. Lett., vol. 11, no. 5, pp. 463–466, May 2004.

[7] 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.

[8] 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.

[9] J. M. Jou, S. R.Kuang, and R. D. Chen, "Design of low-error fixed-width multipliers for DSP applications," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process., vol. 46, no. 6, pp. 836–842, Jun. 1999.

[10] L. Van, S. Wang, and W. Feng, "Design of the lower error fixed-width multiplier and its application," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process., vol. 47, no. 10, pp. 1112–1118, Oct. 2000.

[11] Y. Takahashi and M. Yokoyama, .New cost-effective VLSI implementation of multiplierless FIR _lter using common subexpression elimination,. in Proc. ISCAS 2005, Kobe, Japan, May 23.26, 2005, pp. 845.848.

[12] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete power-of-two coefficient space," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-31, no. 6, pp. 583–591, Jun. 1983.

[13] Y. C. Lim and S. R. Parker, "Discrete coefficient FIR digital filter design based upon an LMS criteria," IEEE Trans. Circuits Syst., vol. CAS-30, no. 10, pp. 723–739, Oct. 1983.

[14] Y. C. Lim et al., "Signed power-of-two term allocation scheme for the design of digital filters," IEEE Trans. Circuits Syst. II, Analog Digit Signal Process., vol. 46, no. 5, pp. 577–584, May 1999.

[15] M. Potkonjak et al., "Multiple constant multiplication: Efficient and versatile framework and algorithms for exploring common subexpression elimination," IEEE Trans. Computer-Aided Design, vol. 15, no. 2, pp. 151–165, Feb. 1996.

[16] A. Dempster and M. D. Macleod, "Use of minimum-adder multiplier blocks in FIR digital filters," IEEE Trans. Circuits Syst. II, vol. 42, pp. 569–577, Sept. 1995.

[17] D. Li, "Minimum number of adders for implementing a multiplier and its application to the design of multiplierless digital filters," IEEE Trans. Circuits and Syst. II, vol. 42, pp. 453–460, July 1995.

[18] S. A. White, "Applications of the distributed arithmetic to digital signal processing:Atutorial review," *IEEE ASSP Mag.*, vol. 6, no. 3, pp. 5–19, Jul. 1989.

[19] P. K. Meher, S. Chandrasekaran, and A. Amira, "FPGA realization of FIR filters by efficient and flexible systolization using distributed arithmetic," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3009–3017, Jul. 2008.

[20] J. P. Choi, S.-C. Shin, and J.-G. Chung, "Efficient ROM size reduction for distributed arithmetic," in *Proc. IEEE Int. Symp. Circuits Syst. ISCAS*, May 2000, vol. 2, pp. 61–64.

[21] S. A. White, "Applications of the distributed arithmetic to digital signal processing:Atutorial review," IEEE ASSP Mag., vol. 6, no. 3, pp. 5–19, Jul. 1989

[22] H.-C. Chen, J.-I. Guo, T.-S. Chang, and C.-W. Jen, "A memory-efficient realization of cyclic convolution and its application to discrete cosine transform," IEEE Trans. Circuits Syst. Video Technol., vol. 15, no. 3, pp. 445–453, Mar. 2005.

[23] J. P. Choi, S.-C. Shin, and J.-G. Chung, "Efficient ROM size reduction for distributed arithmetic," in Proc. IEEE Int. Symp. Circuits Syst.ISCAS, May 2000, vol. 2, pp. 61–64.

[24] C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Electron. Comput., vol. EC-13, no. 1, pp. 14–17, Feb. 1964.

[25] L. Dadda, "Some schemes for parallel multipliers," Alta Frequenza, vol. 34, pp. 349–356, 1965.

[26] 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.

[27] Antonio G. M. Strollo, Nicola Petra, and Davide De Caro,"Dual-Tree Error Compensation for High Performance Fixed-Width Multipliers," IEEE Trans. Electron. Comput., VOL. 52, NO. 8, AUGUST 2005.[28] Yasuhiro Takahashi, Toshikazu Sekine Michio Yokoyama,"A Comparison of Multiplierless Multiple Constant Multiplication using Common Subexpression Elimination Method," IEEE Trans. Electron. Comput., 978-1-4244-2167-1/08/$25.00 ©2008