International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

IJIEMR Transactions, online available on 19th June 2017. Link :
http://www.ijiemr.org/downloads.php?vol=Volume-6&issue=ISSUE-4

Title: Design And Implementation Of Bit Level Optimization Of Adder Sw-Tree For Mcm For Fir Filter.

Volume 06, Issue 04,Page No: 1108 - 1116.

Paper Authors

*** G USHARANI, M SATEESH BABU.**

* Kakinada Institute of Engineering And Technology For Women,Korangi.

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

DESIGN AND IMPLEMENTATION OF BIT LEVEL OPTIMIZATION OF ADDER SW-TREE FOR MCM FOR FIR FILTER

***G USHARANI, **M SATEESH BABU**

P.G.Scholar, Kakinada Institute of Engineering And Technology For Women,Korangi.

, Assistant Professor, Kakinada Institute of Engineering And Technology For Women,Korangi.

ABSTRACT:

Multiple constant multiplications (MCM) scheme is widely used for implementing transposed direct-form FIR filters. While the research focus of MCM has been on more effective common sub-expression elimination, the optimization of addertrees, which sum up the computed sub-expressions for each coefficient, is largely omitted. In this paper, we have identified the resource minimization problem in the scheduling of adder-tree operations for the MCM block, and presented a mixed integer programming (MIP) based algorithm for more efficient MCM-based implementation of FIR filters. Experimental result shows that up to reduction of area and reduction of power can be achieved on the top of already optimized adder/subtractor network of the MCM block. A digital filter is a system that performs mathematical operations on sampled, discrete time signal to reduce or enhance certain aspect of that signal. There are two types of digital filter mainly used that are infinite response (IIR) filter and finite impulse response (FIR) filter.

Keywords: Streaming Content, Leakage Detection, Traffic Pattern, Degree of Similarity.

I. INTRODUCTION

A finite impulse response (FIR) filter is a filter whose impulse response (or response to any finite length input) is of finite duration, because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which may have internal feedback and may continue to respond indefinitely (usually decaying) The impulse response (that is, the output in response to a Kronecker delta input) of an Nth-order discrete-time FIR filter lasts

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

exactly N + 1 samples (from first nonzero element through last nonzero element) before it then settles to zero. FIR filters can be discrete-time or continuous-time, and digital or analog. The direct form has one huge addition at the output, which maps well in multiple accumulation unit (MAC) operations on a digital signal processing (DSP) processor but hardware implementation is complex. The direct form also needs extra pipeline registers to reduce delay of the adder tree. The transposed form of FIR filter already has registers between the adders and achieve high throughput and less delay without adding any extra pipeline registers between the adders. The transposed form also has many small addition separated by delay element. The number of delay element is more in the transposed form and adding delay element to the structures. It makes the design faster, shows the direct form of FIR filter in which the output is obtained by performing the concurrent multiplications of individual delayed signals and respective filter coefficients, followed by accumulation of all the products.), the inputs to the multiplier are the current input signal x[n] and coefficients. The results of individual multiplier go through adder block and delay elements. , there are many papers on the designs and implementations of lowcost or high-speed FIR filters. A finite impulse response (FIR) filter is a digital filter that works on digital inputs. A digital filter is a system that performs mathematical operations on sampled, discrete time signal to reduce or enhance certain aspect of that signal. There are two types of digital filter mainly used that are infinite response (IIR) filter and finite impulse response (FIR) filter. FIR stands for Finite IR filters, whereas IIR stands for Infinite IR filters. IIR and FIR filters are utilized for filtration in digital systems. FIR filters are more widely in use, because they differ in response. FIR filters have only numerators when compared to IIR filters, which have both numerators and denominators.

II. LITERATURE REVIEW

D. R. Bull and D. H. Horrocks: The authors outline a design methodology for the realization of digital filtering structures with significantly reduced numbers of elementary arithmetic operations. The directed acyclic graphs which result from the design algorithm completely describe the filter arithmetically and may be mapped directly onto hardware or software realizations. Vertex rearrangement, retiming and edge elimination techniques are presented which facilitate the generation of a

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

logical graph with an efficient allocation of pipeline registers. An example of the technique is given for a bit-serial realization employing a bit-level pipeline. A. G. Dempster and M.D. Macleod: The computational complexity of VLSI digital filters using fixed point binary multiplier coefficients is normally dominated by the number of adders used in the implementation of the multipliers. It has been shown that using multiplier blocks to exploit redundancy across the coefficients results in significant reductions in complexity over methods using canonic signeddigit (CSD) representation, which in turn are less complex than standard binary representation. Three new algorithms for the design of multiplier blocks are described: an efficient modification to an existing algorithm, a new algorithm giving better results, and a hybrid of these two which trades off performance against computation time.

Significant savings in filter implementation cost over existing techniques result in all three cases. For a given word length, it was found that a threshold set size exists above which the multiplier block is extremely likely to be optimal. In this region, design computation time is substantially reduced.

Existing Method:

In Existing Design Wallace and modified Booth multipliers, have been proposed, the full flexibility of a multiplier is not necessary for the constant multiplications, since filter coefficients are fixed and determined beforehand by the DSP algorithms. Hence, the multiplication of filter coefficients with the input data is generally implemented under shift adds architecture, where each constant multiplication is realized using addition/subtraction and shift operations in an MCM operation.

III. SYSTEM DESIGN

A. Proposed Method

We present a method of derivation of equivalent addertrees to minimize the adder tree resource. We have developed the cost model of the shift-ADD/SUB network by bit-level analysis, which could be reduced by suitable scheduling of operations on the adder-tree. A great deal of research has been done to develop effective algorithms to identify the optimal set of non-redundant sub expressions to achieve the minimum number of logic operators and the minimum logic depth of the MCM

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

B. Module:

- Absolute shift propagation
- Edge Shifts
- Absolute shift propagation
- MCM Operation 1
- MCM operation 2
- Greedy scheduling

C. Module Description

1. Adder-Tree Scheduling Problem

Given input terms T ={ } and their earliest arrival time/delay Di , the objective of an adder-tree scheduling algorithm is to define an assignment of binary addition and subtraction operations to sum up the input terms such that the total delay to produce the final output is minimized.

2. Greedy Adder-Tree Scheduling

The common practice of handling the summation of CS terms of each coefficient is to use the tree-height minimization algorithm [10] to produce a height optimum adder-tree. The tree-height minimization algorithm iteratively collapses the pair {Ti , Tj}with smallest delays using an ADD/SUB to form a new term with delay max (Di ,Dj)+ 1 , until a single term is reduced to. Fig. 2 gives an example of the schedule for an adder-tree on the left with minimum delay

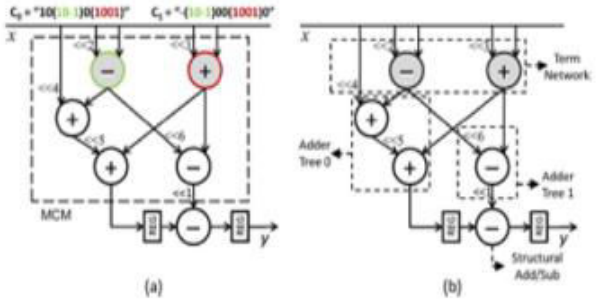


Fig1. Composition of the MCM block.

(a) MCM and common sub expressions.

(b) Term network and addertrees for each coefficients

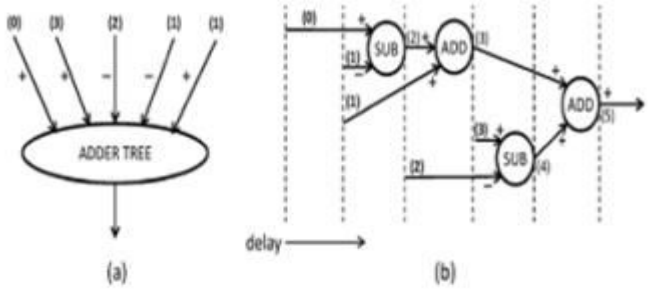


Fig2. (a) an example adder-tree with delays and signs on each input term,

(b) An internal schedule with minimum delay.

Note that either a positive or negative sign is associated with each input term (see Fig. 3.2(a)), which denotes whether the corresponding term should be added to or subtracted from the summation. These signs also determine whether an addition operation or

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

a subtraction operation should be used when the algorithm collapses a pair of terms in the adder-tree based on the following rules. (1) If two input edges are of the same sign, an ADD will be used; otherwise, it will be a SUB. (2) The sign of the output edge is always the same as that of the "left" input edge (i.e., the minuend edge in the subtraction case).Using these two rules, it is possible that the final term producing the summation result may carry a negative sign, such that a negation is needed after the adder-tree to correct the value. For an FIR filter, results from multiple adder trees are accumulated by a structural adder-register line. So the negation can be eliminated by replacing the structural adder with a subtractor (see coefficient C1in Fig. 1 for an example).

3. Cost Model

In order to quantify and minimize the hardware cost of the adder-tree, we model the cost of ADD/SUB operations in this section based on the ripple carry implementation, which is most area efficient and will be picked up by the hardware compiler whenever the timing allows. Without loss of generality, for a single ADD/SUB operation, the pair of its input operands may be of different bit-widths, and one of them is to be left shifted by certain bit positions. We enumerate all the scenarios of shift-add/sub operations in Fig4.

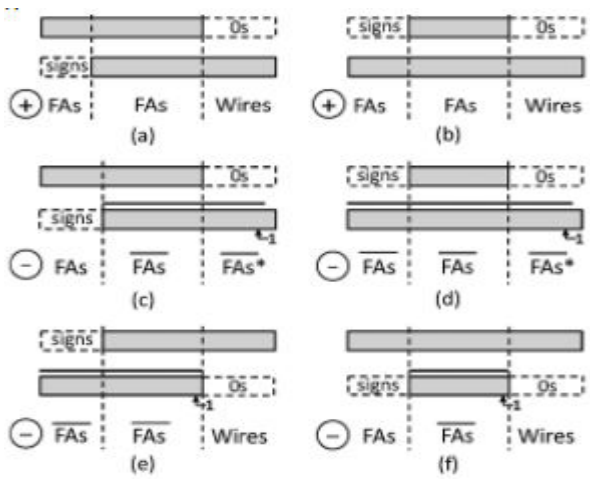


Fig3. Cost of ADD/SUB operation under various input scenarios. Notations: FA—Full Adder, above line— invertor (INV). (a),(b) Cases for ADD. (c),(d),(e),(f)Cases for SUB.

The cost calculation is done separately in three bitsegments. Starting from the least significant bit (LSB), the 1st segment covers the bit positions up to but not including the first bit of the shifted operand; the 3rd segment covers the bits corresponding to the sign extension bits of the sign extended operand; the 2nd segment takes the rest of the bit positions. Two cases of ADD operation are shown in Fig. 3(a) and (b). In both cases, the 2nd and 3rd segments are implemented by one Full Adder (FA) per bit, while the 1st segment cost nothing than wiring. Four cases of SUB operation are

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

shown in Fig. 3(c)–(f). In the first two cases where the shift is with the minuend, the 1st segment is implemented by FAs with invertors on the minuend bits, except for a single special case at the LSB using direct wire connection1 to save a pair of FA and invertor.

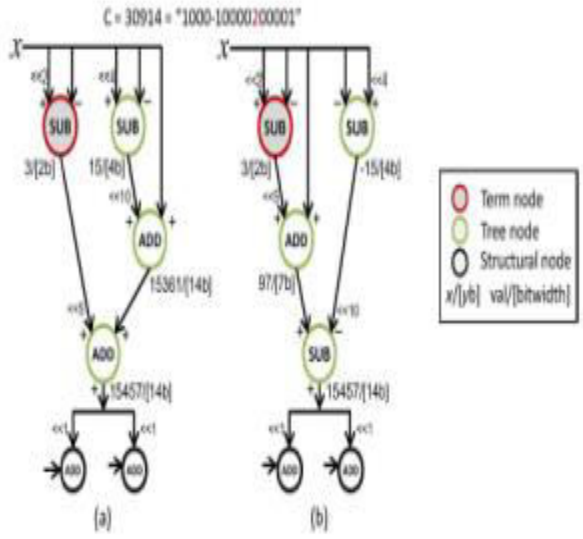


Fig4. (a) Adder-tree by greedy scheduling algorithm.

(b) Bit-level resource optimal adder-tree.

The 1 st segments for the last two cases are wires. The 2nd segment for all cases is implemented in pairs of FAs and invertors. For the 3rd segment, when the sign extension bits are from the subtrahend, inverters are not needed since these bits simply take the value of the inverted sign of the subtrahend. This cost model is verified experimentally from synthesis results of Synopsis Design Compiler for ASICs, and is applicable to cases where either or both of input operands are unsigned signals.

D. Motivational Example

The number of operators on an adder-tree is determined by the number of input terms the coefficient uses from the term network. For a input adder-tree N-1, operators are required. A motivational example in Fig. 4 shows the key differences between a greedily scheduled adder-tree based on the height minimization algorithm and the resource optimized addertree of the same height. First, the bit widths2 of intermediate nodes are significantly smaller. For example, while other adder-tree nodes are of similar bit widths, the one in the second layer is only of 7-bits in the optimal adder-tree (Fig. 4(b)) compared to 14-bits of the non-optimal adder-tree. Note that wider bit width signal may contribute further to higher hardware cost when input to the next layer of operators. Second, subtractions with shift on the subtrahend also reduces operators' cost. For example, for base bit width of 8- bits, the operator performing "1 << 4 - 1 " on Fig. 4(a) costs 11 FAs and 7 INVs according to the cost model, while the operator performing "1 – 1 << 4 " on Fig. 4(b) costs 8 FAs and 8 INVs.

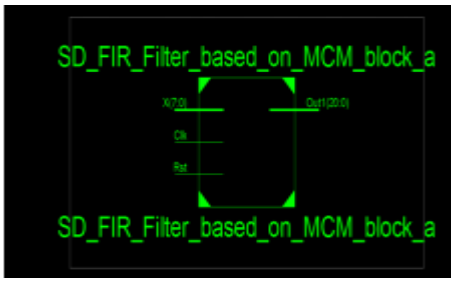
International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

In total, the optimal adder-tree sums up to 30 FAs and 20 INVs, while the non-optimal one is of 40 FAs and 7 INVs.Assuming the ratio of resource consumption of an FA to that of an INV is around 8 to 1, nearly 18% resource is reduced by using the optimal adder-tree in this example. Note that the adder-tree also determines the type of operators used for its structural accumulation. An output edge carrying a (Plus 1 to inverted LSB is equivalent to wiring of the LSB and moving the plus 1 to the second LSB. This optimization is done by most synthesis tools. This refers to additional bitwidth imposed by the ADD/SUB network on top of the base bit width of input signal.) "-" sign requires a SUB on the accumulation line, which usually consumes more hardware than an ADD. For linear phase FIR filters where coefficients are symmetric, each adder-tree corresponds to 2 structural operators.

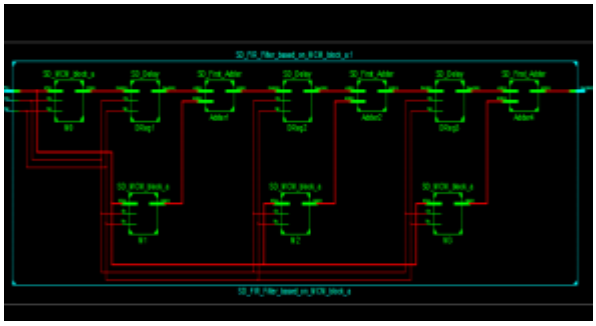
E. Logic Depth Relaxation

The clock performance of the entire FIR filter is decided by the largest of the delays of all coefficients. Assuming the delay of an ADD/SUB operator to be 1 unit, the delay of the constant multiplication by a coefficient can be simply measured by the number of ADD/SUB steps on a maximal path in the part of the network corresponding to the coefficient. We generally use logic depth to describe the required ADD/SUB steps. For a coefficient whose logic depth is less than the filter's logic depth, incrementing (relaxing) its logic depth may reduce the resource consumption.

IV. RESULTS



Block diagram of MCM block



RTL Schematic of MCM block

Simulation result for the proposed adder SW tree.

V. CONCLUSION

In this paper, we have identified the resource minimization problem in the scheduling of adder-tree operations for the MCM block of transposed direct-form FIR filter, and presented an MIP-based algorithm for exact bit-level resource optimization. Experimental result shows that up to reduction of area and can be achieved on top of already optimized ADD/SUB networks of MCM blocks. Further exploration of efficient heuristic algorithms for resource minimization of adder-trees of FIR filters could be done in the future

REFERENCES

[1] Smart card access control system.http://www.smartkeyrfid.com.

[2] Digi international, http://www.digi.com.

[3]L. Aksoy, C. Lazzari, E. Costa, P. Flores, and J. Monteiro, "Design of digit-serial FIR filters: Algorithms, architectures, and a CAD tool," IEEE Trans. Very Large Scale Integration (VLSI) Syst., vol. 21, no. 3, pp. 498–511, Mar. 2013.

[4] M. B. Gately, M. B. Yeary, and C. Y. Tang, "Multiple real-constant multiplication with improved cost model and greedy and optimal searches," in Proc. IEEE ISCAS, May 2012, pp. 588–591.

[5] M. Kumm, P. Zipf,M. Faust, and C.-H. Chang, "Pipelined adder graph optimization for high speed multiple constant multiplication," in Proc.IEEE ISCAS, May 2012, pp. 49–52.

[6] P. K. Meher and Y. Pan, "Mcm-based implementation of block fir filters for high-speed and low-power applications," in Proc. VLSI andSystem-on-Chip (VLSI-SoC), 2011 IEEE/IFIP 19th Int. Conf., Oct.2011, pp. 118–121.

[7]Feng Lu; Xiaoshengchen;ye,T.T.," performanceanalysis of stacked RFID tags." 2009 IEEE international conference on RFID.

[8] Floerkemeier c sarma s an overview of RFID system interfaces and reader protocols,RFID,2008 IEEE international conference.

[9] R. Mahesh and A. Vinod, "A new common subexpression elimination algorithm for realizing low-complexity higher order digital filters," IEEE Trans. Computer-Aided

Des.Integr.Circuits Syst., vol. 27, no. 2, pp. 217–229, 2008.

[10] Y. Voronenko and M. Püschel, "Multiplierless multiple constant multiplication," ACM Trans. Algorithms, vol. 3, no. 2, 2007.

[11] L.o'Gorman,"Comparing passwords, tokens, & Biometrics for the user authentication, proceedings of the IEEE,Dec 2003.

[12] D.maltoni, D.manio, A.K.Jain&S.prabhakar Hand book of finger print recognition, sprinager2003.

[13] I. C. Park and H. J. Kang, "Digital filter synthesis based on minimal signed digit representation," in Proc. Design Autom. Conf. (DAC), 2001. conferencepatternsrecognition (ICPR) (Sep-2000) vol.1, PP. 25-29.

[14] Prabhakar, S., Wang, J., Jain,A.K., Pankanti, S., and Bolle, R.Minutiae verification & classification for finger print matching. In proc. 15thinternational

[15] Abutaleb and Kamal(1999). Abutaleb A. S. and Kamal m., " A genetic algorithm for the estimation of ridges in finger prints", IEEE transactionOn image processing, vol.8, no.8,p.1134,1999.

AUTHORS:

M SATEESH BABU Received the M.TECH degree in DSCE from JNTUA COLLEGE OF ENGINEERING,ANANTHAPUR,
ASST.PROF. IN ECE DEPT...,KIET WOMEN.

USHARANI GALI pursuing M.Tech VLSI&ES in Kakinada Institute of Engineering and technology for Women(JNTUK) Korangi. She received Bachelor degree in Department of Electronics and Communication Engineering from ADITYA ENGINEERING COLLEGE(JNTUK) SURAMPALAM.