# An Extensive Literature Review on Adder-Trees for Multiple Constant Multiplication FIR Filter

**Pragati Mahale[1] Prof. Ruby Awasthi[2] & Dr. Rita Jain[3]**

*[1]M-Tech Research Scholars, [2]Research Guide, [3]HOD*

*Department of Electronics & Communication Engineering LNCT, Bhopal*

*Abstract - Multiplying by known constants is a common operation in many digital signal processing (DSP) algorithms. High performance DSP systems are implemented in custom hardware, in which the designer has the ability to choose which logic elements will be used to perform the computation. By exploiting the properties of binary multiplication, it is possible to realize constant multiplication with fewer logic resources than required by a generic multiplier.*

*Keyword: Critical path, genetic algorithm, high-speed, multiple constant multiplications (MCM).*

## I. INTRODUCTION

Multiplying a variable by a set of known constant coefficients is a common operation in many digital signal processing (DSP) algorithms. Compared to other common operations in DSP algorithms, such as addition, subtraction, using delay elements, etc., multiplication is generally the most expensive. There is a trade-off between the amount of logic resources used (i.e. the amount of silicon in the integrated circuit) and how fast the computation can be done. Compared to most of the other operations, multiplication requires more time given the same amount of logic resources and it requires more logic resources under the constraint that each operation must be completed within the same amount of time.

A general multiplier is needed if one performs multiplication between two arbitrary variables. However, when multiplying by a known constant, can exploit the properties of binary multiplication in order to obtain a less expensive logic circuit that is functionally equivalent to simply asserting the constant on one input of a general multiplier. In many cases, using a cheaper implementation for only multiplication still results in significant savings when considering the entire logic circuit because multiplication is relatively expensive. Furthermore, multiplication could be the dominant operation, depending on the application.

There are many hand-held products that include digital signal processing (DSP), for example, cellular phones and hearing aids. For this type of portable equipment a long battery life time and low battery weight is desirable. To obtain this the circuit must have low power consumption.

### Digital Filters

Frequency selective digital filters are used in many DSP systems. The filters studied here are assumed to be causal, linear, time-invariant filters.

### IIR Filters

Infinite impulse response (IIR) is a property applying to many linear time-invariant systems. Common examples of linear time-invariant systems are most electronic and digital filters. Systems with this property are known as IIR systems or IIR filters, and are distinguished by having an impulse response which does not become exactly zero past a certain point, but continues indefinitely. This is in contrast to a finite impulse response in which the impulse response h(t) does become exactly zero at times t > T for some finite T, thus being of finite duration.

In practice, the impulse response even of IIR systems usually approaches zero and can be neglected past a certain point. However the physical systems which give rise to IIR or FIR (finite impulse response) responses are dissimilar, and therein lies the importance of the distinction. For instance, analog electronic filters composed of resistors, capacitors, and/or inductors (and perhaps linear amplifiers) are generally IIR filters. On the other hand, discrete-time filters (usually digital filters) based on a tapped delay line employing no feedback are necessarily FIR filters. The capacitors (or inductors) in the analog filter have a "memory" and their internal state never completely relaxes following an impulse. But in the latter case, after an impulse has reached the end of the tapped delay line, the system has no further memory of that impulse and has returned to its initial state; its impulse response beyond that point is exactly zero.

### FIR Filters

In signal processing, 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 exactly N + 1 samples (from first nonzero element through last nonzero element) before it then settles to zero.

### Applications of Constant Multiplication

Multiplication by a set of constants occurs when multiplying by a constant vector or a constant matrix. For example, the dot product $a \cdot b$ gives the scalar projection of $a$ on to b (or vice versa). Multiplication by a constant matrix is nothing more than performing the dot product between several constant vectors (which collectively form a matrix) and a variable vector (the elements of this vector are the inputs). Multiplying by a constant matrix can thus be regarded as a linear transformation of coordinates, which is used in many applications. For example, the conversion from the RGB (red, green, blue) color space to the YUV color space (Y represents brightness s, U and V represent chroma) involves multiplication by a constant 3x3 matrix. Because the human eye is more sensitive to brightness than coloring (chroma), we can compress the information in the U and V components with only a minor perceived distortion. This is exploited in JPEG and MPEG for compressing images and video, respectively. However, most display devices require color to be provided in the RGB format, hence the need for color space conversion. Any application which involves a predefined linear transformation of coordinates will use multiplication by a set of constants.

### Custom Hardware

In order to achieve higher computational throughput and/or lower energy consumption, many DSP algorithms are implemented in custom hardware. Examples of custom hardware include application specific integrated circuits (ASICs) and field programmable gate arrays (FPGAs). These benefits are further enhanced by finding lower cost implementations of constant coefficient multiplication, as smaller and cheaper implementations of DSP systems are obtained. In ASICs, a smaller logic circuit results in less material costs and lower energy consumption ceteris paribus.

Alternatively, for the same amount of silicon, a higher computational throughput can be obtained by placing more computational units in parallel. FPGAs typically have some dedicated multipliers, but a large DSP design could require more multiplications than available multipliers, in which case one must build soft multipliers (i.e. using the programmable logic elements). In FPGAs, finding an implementation that requires less logic elements may enable one to fit the system on a smaller and likely cheaper FPGA, among various other benefits. The benefits manifest when computational units are used in parallel. For example, the total computational throughput may need to be high enough to satisfy a real-time constraint (such as a video decoder which must output a frame of video of size $x$ by $y$ pixels every $z$ seconds).

## II. LITERATURE SURVEY

X. Lou, Y. J. Yu and P. K. Meher, [1] in this research work, critical path of multiple constant multiplication (MCM) block is analyzed precisely and optimized for high-speed and low-complexity implementation. A delay model based on signal propagation path is proposed for more precise estimation of critical path delay of MCM blocks than the conventional adder depth and the number of cascaded full adders. A dual objective configuration optimization (DOCO) algorithm is developed to optimize the shift-add network configuration to derive high-speed and low-complexity implementation of the MCM block for a given fundamental set along with a corresponding additional fundamental set. A genetic algorithm (GA)-based technique is further proposed to search for optimum additional fundamentals. In the evolution process of GA, the DOCO is applied to each searched additional fundamental set to optimize the configuration of the corresponding shift-add network. Experimental results show that the proposed GA-based technique reduces the critical path delay, area, power consumption, area delay product and power delay product by 32.8%, 4.2%, 5.8%, 38.3%, and 41.0%, respectively, over other existing optimization methods.

R. Pasko, P. Schaumont, V. Derudder, S. Vernalde and D. Durackova, [2] The problem of an efficient hardware implementation of multiplications with one or more constants is encountered in many different digital signal-processing areas, such as image processing or digital filter optimization. In a more general form, this is a problem of common sub expression elimination, and as such it also occurs in compiler optimization and many high-level synthesis tasks. An efficient solution of this problem can yield significant improvements in important design parameters like implementation area or power consumption. In this research work, a new solution of the multiple constant multiplication problem based on the common subexpression elimination technique is presented. The performance of their method is demonstrated primarily on a finite-duration impulse response filter design. The idea is to implement a set of constant multiplications as a set of add-shift operations and to optimize these with respect to the common sub expressions afterwards. Authors show that the number of add/subtract operations can be reduced significantly this way. The applicability of the presented algorithm to the different high-level synthesis tasks is also

indicated. Benchmarks demonstrating the algorithm's efficiency are included as well.

M. Martinez-Peiro, E. I. Boemo and L. Wanhammar, [3] In this work, a new algorithm called no recursive signed common sub expression elimination (NR-SCSE) is discussed, and several applications in the area of multiplier less finite-impulse response (FIR) filters are developed. While the recursive utilization of a common sub expression generates a high logic depth into the digital structure, the NR-SCSE algorithm allows the designer to overcome this problem by using each sub expression once. The research work presents a complete description of the algorithm, and a comparison with two other well-known options: the graph synthesis, and the classical common sub expression elimination technique. Main results show that the NR-SCSE implementations of several benchmark circuits offer the best relation between occupied area and logic depth respect to the previous values published in the technical literature.

Fei Xu, Chip-Hong Chang and Ching-Chuen Jong,[4] In this research work, a new algorithm, called contention resolution algorithm for weight-two sub expressions (CRA-2), based on an ingenious graph synthesis approach has been developed for the common sub expression elimination of the multiplication block of digital filter structures. CRA-2 provides a leeway to break away from the local minimum and the flexibility of varying optimization options through a new admissibility graph. It manages two-bit common sub expressions and aims at achieving the minimal logic depth as the primary goal. The performances of their proposed algorithm are analyzed and evaluated based on benchmarked finite-impulse-response filters and randomly generated data. It is demonstrated that CRA-2 achieves the shortest logic depth with significant reduction in the number of logic operators compared with other reported algorithms.

O. Gustafsson and L. Wanhammar, [5] Subexpression sharing is an important implementation issue when one data is multiplied with many constants or a sum of products is computed. By modelling the sub expression sharing problem using integer linear programming (ILP) an optimal solution can be found. Further, the model can be directly incorporated with the design of algorithms that have linear design constraints, e.g., linear-phase FIR filters. The proposed method is compared with previously reported algorithms. It produces better results than other sub expression sharing methods, even though it is still not comparable with the optimal method based on graph representation. However, the possibility to expand the ILP model beyond sub expression sharing is discussed. This would then produce identical results to the optimal adder graph method.

## III. PROBLEM IDENTIFICATION

The critical path of MCM blocks are analyzed based on the signal path and a fine-grained delay model for CPD estimation had been proposed. Based on precise estimate of critical path, authors have proposed an algorithm named DOCO to optimize the shift-add network configuration of MCM blocks for the reduction of CPD and hardware complexity subject to an additional fundamental set. In order to find the optimum additional fundamentals for a given fundamental set, a GA-based search method is proposed. The DOCO algorithm is adopted in the proposed GA-based technique to optimize the shift-add network configurations. Experimental results show that a solution generated by the proposed GA-based technique outperforms existing algorithms in terms of CPD, area, power consumption, ADP and PDP. The CPD, area, power, ADP, and PDP are reduced by 32.8%, 4.2%, 5.8%, 38.3%, and 41.0%, respectively, in average over the existing algorithms.

## IV. CONCLUSION

In this review this study has been done that can minimize the energy consumption per operation for the arithmetic parts of DSP circuits, such as digital filters. More specific, the focus is on single- and multiple-constant multiplication using serial arithmetic. The possibility to reduce the complexity and energy consumption is investigated. The main difference between serial and parallel arithmetic, which is of interest here, is that a shift operation in serial arithmetic require a flip-flop, while it can be hardwired in parallel arithmetic.

## REFRENCES

[1] X. Lou, Y. J. Yu and P. K. Meher, "Fine-Grained Critical Path Analysis and Optimization for Area-Time Efficient Realization of Multiple Constant Multiplications," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 62, no. 3, pp. 863-872, March 2015.

[2] R. Pasko, P. Schaumont, V. Derudder, S. Vernalde and D. Durackova, "A new algorithm for elimination of common subexpressions," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 18, no. 1, pp. 58-68, Jan 1999.

[3] M. Martinez-Peiro, E. I. Boemo and L. Wanhammar, "Design of high-speed multiplierless filters using a nonrecursive signed common subexpression algorithm," in IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 49, no. 3, pp. 196-203, Mar 2002.

[4] Fei Xu, Chip-Hong Chang and Ching-Chuen Jong, "Contention resolution algorithm for common subexpression elimination in digital filter design," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 52, no. 10, pp. 695-700, Oct. 2005.

[5] O. Gustafsson and L. Wanhammar, "ILP modelling of the common subexpression sharing problem," Electronics, Circuits and Systems, 2002. 9th International Conference on, 2002, pp. 1171-1174 vol.3.

[6] L. Aksoy, E. Costa, P. Flores and J. Monteiro, "Optimization of area under a delay constraint in digital filter synthesis using SAT-based integer linear programming," 2006 43rd ACM/IEEE Design Automation Conference, San Francisco, CA, 2006, pp. 669-674.

[7] A. Hosangadi, F. Fallah and R. Kastner, "Reducing hardware complexity of linear DSP systems by iteratively eliminating two-term common subexpressions," Proceedings of the ASP-DAC 2005. Asia and South Pacific Design Automation Conference, 2005., 2005, pp. 523-528 Vol. 1.

[8] L. Aksoy, E. da Costa, P. Flores and J. Monteiro, "Exact and Approximate Algorithms for the Optimization of Area and Delay in Multiple Constant Multiplications," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 27, no. 6, pp. 1013-1026, June 2008.

[9] D. R. Bull and D. H. Horrocks, "Primitive operator digital filters," in IEE Proceedings G - Circuits, Devices and Systems, vol. 138, no. 3, pp. 401-412, June 1991.

[10] A. G. Dempster and M. D. Macleod, "Constant integer multiplication using minimum adders," in IEE Proceedings - Circuits, Devices and Systems, vol. 141, no. 5, pp. 407-413, Oct 1994.

[11] O. Gustafsson, A. G. Dempster and L. Wanhammar, "Extended results for minimum-adder constant integer multipliers," Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on, 2002, pp. I-73-I-76 vol.1.

[12] A. G. Dempster and M. D. Macleod, "Use of minimum-adder multiplier blocks in FIR digital filters," in IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 42, no. 9, pp. 569-577, Sep 1995.

[13] F. Xu, J. J. Chen, C. H. Chang, and C. C. Jong, "A modified reduced adder graph algorithm for multiplier block minimization in digital filters," in Proc . IEEEAs ia - Pacific Conf . Circuits Syst ., Dec. 2004, vol. 2, pp. 705–708.

[14] R. I. Hartley, "Optimization of canonic signed digit multipliers for filter design," in Proc . IEEE Int . Symp. Circuits Syst., Singapore, Jun. 1991, vol. 4, pp. 1992–1995.

[15] D. Bull and D. Horrocks, "Primitive operator digital filters," in IEE Proc. Inst. Elec. Eng. Circuits , Devices, Syst., Jun. 1991, vol. 138, no. 3, pp. 401–412.