

# A Comparison of Multiplierless Multiple Constant Multiplication using Common Subexpression Elimination Method

Yasuhiro Takahashi, Toshikazu Sekine  
 Department of Electrical and Electronic Engineering  
 Gifu University,  
 1-1 Yanagido, Gifu-shi 501-1193 Japan  
 Email: {yasut, sekine}@gifu-u.ac.jp

Michio Yokoyama  
 Department of Bio-system Engineering  
 Yamagata University,  
 4-3-16 Jonan, Yonezawa-shi 992-8510 Japan  
 Email: yoko@yamagata-u.ac.jp

**Abstract**—The common subexpression elimination (CSE) techniques address the issue of minimizing the number of adders needed to implement the multiple constant multiplication (MCM) blocks. In this paper, we provide a comparison of hardware reductions achieved using the horizontal, vertical, oblique and combining horizontal and vertical CSEs in realizing constant multipliers. Our FPGA implementation results included in 52 MCM examples show that three different (horizontal, horizontal and vertical, and efficient horizontal and vertical) CSEs have a good area-time product performance, in the MCM matrix range of 800 and over.

## I. INTRODUCTION

In the digital signal processing (DSP) algorithms, many fixed transforms (e.g., FIR/IIR filter with fixed coefficients, DCT, DFT, etc) do not require the flexibility of a general-purpose multiplier as the multiplicand has a limited number of values. From this reason, it is attractive to carry out the multiplication by using shifts and adds. The shifts can be realized by using hard-wired shifters and hence they are essentially free. Furthermore, we can reduce the adder area by using the common subexpression elimination (CSE) techniques. The CSE tackles the multiple constant multiplication (MCM) problem [1], [2] by minimizing the number of additions through extracting common parts among the constants represented in canonic signed digit (CSD) [3]–[7], [12]–[15]. There are three different kinds of common subexpressions: horizontal, vertical and oblique. Due to the computational complexity and the fact that linear phase FIR filters are symmetrical, the search for redundant computations in multiplier block is normally confined to horizontal common subexpressions. Recently, Jang et al. [5] proposed a method of further reducing the number of adders by using vertical CSE, and Vinod et al. [6] proposed a combining horizontal and vertical CSE. However, the structures for these techniques are designed without any consideration of the number of registers (i.e. time delay elements). The gate number ratio of adders to registers is 1 : 0.6–0.8 [8]; therefore, in case of structure with many registers, the implementation cost cannot be reduced. In our previous paper [7] we have presented an improved horizontal and vertical CSE which is able to reduce the number of adders

and registers, but we have only reported on the implementation results of multiplierless FIR filter.

In this paper, we report a comparison of hardware reductions achieved using the horizontal, vertical, oblique and combining horizontal and vertical CSEs in realizing *multiple constant multipliers* (MCM). The rest of this paper is organized in four sections. Section II describes the definition of the MCM. Section III provides a brief review of the five different CSE approaches: horizontal, oblique [3], vertical [5], horizontal and vertical [6], and efficient horizontal and vertical [7]. Section IV presents the FPGA implementation results of the 52 MCM design examples and discussion. Finally, conclusions are drawn in section V.

## II. MULTIPLE CONSTANT MULTIPLICATION

A common feature of many digital signal processing algorithms is that they involve computations of the form

$$Y_i = a_{ij}X_j \quad (i = 0, 1, \dots, N-1; j = 0, 1, \dots, M-1), \quad (1)$$

where  $X_i$  and  $Y_i$  are input and output variable vectors, respectively. Also,  $a_{ij}$  is a set of constant coefficients,  $N$  is the number of coefficients and  $M$  is the word length. In other words, this is a matrix variable multiplication of the form

$$\begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{N-1} \end{bmatrix} = \begin{bmatrix} a_{00} & \cdots & a_{0M-1} \\ a_{11} & \cdots & a_{1M-1} \\ \vdots & \ddots & \vdots \\ a_{N-11} & \cdots & a_{N-1M-1} \end{bmatrix} \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix}, \quad (2)$$

and its block diagram is shown in Fig. 1. One typical example is the transposed form FIR filter that one input data is multiplied with the filter coefficients. In this paper, we perform multiple multiplications in Equation (2) using the registers and the adders/subtractors in order to reduce the area. Then the problem of reducing the costs is stated as the problem of minimizing the weighted sum of the numbers of the registers and adders/subtractors which are needed to perform all of the multiplications. That is, the objective cost function (CF) to be

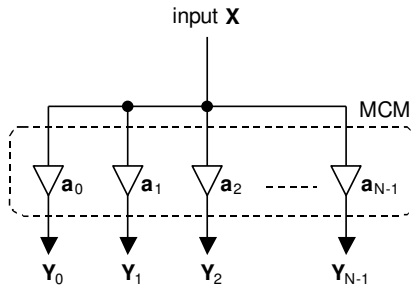


Fig. 1. Block diagram of MCM circuit.

minimized is written as:

$$CF = \beta N_{\text{reg}} + \gamma N_{\text{as}} \quad (\beta > 0, \gamma > 0), \quad (3)$$

where  $N_{\text{reg}}$  and  $N_{\text{as}}$  are the number of registers and adders/subtractors, respectively,  $\beta$  and  $\gamma$  are weights.

The above is called the multiple constant multiplication (MCM) problem. But the MCM problem is very complex that it is believed to be NP-hard. Hence, we have to find heuristics referred to as the CSE.

### III. COMMON SUBEXPRESSION ELIMINATION METHOD

Common subexpression elimination proposed to tackle the MCM problem minimizes the number of additions by extracting the common parts among the constructs represented in binary form [1], [2]. Recently, some new techniques for CSE, oblique (i.e. Hartley) [3], vertical (Jang et al.) [5], and combining horizontal and vertical (Vinod et al.) CSE technique [6] have been proposed a powerful solution to reduce the complexity in MCM, using the CSD representation.

Figure 2 illustrates the three different (horizontal, vertical and oblique) types of common subexpressions where  $\bar{1}$  denotes  $-1$ . The shared cells in this figure indicate contentions, where two or more common subexpressions share the same nonzero digit. A contention implies a potential inhibition of sharing some common subexpressions. The notations shown in Fig. 3 are used to express the three different types of subexpression. In this figure,  $x[-i]$  denotes the value of  $x$  after  $i$  sample delays and  $\ll j$  stands for left shift of  $j$  digits.  $i$  and  $j$  can be deemed as the height and length of the

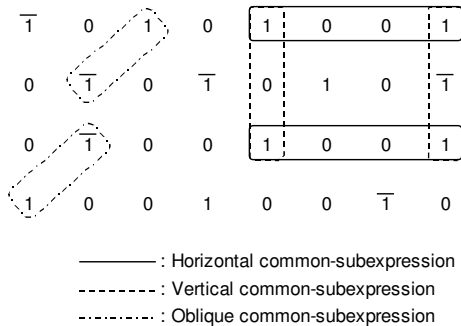


Fig. 2. Common subexpressions in the constant coefficients.

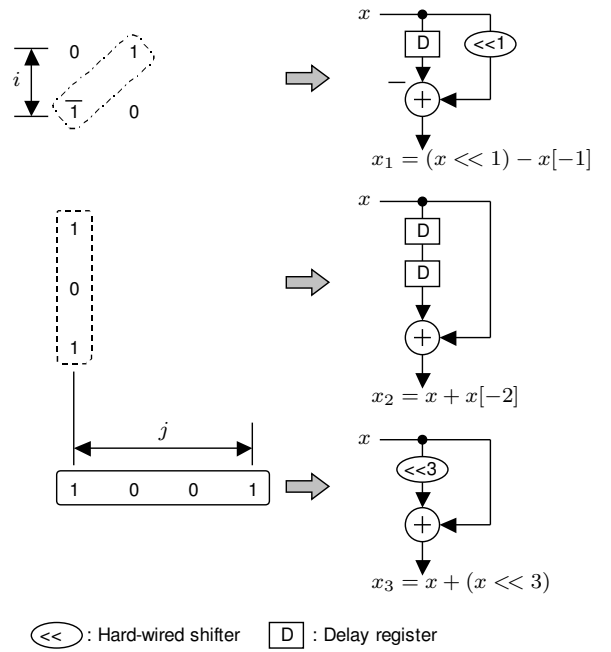


Fig. 3. Notations and circuit configurations of common subexpressions

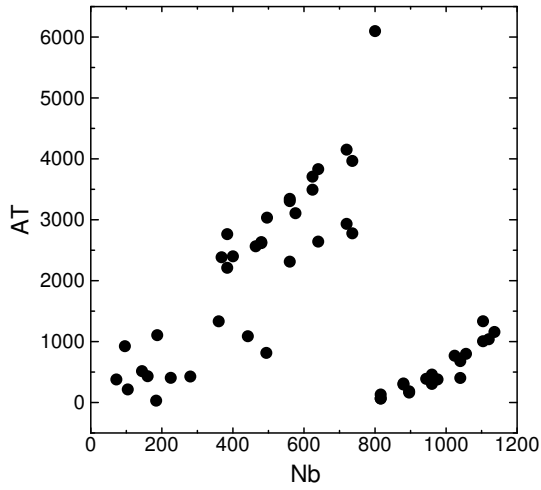
common subexpressions. We so assume that shift operations are essentially free, they can be hard-wired. However, the structures for vertical and oblique techniques are designed without any consideration of the number of registers. Vinod et al.'s CSE technique [6] has used combining horizontal and vertical common subexpression to reduce the number of adders. However, the structures generated by using vertical CSE technique are still designed without any consideration of the number of registers. Therefore, if the structure of MCM contains many registers, the implementation cost cannot be reduced. Our improved horizontal and vertical CSE technique [7] has been proposed an efficient way to find the correct bit-patterns for horizontal and vertical CSEs. The proposed 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. Using the proposed method, the MCM area of the FIR filters has been reduced by an average of 20%.

## IV. IMPLEMENTATION RESULTS AND DISCUSSIONS

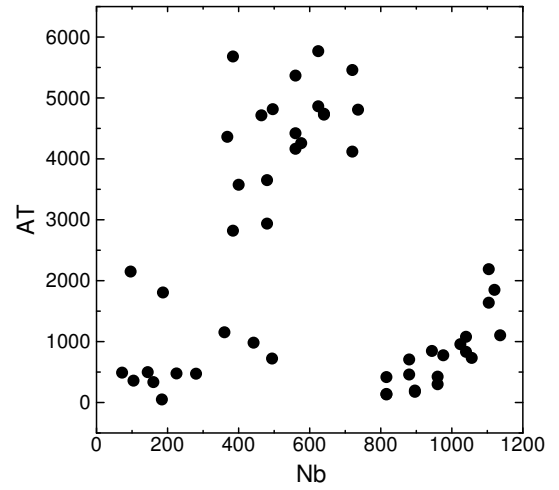
### A. Implementation Results

In order to confirm whether the area-time product ( $AT$ ) of constant multiplier are dependent on the number of coefficients ( $N$ ) or the size of wordlength ( $b$ ), we use the 52 examples from published papers during the two decades (e.g. [3], [4], [6], [7], [9]–[15]). These examples are included in the multiplierless FIR/IIR filters, filter banks, polyphase filters, DCT, DFT, and so on.

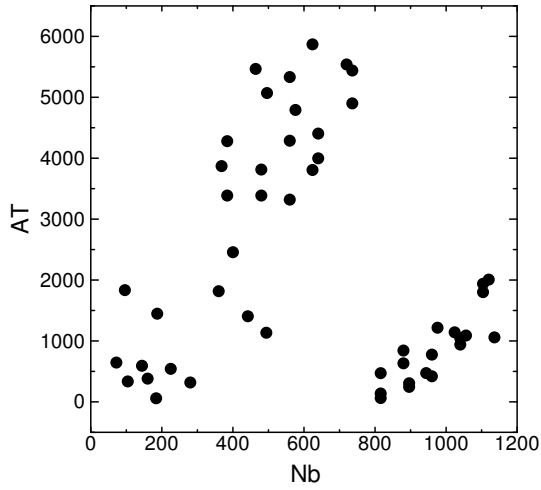
A testbed prepared for evaluating 52 examples is constructed with Xilinx XC4VLX15-11 (VirtexIV series) FPGA. The Xilinx XC4VLX15-11 FPGA consists of a  $64 \times 24$  array of



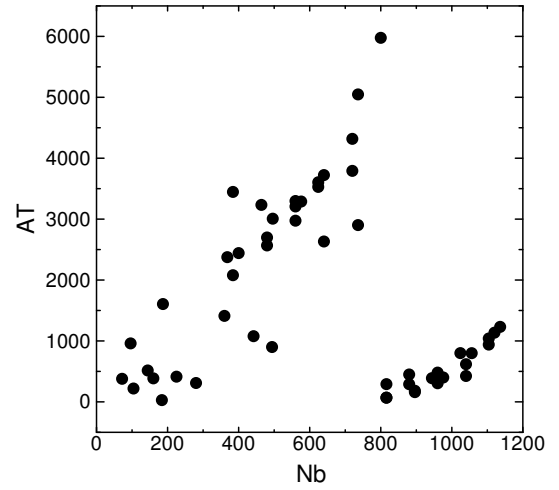
(a) Horizontal CSE.



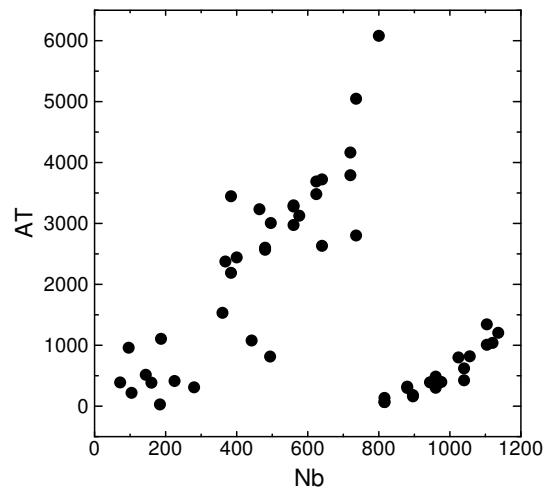
(b) Oblique (Hartley's) CSE.



(c) Vertical (Jang's) CSE.



(d) Horizontal and vertical (Vinod's) CSE.



(e) Efficient horizontal and vertical (Our proposed) CSE.

Fig. 4. Comparison of area-time product versus coefficients-wordlength product.

CLBs, each having two 4-input LUTs, two 4-input LUTs and four D flip-flops (D-FFs). The design examples are described by Verilog-HDL for implementation, and are also implemented on XC4VLX15-11 using Xilinx ISE Webpack 7.1i. In these implementations, the automatic placement and routing compilation option are used.

Figure 4 shows the distribution map of the five different CSEs. In these graphs,  $Nb$  is a multiplicand matrix size. The maps indicate that if  $AT$  is smaller and the variation in the plots is smaller, the CSE would be a good performance. From these results we found that horizontal, horizontal and vertical, efficient horizontal and vertical CSEs have especially a good performance in the matrix  $Nb$  of 800 and over.

### B. Discussions

In the implementations, we sampled research articles from 52 selected papers during the two decades, however we have unfortunately not found the matrix table of 1200 over. Further investigation is so required to assess the performance of these CSEs for the design of the MCM.

This paper also concerns only the area-time product performance. In a FPGA (or VLSI) implementation, it might be desirable to limit the logic depth. This can be achieved by including the logic depth, with an appropriate weighting, into the cost measure throughout the process. For large transform sizes this would require large area, but it would be capable of extremely high processing throughput. Therefore, we might need a new indicator function for factor analysis with application in MCM.

## V. CONCLUSION

This paper has been reported a comparison of hardware reductions achieved using the horizontal, vertical and oblique CSEs in realizing constant multipliers. From the FPGA implementation results included in 52 MCM examples, 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.

### ACKNOWLEDGMENT

The authors would like to thank Mr. M. Suzuoki, Mr. S. Sakakibara, and Mr. Y. Kamiya for their help and support in developing the Verilog-HDL script.

This work was supported, in part, by a grant from Ogawa Foundation for Science and Technology of Pacific Industrial Co., Ltd. and by a grant from Research Foundation for the Electrotechnology of Chubu.

## REFERENCES

- [1] M. Mehendale, S. D. Sherlekar, and G. Venkatesh, "Synthesis of multiplier-less FIR filters with minimum number of additions," in *Proc. 1995 IEEE/ACM Int. Conf. on Computer-Aided Design (ICCAD 1995)*, San Jose, CA, Nov. 5–9, 1995, pp. 668–671.
- [2] M. Potkonjak, M. B. Srivastava, and A. Chandrakasan, "Multiple constant multiplications: 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.
- [3] R. I. Hartley, "Optimization of canonic signed digit multipliers for filter design," in *Proc. 1991 IEEE Int. Symp. on Circuits and Systems (ISCAS 1991)*, Singapore, June 11–14, 1991, pp.1992–1995.
- [4] R. Paško, P. Schaumout, V. Derudder, S. Vernalde, and D. Đuračković, "A new algorithm for elimination of common subexpressions," *IEEE Trans. Computer-Aided Design*, vol. 18, no. 1, pp. 58–68, Jan. 1999.
- [5] Y. Jang and S. Yang, "Low-power CSD linear phase FIR filter structure using vertical common sub-expression," *Electron. Lett.*, vol. 38, no. 15, pp. 777–779, July 2002.
- [6] A. P. Vinod, E. M-K. Lai, A. B. Premkumar, and C. T. Lau, "FIR filter implementation by efficient sharing of horizontal and vertical common subexpressions," *Electron. Lett.*, vol. 39, no. 2, pp. 251–253, Jan. 2003.
- [7] Y. Takahashi and M. Yokoyama, "New cost-effective VLSI implementation of multiplierless FIR filter using common subexpression elimination," in *Proc. ISCAS 2005*, Kobe, Japan, May 23–26, 2005, pp. 845–848.
- [8] K. Suzuki, H. Ochi, and S. Kinjo, "A design of FIR filter using CSD with minimum number of registers," in *Proc. 1996 IEEE Asia Pacific Conf. on Circuits and Systems (APCCAS 1996)*, Seoul, Korea, Nov. 18–21, 1996, pp. 227–230.
- [9] H. Samuelli, "An improved search algorithm for the design of multiplierless FIR filters with powers-of-two coefficients," *IEEE Trans. Circuits Syst.*, vol. 36, no. 7, pp. 1044–1047, July 1989.
- [10] X. Xu and B. Nowrouzian, "Local search algorithm for the design of multiplierless digital filters with CSD multiplier coefficients," in *Proc. 1999 IEEE Canadian conf. on Electrical and Computer and Engineering (CCECE 1999)*, Edmonton, Canada, May 9–12, 1999, pp. 811–816.
- [11] K. A. Kotteri, A. E. Bell, and J. E. Carletta, "Design of multiplierless, high-performance, wavelet filter banks with image compression applications," *IEEE Trans. Circuits Syst. I*, vol. 51, no. 3, pp. 483–494, Mar. 2004.
- [12] C. Y. Yao, H. H. Chen, T. F. Lin, C. J. Chien, and C. T. Hsu, "A novel common-subexpression-elimination method for synthesizing fixed-point FIR filters," *IEEE Trans. Circuits Syst. I*, vol. 51, no. 11, pp. 2215–2221, Nov. 2004.
- [13] A. P. Vinod and E. M-K. Lai, "Comparison of the horizontal and the vertical common subexpression elimination methods for realizing digital filters," in *Proc. ISCAS 2005*, pp. 496–498.
- [14] Y. Takahashi, T. Sekine, and M. Yokoyama, "70 MHz multiplierless FIR Hilbert transformer in 0.35  $\mu\text{m}$  standard CMOS library," *IEICE Trans. Fundamentals*, vol. E90-A, no. 7, pp. 1376–1383, July 2007.
- [15] N. Banerjee, J. H. Choi, and K. Roy, "A process variation aware low power synthesis methodology for fixed-point FIR filters," in *Proc. 2007 ACM/IEEE Int. Symp. on Low Power Design (ISLPED 2007)*, Portland, OR, Aug. 27–29, 2007, pp. 147–152.