DESIGN & IMPLEMENTATION OF FIXED WIDTH MODIFIED BOOTH MULTIPLIER

[1]SAROJ P. SAHU, [2]RASHMI KEOTE

[1]M.tech IVth Sem(Electronics Engg.), [2]Assistant Professor,Yeshwantrao Chavan College of Engineering, Nagpur,
Email; saroj.sahu3990@gmail.com , rashmikeote@gmail.com

Abstract— Multiplication is the main operation in many signal processing algorithms. High accuracy and low power dissipation are the most important objectives in many multimedia and lossy applications such as filtering, convolution, Euclidean distance, fast Fourier transform (FFT).The fixed width multipliers are used to maintain a fixed format and allow a little accuracy loss of output data.-In this paper for the reduction of truncation errors modifies the partial product matrix and derive an error compensation function. A simple compensation circuit mainly composed of the simplified sorting network is also proposed. The proposed error compensation circuits offer reduction in mean square error over the previous circuits.

Keyword: Fixed width multiplier, Error compensation circuit, Mean error, Mean square error, Modified booth multiplier

I. INTRODUCTION

Multiplier is the basic components in many multimedia and digital signal processing (DSP) chips because it enhances the chip's power, performance and area. To obtain a higher speed, parallel multipliers are always preferred at the cost of high area complexity. In the past, many multiplication algorithms (architectures) have been proposed to reduce the power, area and increase the speed of the multipliers. The modified Booth algorithm is widely used to implement multiplication in DSP systems and other applications.

As per the Literature Survey on booth multiplier for lossy application several error compensation approaches have been proposed to effectively reduce the truncation error of fixed-width modified Booth multipliers.

In [1], a high-accuracy error compensation circuit for the fixed-width modified Booth multiplier is proposed. First slightly modify the partial product matrix of Booth multiplication to reduce the partial product bits in the truncated portion of DTFM. Then, the correlation between Booth encoded outputs and the truncated product error of DTFM is analyzed and explored to derive an effective and simple error compensation function. Finally, a simple compensation circuit composed of a simplified sorting network and some adder cells is developed according to the proposed error compensation function.

In [2], A low-error reduced-width Booth multiplier using a proper compensation vector is proposed. The compensation vector is dependent on the input data. The compensation value was generated by using statistical analysis and linear regression analysis. This approach can significantly decrease the mean error of fixed-width modified Booth multipliers, but the maximum absolute error and the mean-square error are still large.

In [3], paper presents an error compensation method for a modified Booth fixed-width multiplier that receives a –bit input and produces a -bit product. The truncated bits are divided into two groups that is major group and minor group. To obtain better error performance with a simple error compensation circuit, Booth encoded outputs are utilized to generate the error compensation value.

In [4], a systematic design methodology for the low-error fixed-width modified Booth multiplier via exploring the influence of various indices in a binary threshold was developed to decrease the product error.

The fixed-width modified Booth multipliers in [3] and [4] achieve better error performance in terms of the maximum absolute error and the mean-square error when compared with the previous published multiplier in [2]. However, their mean errors are much larger than that of [2].

The main objective of this paper is to design modified Booth multiplier for lossy application that is evaluate performance in terms of Accuracy, delay & hence the Power. The accuracy will be achieved by minimizing the truncation error by reducing Partial Products of the multiplication. The mean error & mean square errors are reduce by using a high-accuracy error compensation circuit for the fixed-width modified Booth multiplier. The circuit makes the error distribution not only be symmetric to but also centralize in zero error as much as possible. Therefore, the mean and mean-square errors can be significantly reduced. After achieving this goal the proposed modified multiplier will be applied to the lossy application like filter designing.

II. FUNDAMENTAL OF MODIFIED BOOTH MULTIPLIER

Let us consider multiplication operation of two n-bit signed numbers say A and B, where A is n-bit

multiplicand and B is n-bit multiplier, which is given below:

$$A = a_{n-1}a_{n-2}\cdots a_0 \quad (1)$$

$$B = b_{n-1}b_{n-2}\cdots b_0 \quad (2)$$

The two's complement representations of A and B can be expressed as given below:

$$A = -a_{n-1}2^{n-1} + \sum_{i=0}^{n-2} a_i 2^i, \quad B = -b_{n-1}2^{n-1} + \sum_{i=0}^{n-2}$$

The modified Booth encoding truth table is shown in Table I.

b_{2i+1}	b_{2i}	b_{2i-1}	Operation	neg_i	two_i	one_i	$zero_i$	cor_i
0	0	0	+0	0	0	0	1	0
0	0	1	+A	0	0	1	0	0
0	1	0	+A	0	0	1	0	0
0	1	1	+2A	0	1	0	0	0
1	0	0	-2A	1	1	0	0	1
1	0	1	-A	1	0	1	0	1
1	1	0	-A	1	0	1	0	1
1	1	1	-0	1	0	0	1	0

TABLE I: MODIFIED BOOTH ENCODING TABLE

The table shows the modified booth encoding truth table, where b_{2i+1}, b_{2i}, b_{2i-1} represents multiplier's bit and A represents multiplicand. This modified Booth encoding which groups the bits of the multiplier into triplets. The modified booth encoder is having five outputs. They are $Zero_i$, One_i, Two_i, Neg_i and Cor_i. Neg_i is the negation of the operation. Cor_i indicates that the partial product row is positive or negative.

By modified Booth encoding which groups the bits of the multiplier into triplets, B can be expressed as:

$$B = \sum_{i=0}^{n/2-1} M_i 2^{2i} = \sum_{i=0}^{n/2-1} (-2b_{2i+1} + b_{2i} + b_{2i-1})2^{2i} \quad (4)$$

Where $b_{-1} = 0$ and Mi = {-2,-1, 0, 1, 2}. Depending on the encoded result shown in Table I, the Booth encoder and partial product generation circuit proposed in [4] (depicted in Fig. 1(a) and 1(b), respectively) are adopted to choose one of multiple multiplicands -2A,-A , 0, A and 2A for generating each partial product row PP_i , where $0 \le i \ge n/2-1$ and aj bar is the complement of a_j. The 2A in Table I is realized by left shifting A one bit. As for the negation operation, each bit of A is inverted and an extra binary value "1" is added to the least significant bit of next partial product row. Adding "1" can be implemented as a correction bit cor_i which indicates that the partial product row is positive or negative. If cor_i=0 then partial product row is positive and if cor_i=1 then partial product row is negative. The sign bit for each partial product row PP_i must be properly extended up to the $(2n-1)^{th}$ bit position because each partial product row is represented in two's complementation. In the proposed low power modified Booth multiplier, only the partial product bits in LP_{minor} are removed and the carry value propagated from LPminor to LP_{major} must be estimated by a simple circuit to compensate for the truncation error.

III. PROPOSED FIXED WIDTH MODIFIED BOOTH MULTIPLIER

In the proposed modified booth multiplier, the values of the partial product bits are dependent on the outputs of the booth encoder shown in the booth encoder table. So we first obtain the relation between the outputs of Booth encoders and the carry value propagated from LP_{minor} to LP_{major}. After finding this relation we designed partial product generation circuit as shown in the fig (1).

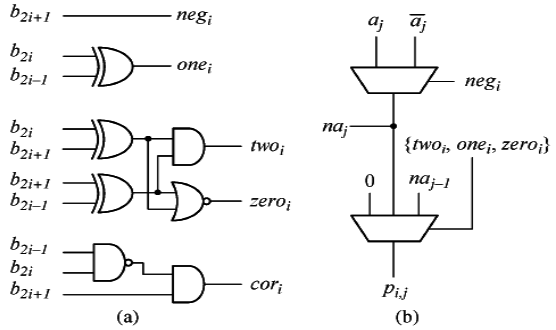

Figure 1. (a) Modified Booth encoder. (b) Partial product generation circuit.

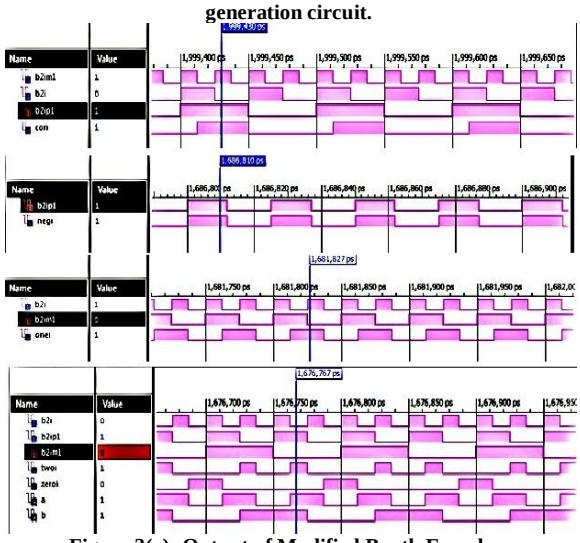

Figure 2(a): Output of Modified Booth Encoder.

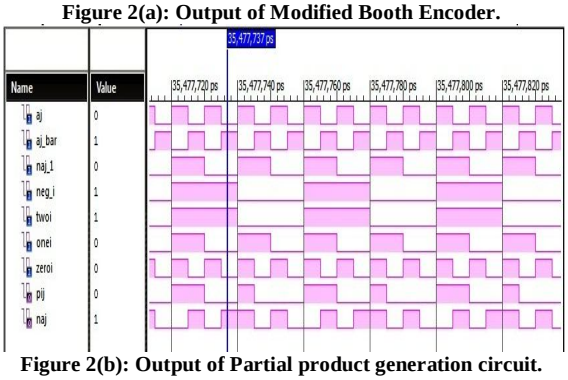

Figure 2(b): Output of Partial product generation circuit.

Then a simple error compensation function is obtained. Error compensation function will take the outputs of booth encoder as inputs and then generates the approximate carry value. This is derived for reducing the truncation error and makes the error distribution as symmetric and centralized as possible. Finally, a simplified and fast compensation circuit is constructed to form a low power, high-accuracy fixed-width multiplier.

A : Proposed Error Compensation Function

The various error compensation methods are proposed to reduce truncation error. There are two schemes to produce error compensation value, first is the constant scheme and second is adaptive scheme. The constant scheme [5] pre-computes the constant error compensation value and then supplies them to the carry inputs of the retained adder cells. This scheme is simple but truncation error is relatively large. The second scheme is adaptive scheme [6]–[8], in which it adaptively adjust the compensation value according to the input data at the cost of a little higher hardware complexity. Higher accuracy can be obtained by this scheme than the constant scheme.

Let SUM (MP) and SUM (LP) represent the sum of partial product bits in MP and LP, respectively, then the 2n-bit output product P can be expressed as:

$$P = A \times B = SUM(MP) + SUM(LP)$$

The new matrix is obtained by adding the least significant bit $P_{n/2-1}$, 0 of $PP_{n/2-1}$ and $cor_{n/2-1}$ in advance to generate a sum $E_{n/2-1}$,0 and a carry lambda at the (n-2)th and(n-1)th bit positions, respectively. The proposed 8 × 8 modified Booth partial product matrix is shown below:

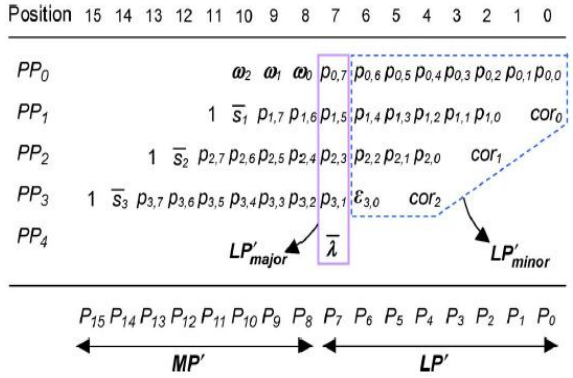

Figure 3. The proposed 8 × 8 modified Booth partial product matrix.

The weight of the extra "1" located at the (n-1)th bit position of the PTM is equal to the weight of lambda, they can be added up to generate a sum $\overline{\lambda}$ (i.e., the complement of λ) and a carry lambda which must be propagated to the nth bit position. Then, this carry can be incorporated with the sign extension bits s0bar s0s0 of PP0 to produce the new partial product bits w_2, w1, w0.

$$\overline{\lambda} = \overline{[(a_0 \wedge one_{n/2-1}) \wedge \overline{zero_{n/2-1}} \wedge b_{n-1}]} \quad (5)$$

$$\omega_2 = \overline{(s_0 \wedge \overline{\lambda})} \quad (6)$$

$$\omega_1 = \overline{\omega_2} \quad (7)$$

$$\omega_0 = \overline{[(s_0 \vee \overline{\lambda}) \wedge \omega_2]} \quad (8)$$

The corresponding circuit for generating $\overline{\lambda}$ w_2,w_1 and w_0 is depicted in Fig. 6.

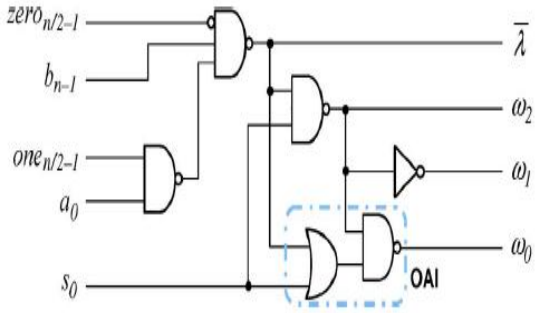

Figure 4. The circuit to produce $\overline{\lambda}$ and w_2, w1, w0.

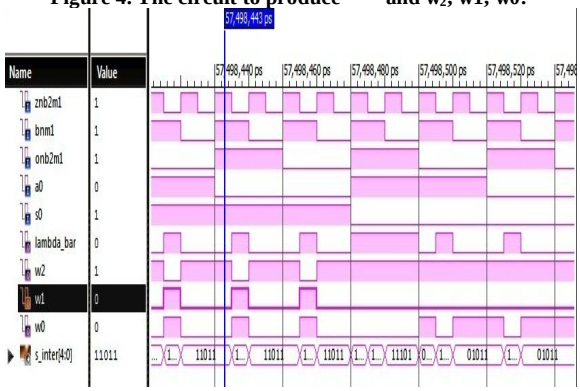

Figure 4(a): Output of error compensation function.

B. Proposed Low Error Compensation Circuit

We consider that zeroi for $0 \leq i \geq n/2-1$ can be sorted and the sorted outputs are p_j for $0 \leq j \geq n/2-1$. Moreover, if the largest bits (i.e., bits equal to "1") are gathered to the less significant positions, then $a_k = p_{2k}$ for $0 \leq k \geq m$. where m= [(n/2-1)/2], that is the output of SC-generator. There are two types of comparison-based sorting networks [9]. First is the bitonic and the second is odd-even merge sorting networks, which is suited to hardware implementation. The odd-even merge sorter has the same number of compare levels as the bitonic sorter but requires lesser comparators, thus we adopt and simplify the odd-even merge sorting network to realize the SC-generator. Figs. 7(a) and 9(a) shows the odd-even merge sorting networks for n=8 and n =16 respectively. These sorting networks are composed of appropriately connected comparators. Each comparator takes in two input bits and either passes them directly or switches them. With inputs a and b, the outputs max(a, b) and min(a, b). In addition, the sorting networks can be further simplified by using NAND, NOR, AND-OR INVERTER (AOI), and OR-AND-INVERTER (OAI) gates as shown in Figs. 7(b) and 9(b) for n=8 and n=16, respectively. The SC-generator for

different n can be constructed in a similar fashion. After the estimated carries α_1, $\alpha_{1..}$ and α_m are generated by SC-generator, they are fed into the to produce the final fixed-width product.

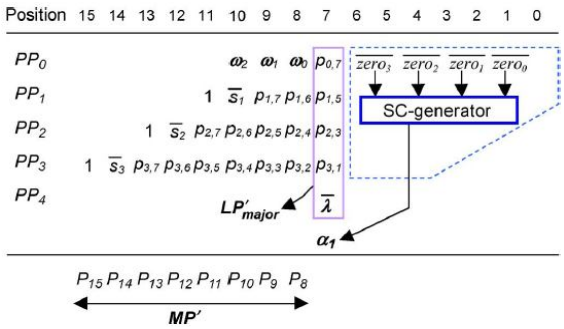

Figure 5: Final partial product matrix of proposed fixed-width modified Booth multiplier for n=8.

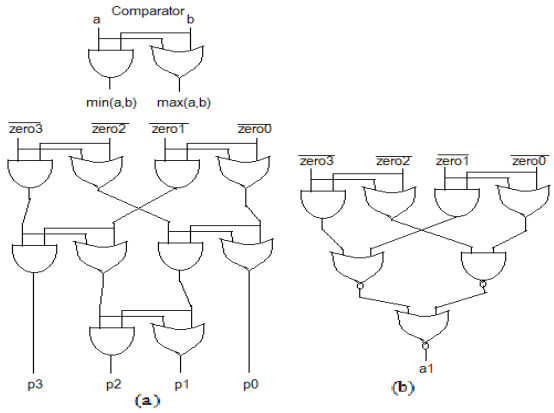

Figure 6 (a) Odd-even merge sorting network for n=8. (b) Proposed SC-generator for n=8.

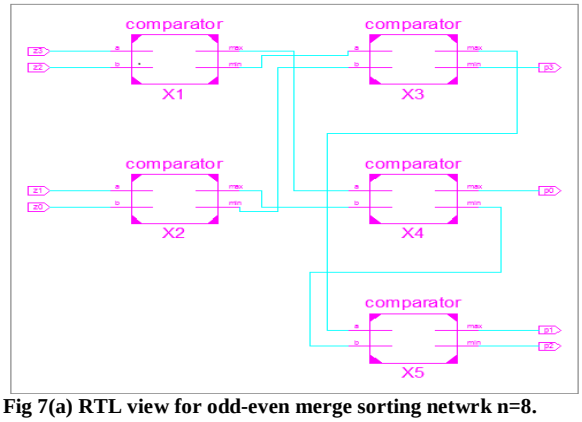

Fig 7(a) RTL view for odd-even merge sorting netwrk n=8.

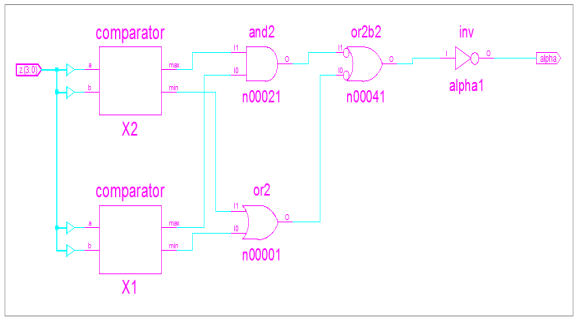

Fig 7(b) RTL view for SC-generator n=8.

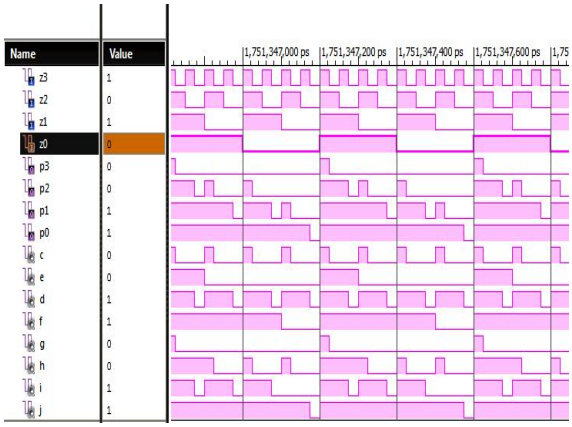

Figure 8(a): Output of Odd-even merge sorting network for n=8

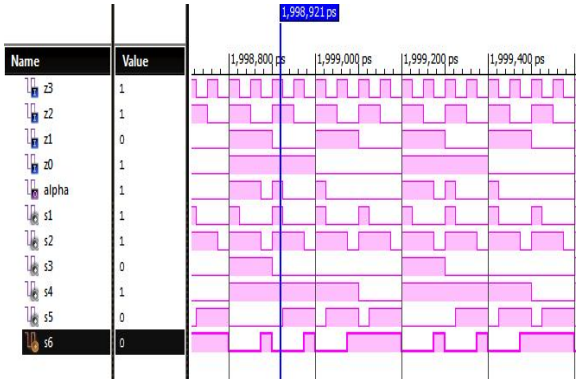

Figure 8(b): Output of Proposed SC-generator for n=8.

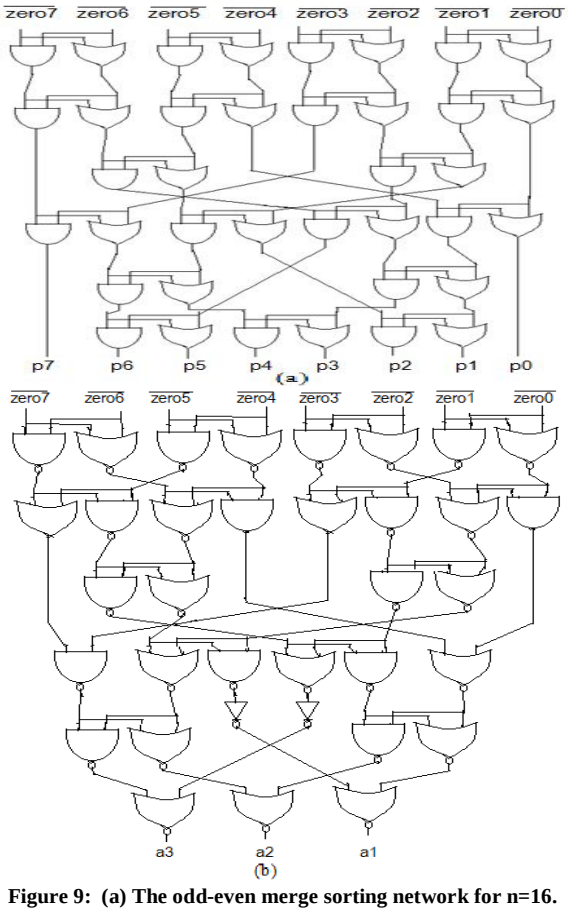

Figure 9: (a) The odd-even merge sorting network for n=16. (b) The proposed SC-generator for n=16.

CONCLUSION

This paper develops a more generalized methodology for designing a family of low-power area-efficient fixed-width multipliers. In the proposed multiplier, modified the partial product matrix of Booth multiplication and proposed error compensation function. This compensation function makes the error distribution be more symmetric to and centralized in the error equal to zero, leading the fixed-width modified Booth multiplier to very small mean and mean-square errors. Also a sorting network designed to realize the compensation function. The proposed error compensation circuits offer reduction in mean square error over the previous circuits. As a expected outcome, the proposed multipliers are applicable to lossy applications to reduce the area and power consumption of the whole system.

REFERENCES

[1] Jiun-Ping Wang, Shiann-Rong Kuang, Member, IEEE, and Shish-Chang Liang, "High-Accuracy Fixed-Width Modified Booth Multipliers for Lossy Applications" IEEE transaction on VLSI system, VOL. 19, NO. 1, JANUARY 2011

[2] S. J. Jou, M.-H. Tsai, and Y.-L. Tsao, "Low-error reduced-width Booth multipliers for DSP applications," IEEE Trans. Circuits Syst. I, Fudam. Theory Appl., vol. 50, no. 11, pp. 1470–1474, Nov. 2003.

[3] K.-J. Cho, K.-C. Lee, J.-G. Chung, and K. K. Parhi, "Design of low error fixed-width modified Booth multiplier," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 12, no. 5, pp. 522–531, May 2004.

[4] M.-A. Song, L.-D. Van, and S.-Y. Kuo, "Adaptive low-error fixed width Booth multipliers," IEICE Trans. Fundamentals, vol. E90-A, no. 6, pp. 1180–1187, Jun. 2007.

[5] Y. C. Lim, "Single precision multiplier with reduced circuit complexity for signal processing applications," IEEE Trans. Computer, vol. 41, no. 10, pp. 1333–1336, Oct. 1992.

[6] J. M. Jou, S. R. Kuang, and R. D. Chen, "Design of low-error fixed width multipliers for DSP applications,"IEEE Trans. Circuits Syst. I, Exp. Briefs, vol. 46, no. 6, pp. 836–842, June 1999.

[7] L. D. Van and C. C. Yang, "Generalized low-error area-efficient fixed width multipliers," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 52, no. 8, pp. 1608–1619, Aug. 2005.

[8] A. G. M. strollo, N. Petra, and D. D. Caro, "Dual-tree error compensation for high performance fixed-width multipliers," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 52, no. 8, pp. 501–507, Aug. 2005.

[9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, 2nd ed. Cambridge, MA: MIT Press, 1990.

[10] A. A. Farooqui and V. G. Oklobdzija, "General data-path organization of a MAC unit for VLSI implementation of DSP processors," in Proc. IEEE Int. Symp. Circuits Syst., 1998, vol. 2, pp. 260–263.

★★★