Design of Approximate Multiplier using 5:2 Compressors

B.Sree mangamma[1], M.Madhu Babu[2]
[1]*M.Tech Student, Dept. of Electronics and Communication Engineering, JNTUACEA, AP, INDIA*
[2]*Dept. of Electronics and Communication Engineering, JNTUCEA, AP, INDIA*

***Abstract*— To design effective approximate multiplier we use approximate compressors. By employing these compressors in the Dadda multiplier structure provides better power and speed. In this paper we proposed a approximate compressors with reduced area, delay and power with comparable accuracy when compared with the existing architectures. So that a total of two different 4:2 approximate compressors are analyzed and one 5:2 compressor proposed. The proposed compressors are utilized to implement 8*8 and 16*16 Dadda multiplier. The analysis is further extended to project the application of the implemented design in error resilient applications like image smoothing and image multiplication.**

***Index Term* - Approximate 5:2 Compressor, approximate computing, speed, power, area.**

I. INTRODUCTION

In digital design approximate computing plays a major role. It enhances the performance of Area, power and speed. Multiplier has three steps: partial product generation, partial product reduction and carry propagation addition. Approximation can be introduced in any of these blocks. By reducing the partial products we can increase the power dissipation and delay of the system. In the place of partial products we use full adders, half adders in earlier researches. We replace compressors in the place full adders and half adders which are used to reduce the delay associated with partial products. We can introduce approximations in any of these blocks. A Compressor is a logic circuit which counts the number of 1's at the input. Higher order compressors are also used some of those are 4:2 compressors or 5:2 compressors. In these compressors X-OR circuits are used. Speed of operation which is inversely proportional to the delay of the system. In order to maintain the balance between delay, area and power, approximate computing has emerged as a promising solution. Approximation in arithmetic operations results in faster systems with lesser design complexity and power consumption. There is a lot of research has been carried out to increase the efficiency of approximate multiplication. In the multiplication operation, partial product summation has in-arguably been the major contributor to the power dissipation and delay in the system. Compressors estimate the count of logic 1 in the input using half adders and/or full adders. There are different topologies that are used for compressors are 7: 3, 5:2, 4: 2, and 3: 2.

II. APPROXIMATE MULTIPLIERS

Approximate multipliers are widely being used for energy-efficient computing in applications that exhibit an inherent tolerance to inaccuracy. The accuracy is the major design parameter for any multiplier. Apart from area and power, it is difficult to identify the exact approximate multiplier. Multiplication is undoubtedly a performance determining operation in Artificial Intelligence and DSP applications. These applications demand high speed multiplier architectures to necessitate high speed parallel operations. By using the approximation in multipliers, they make fast calculations with minimized hardware complexity, delay and power. By having accuracy at certain levels. Compressors calculate the sum and carry at each level simultaneously. The carry is added to the higher significant sum bit in the next stage. This is followed till the generation of final product. The approximate multipliers can provide the results which are almost near to the exact multipliers. So we can these are better in terms of accuracy, power and area when compared to the exact multipliers.

III. EXISTED SYSTEM

In this paper, we focus on three factors that are suitable for selecting the approximate multiplier.

(1) the type of approximate full adder (FA) used to construct the multiplier,
(2) the architecture, i.e., array or tree, of the multiplier and
(3) the placement of sub-modules of approximate and exact multipliers in the main multiplier module. Based on these factors, we explored the design space for circuit level implementations of approximate multipliers. We used circuit level implementations of some of the most widely used approximate full adders, i.e., approximate mirror adders, XOR/XNOR based approximate full adders and Inexact adder cell. These FA cells are then used to develop circuits for the approximate high order compressors as building blocks for 8x8 array and tree multipliers. We then develop various implementations of higher bit multipliers by using a combination of exact and inaccurate 8x8 multiplier cells.. The design space of these multipliers is explored based on their power, area, delay and error and the best approximate multiplier designs are identified.

A. EXACT 4:2 COMPRESSORS

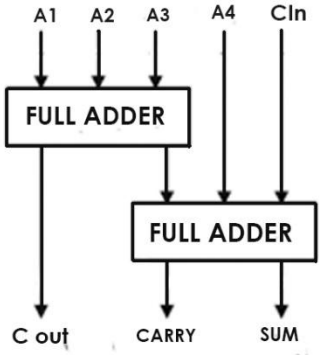

Figure 1. Exact 4:2 compressor

The general block diagram of an exact 4:2 compressor is shown in Figure 1. It consists of five inputs, three outputs and two cascaded full adders. A1, A2, A3, A4 and CIN are the inputs and COUT, CARRY and SUM are the outputs of the exact 4:2 compressor. COUT, CARRY and SUM are given as

$$COUT = A3(A1 \oplus A2) + A1(A1 \oplus A2) \quad (1)$$

$$CARRY = CIN\,(A1 \oplus A2 \oplus A3 \oplus A4) + A4(A1 \oplus A2 \oplus A3 \oplus A4) \quad (2)$$

$$SUM = CIN \oplus A1 \oplus A2 \oplus A3 \oplus A4 \quad (3)$$

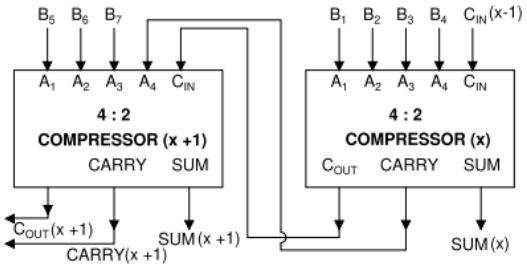

Figure 2. Compressor chain.

Figure 2 represents the compressor chain. Where CIN is the input carry from the previous 4:2 compressor that moved to lower significant bits CARRY and COUT.

TABLE 1. Truth table for exact 4:2 compressor.

A_1	A_2	A_3	A_4	C_{IN}	C_{OUT}	$CARRY$	SUM
0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1
0	0	0	1	0	0	0	1
0	0	0	1	1	0	1	0
0	0	1	0	0	0	0	1
0	0	1	0	1	0	1	0
0	0	1	1	0	0	1	0
0	0	1	1	1	0	1	1
0	1	0	0	0	0	0	1
0	1	0	0	1	0	1	0
0	1	0	1	0	0	1	0
0	1	0	1	1	0	1	1
0	1	1	0	0	1	0	0
0	1	1	0	1	1	0	1
0	1	1	1	0	1	0	1
0	1	1	1	1	1	1	0
1	0	0	0	0	0	0	1
1	0	0	0	1	0	1	0
1	0	0	1	0	0	1	0
1	0	0	1	1	0	1	1
1	0	1	0	0	1	0	0
1	0	1	0	1	1	0	1
1	0	1	1	0	1	0	1
1	0	1	1	1	1	1	0
1	1	0	0	0	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	0	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	0	1	0	1
1	1	1	0	1	1	1	0
1	1	1	1	0	1	1	0
1	1	1	1	1	1	1	1

the outputs of order ‘1’ with higher significance than the input CIN . Table 1 represents the truth table for theexactcompressor.

B. APPROXIMATE 4:2 COMPRESSOR

On applying approximation to 4:2 compressors, output can be reduced to two partial products at last stage. Approximation is done by eliminating cout. The error occurs only at the input combination with a value of '1111'. Where the carry is '11'and sum is '11'an error of -1 is introduced. An 8*8 multiplication operation using approximate 4:2 compressors is shown in figure 3.

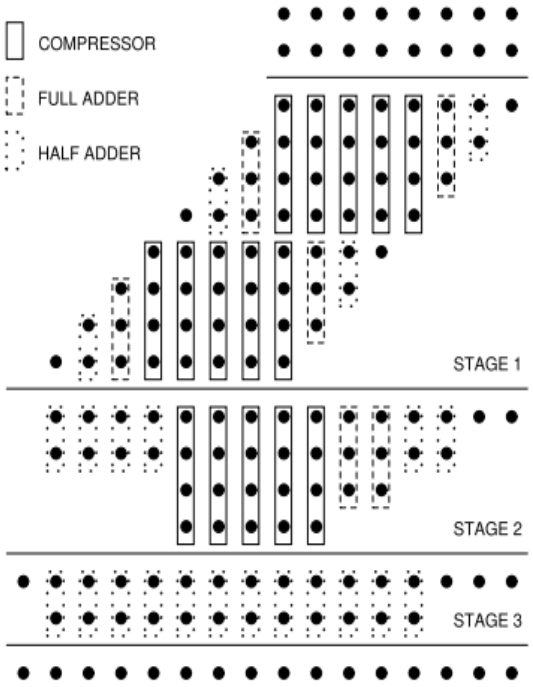

Figure 3. 8*8 Approximate multiplier

Level 1 in stage has $A_0=P_{p0.0}$ does not involve any operation. A half adder is required to generate A1 in level 2. The carry bit from half adder is passed onto Stage 2. Starting from Level 3, the number of terms in partial product array increases to 4 or more and reduces to 1 as the level increases. At this point, a 4 : 2 compressor facilitates fast partial product summation using approximation.

We have three stages in 8*8 Approximate multiplier. As because the input bit size of the multiplier is 8 bits. If we increase the bit size to 16 bits then the stages of the Approximate multiplier is also increases. So the number of stages in an Approximate Multiplier is inversely proportional to the bit size of an multiplier.

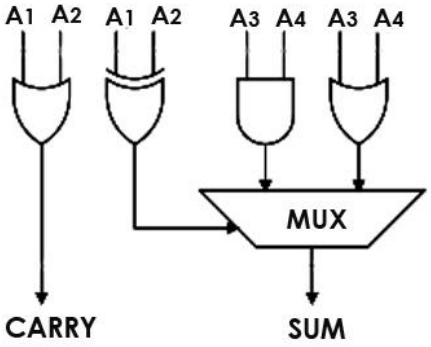

Figure 4: Approximate 4:2 compressor

The figure 3 represents the approximate 4:2 compressor having two OR gates, one gate, and one EX-OR gate. The mux circuit is used to generate the sum value and OR gate is used to generate carry value.

The existed system consists of 8*8 multiplier designed with approximate 4:2 compressors in the partial products. The figure 4 represents the 8*8 Approximate Multiplier. In partial product reduction we use half adder circuit for two inputs and full adder circuit for three inputs. For four input reduction we use 4:2 compressor in the partial product reduction.

Table 2: Truth table for approximate 4:2 compressor

A_1	A_2	A_3	A_4	$CARRY$	SUM
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	1
0	0	1	1	**0**	**1**
0	1	0	0	**1**	**0**
0	1	0	1	1	0
0	1	1	0	1	0
0	1	1	1	1	1
1	0	0	0	**1**	**0**
1	0	0	1	1	0
1	0	1	0	1	0
1	0	1	1	1	1
1	1	0	0	1	0
1	1	0	1	1	1
1	1	1	0	1	1
1	1	1	1	**1**	**1**

IV. PROPOSED HIGH SPEED APPROXIMATE 5:2 COMPRESSOR

A new 5:2 compressor design is proposed to reduce space, delay, and power consumption in multipliers for partial product accumulation.

A. EXACT 5:2 COMPRESSOR

The general block diagram of an exact 5: 2 compressor is shown in figure 5. It has five inputs, two outputs, and three full adders are cascaded. The terms COUT, CARRY, and SUM are defined as

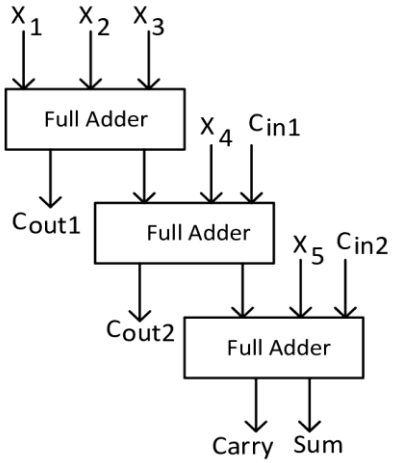

Figure 5: Exact 5:2 compressor

$$COUT = C1\,(X1 \oplus X2 \oplus X3)\,(X4 \oplus X5 \oplus CIN1) + C2\,(X1 \oplus X2 \oplus X3)\,(X4 \oplus X5 \oplus CIN2) + C1C2 \quad (1)$$

$$CARRY = ((X1 \oplus X2 \oplus X3)\,(X4 \oplus X5 \oplus CIN)) \oplus C1 \oplus C2 \quad (2)$$

$$SUM = CIN \oplus X1 \oplus X2 \oplus X3 \oplus X4 \oplus X5 \quad (3)$$

Figure 6 represents the proposed high-speed 5: 2 approximate compressor. A1, A2, A3, A4, and A5 are the compressor inputs, whereas CARRY and SUM are the outputs. SUM is generated using a multiplexer (MUX)-based design technique. The XOR gate's output serves as the MUX's select line. (A3A4A5) is selected when the choose line is high, while (A3 + A4 + A5) is selected when it is low. The suggested 5: 2 compressor can simplify carry generation logic to an OR gate by inserting an error with error distance 1 in the truth table of the precise compressor.

B. APPROXIMATE 5:2 COMPRESSOR

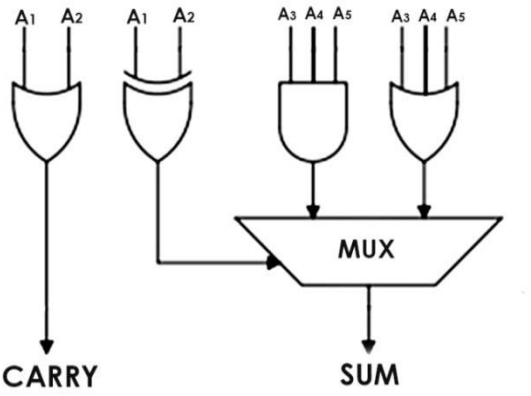

Figure 6 : Approximate 5:2 compressor

The following are the logical formulations for realising SUM and CARRY.

SUM = (A1⊕A2) A3A4A5+ (A1+A2) (A3+A4+A5) (4)

CARRY = (A1+A2) (5)

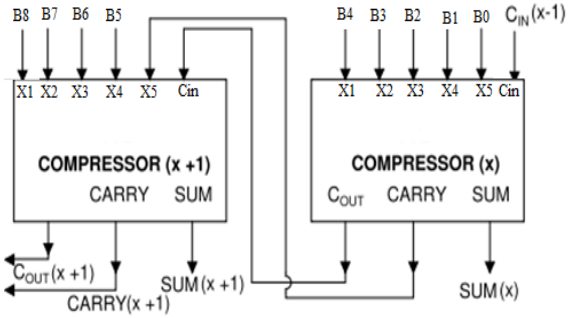

Figure 7. Approximate 5:2 compressor chain

In 5:2 compressor chain, Where CIN is the input carry from the previous 5:2 compressor that moved to lower significant bits CARRY and COUT to the next compressor. The inaccuracy is eliminated by cascading the compressor in multiples of 2.

C. 16*16 DADDA MULTIPLIER USING PROPOSED 5: 2 COMPRESSORS

A multiplier is typically composed of three components. In the first portion, AND gates are utilized to produce partial products. In the second section, compressors are utilized to reduce the maximum height of PPM (partial product matrix).

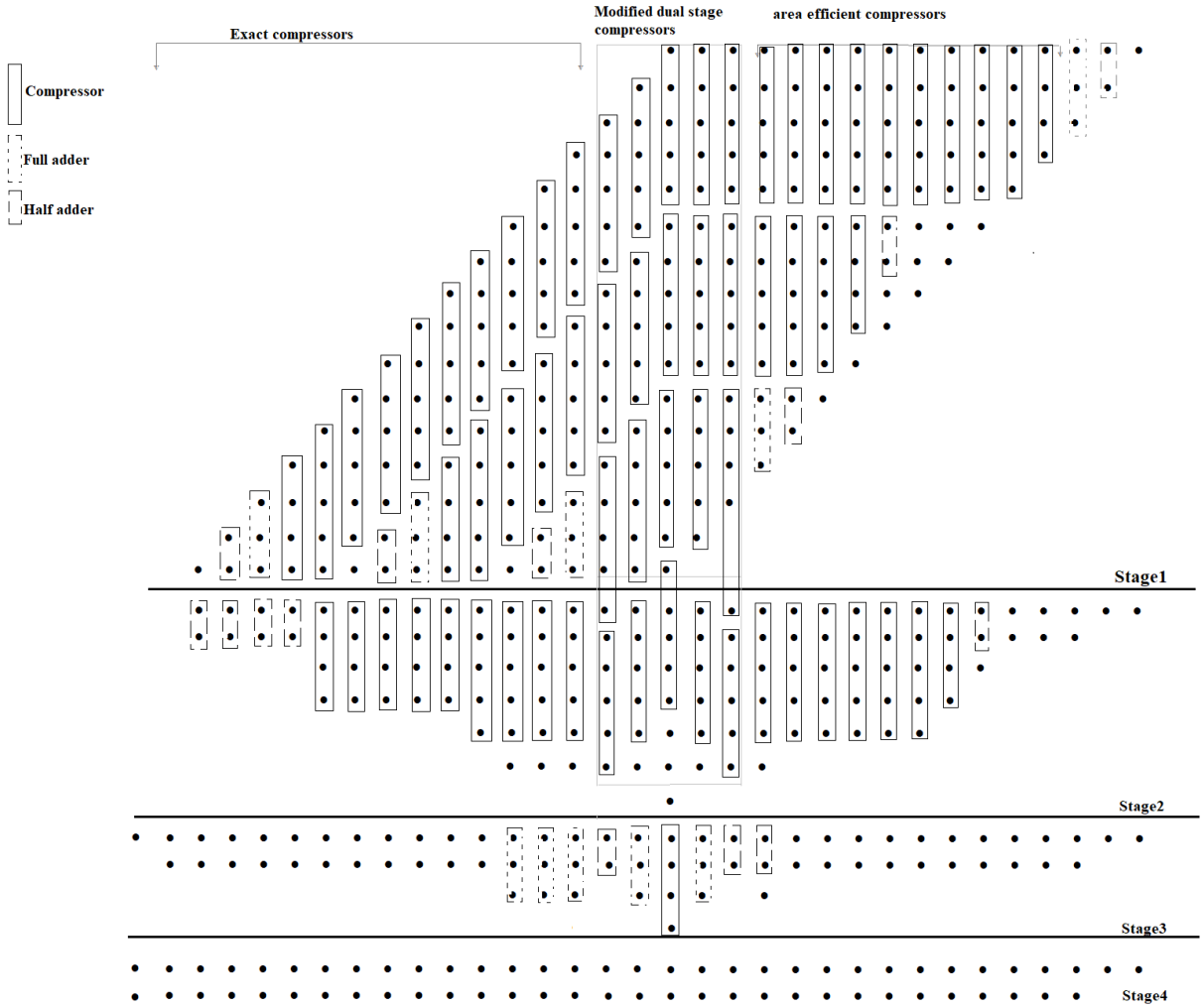

Figure 8. Proposed 5:2 compressors used in 16*16 multiplier.

In the third step, a carry propagation adder is used to obtain the final output. As a result, The PPM reduction circuitry is major factor for the multiplier's design complexity Design [1-6] focuses on optimizing the PPM reduction circuitry. In this section, we provide a 16*16 multiplier design. Weights are classified into three categories based on their significance: larger significance weights, medium significance weights, and lower significance weights. It should be noted that the designers can adjust the number of higher significance weights, intermediate significance weights, and lower significance weights for the trade-off between power consumption and computation accuracy.

Our PPM reduction circuitry employs an importance-driven logic compression technique to reduce power consumption with a little error: The higher weights utilize exact (5:2) compressors, the middle weights use two stage approximate (5:2) compressors, while the lower weights use inaccurate (5:2) area efficient compressors (OR-tree based approximation). Significance weights are considered in the second and third stages. Each weight has at most two product phrases when the second and third phases are finished. As a consequence, the final output may be produced using a carry propagation adder.

V. SIMULATION RESULTS

This section presents the analysis of the proposed 5:2 compressor architectures and, 16 × 16 Dadda multiplier designed with the proposed compressors. The analysis is carried out to determine the efficiency of the proposed design, The proposed 5:2 compressor design is compared with accurate 5: 2 compressor.

The designs are implemented using Xilinx ISE 14.7. The simulations of compressor designs are carried out with a supply voltage of 1 V at 1 GHz operating frequency.

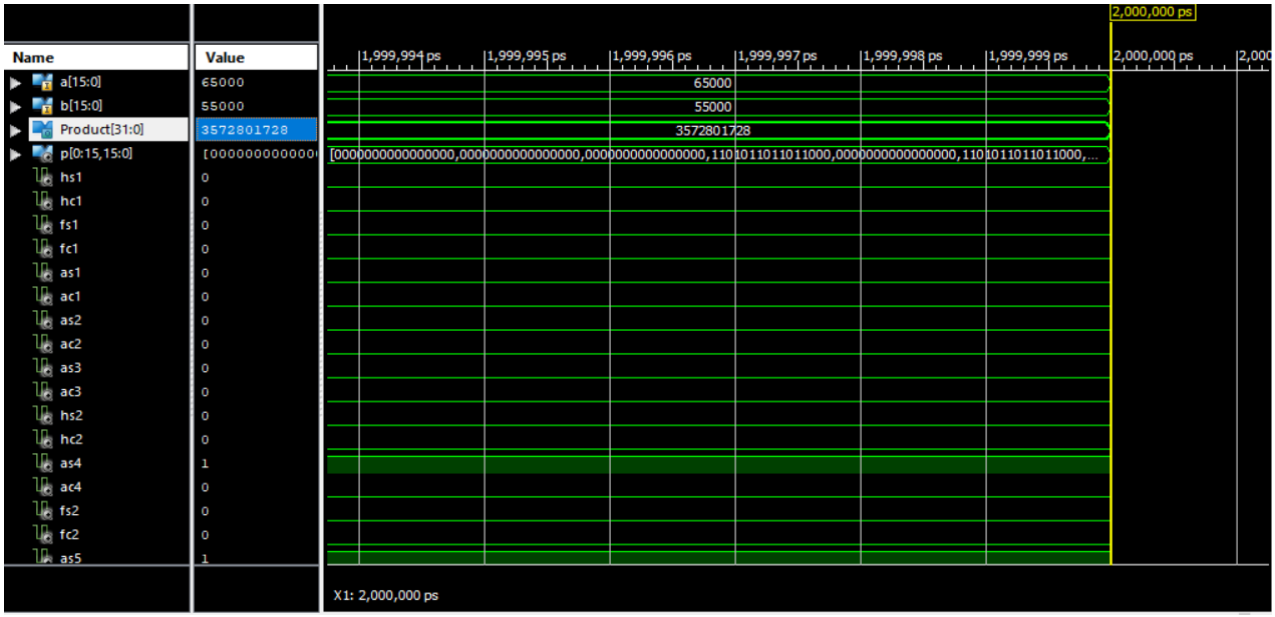

Figure 9: Simulation results for proposed Approximate Multiplier using 5:2 Compressor

Table 3: Synthesis results of existed and proposed mulltipliers

PARAMETERS	Approximate multiplier using 4:2 compressor	Approximate multiplier using 5:2 compressor
No of LUT's	581	548
Delay(ns)	39.790	37.628
Power(m.Watt)	10.288	9.704

VI. CONCLUSION

This paper presents approximate 4:2 compressor architectures. A high speed area efficient 5:2 compressor architecture is proposed, which achieved a considerable reduction in area, delay and power when compared to other state-of-the-art compressor designs. The proposed design has comparable accuracy with 25% error rate and equal positive and negative absolute error deviation of 1. As a result, the proposed design reduces MED and MRED considerably without reducing the error rate. The architecture was designed and implemented at the transistor level using 45-nm technology with a supply voltage of 1 V. The design used in image processing applications, like image multiplication and image smoothing.

REFERENCES

[1] S. Hashemi, R. I. Bahar, and S. Reda, ''DRUM: A dynamic range unbiased multiplier for approximate applications,'' in Proc. IEEE/ACM Int. Conf. Computer-Aided Des. (ICCAD), Austin, TX, USA, Nov. 2015, pp. 418–425.

[2] S. Narayanamoorthy, H. A. Moghaddam, Z. Liu, T. Park, and N. S. Kim, ''Energy-efficient approximate multiplication for digital signal processing and classification applications,'' IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 23, no. 6, pp. 1180–1184, Jun. 2015.

[3] G. Zervakis, S. Xydis, K. Tsoumanis, D. Soudris, and K. Pekmestzi, ''Hybrid approximate multiplier architectures for improved poweraccuracy trade-offs,'' in Proc. IEEE/ACM Int. Symp. Low Power Electron. Des. (ISLPED), Rome, Italy, Jul. 2015, pp. 79–84.

[4] B. Shao and P. Li, ''Array-based approximate arithmetic computing: A general model and

applications to multiplier and squarer design,'' IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 62, no. 4, pp. 1081–1090, Apr. 2015.

[5] C.-H. Chang, J. Gu, and M. Zhang, ''Ultra low-voltage low-power CMOS 4-2 and 5-2 compressors for fast arithmetic circuits,'' IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 51, no. 10, pp. 1985–1997, Oct. 2004.

[6] A. Saha, R. Pal, A. G. Naik, and D. Pal, ''Novel CMOS multi-bit counter for speed-power optimization in multiplier design,'' AEU-Int. J. Electron. Commun., vol. 95, pp. 189–198, Oct. 2018.

[7] S.-F. Hsiao, M.-R. Jiang, and J.-S. Yeh, ''Design of high-speed lowpower 3-2 counter and 4-2 compressor for fast multipliers,'' Electron. Lett., vol. 34, no. 4, p. 341, Feb. 1998.

[8] Z. Wang, G. A. Jullien, and W. C. Miller, ''A new design technique for column compression multipliers,'' IEEE Trans. Comput., vol. 44, no. 8, pp. 962–970, Aug. 1995.

[9] J. Gu and C.-H. Chang, ''Ultra low voltage, low power 4-2 compressor for high speed multiplications,'' in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), Bangkok, Thailand, 2003, pp. 321–324.

[10] A. Momeni, J. Han, P. Montuschi, and F. Lombardi, ''Design and analysis of approximate compressors for multiplication,'' IEEE Trans. Comput., vol. 64, no. 4, pp. 984–994, Apr. 2015

[11] O. Akbari, M. Kamal, A. Afzali-Kusha, and M. Pedram, ''Dual-quality 4:2 compressors for utilizing in dynamic accuracy configurable multipliers,'' IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 25, no. 4, pp. 1352–1361, Apr. 2017

[12] M. Ha and S. Lee, ''Multipliers with approximate 4–2 compressors and error recovery modules,'' IEEE Embedded Syst. Lett., vol. 10, no. 1, pp. 6–9, Mar. 2018.

[13] Y. Guo, H. Sun, L. Guo, and S. Kimura, ''Low-cost approximate multiplier design using probability-driven inexact compressors,'' in Proc. IEEE Asia Pacific Conf. Circuits Syst. (APCCAS), Chengdu, China, Oct. 2018, pp. 291–294.