IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DESIGN AND ANALYSIS OF DADDA MULTIPLIER USING APPROXIMATE 4-2 COMPRESSOR

[1] VINOD KUMAR ANCHA, [2] K V HAREESH

[1]Student, [2] Asst. Professor

[1]Electronics and Communication Engineering,

[1]MLEC, Singarayakonda, Andhra Pradesh, India

Abstract: Approximate computing plays important role in error-tolerant and multimedia applications to reduce the power dissipation, speed and area with trade-off in accuracy. This paper proposes the design and analysis of a novel approximate 4–2 compressor and proposed design of Dadda Multiplier is presented using proposed compressor and to reduce the error at the output. Through several experimental evaluations on proposed designs, the efficiency of the proposed compressor and multiplier are calculated and their parameters are compared with the state-of-the-art approximate multipliers. From the results it is observed that the proposed compressor achieve a significant minimization in error rate compared to other approximate compressors presented in the literature. The proposed multiplier shows 35%, 36% and 17% reduction in power dissipation, delay and area respectively compared with exact multiplier. The proposed compressors are utilized to design 8× 8 Dadda multipliers. These multipliers have close accuracy when compared with state-of-the-art approximate multipliers.

Index Terms **– Area, Approximate computing, Compressor, Multimedia, Multiplier**

I. INTRODUCTION

The computation units in a processor have to deliver high performance and execution efficiency. These can be achieved by introducing approximation. Speed of operation of system is inversely proportional to the delay of the system requires immense parallel operations that require huge hardware and power consumption [19], [20].Energy and area efficient systems can be implemented by relaxing the precision and accuracy of the system. In order to maintain the tradeoff between delay, area and power, approximate computing has one of a promising solution.

Approximation computing result in faster systems operations with minimal design complexity

and power dissipation [21]_[23].The tradeoff between reduction in accuracy and area, delay, power dissipation, which does not affect the typical operation for machine learning and multimedia applications. the inability of human eye to detect difference in finer details within images and videos, effectively take as advantage to implement machine learning and multimedia applications. This level of error tolerance is used to implement approximate computing circuits for Artificial intelligence (AI) and Digital Signal Processing (DSP) applications.

Among different arithmetic units, adders and multipliers are the main prerequisites for microprocessors, digital filters and digital signal processors etc., [2]. A couple of approximate adders and multipliers have been proposed in literature. Four approximate full adders designed with logic complexity reduction are proposed in[3] and they are used in the implementation of approximate multipliers. in [4] approximate adders and subtractors using Memristor are proposed. An approximate multiplier implemented using an approximate adder and re-ordering the partial products is proposed in [5]. By considering only a part of m-bits from an n-bit operand based Approximate multiplication is proposed in [6]. the input operand is approximate using Dynamic Segment Method (DSM) and Static Segment Methods (SSM), a Dynamic Range Unbiased Multiplier (DRUM) is proposed In [7] ,which contains only a fixed number of input bits after a leadinglogic-1 for the multiplication. several techniques are available in the literature to minimize the delay and power consumption at partial product summation of a multiplier. Among them, compressors based method is the most popular. Compressors are implemented using full adders and/or half adders to count the number of ‘‘logic-1’s” in the input.

Various compressors 3–2, 4–2, 5–2 and 7–3 were proposed in the last two decades [2,8–10]. the 4–2 compressor is used for designing regularly structured Dadda multiplier [11].Optimized methods of 4–2 compressors have been proposed in [9,12,13]. To minimize the transistor count and delay. Two approximate 4–2 compressors approximation in the logic level are proposed in [14] To reduce the power consumption and delay. in [15], Four other dual-quality reconfigurable 4–2 approximate compressors are proposed .which have the capable to switch between accurate and approximate operation. in [16] An approximate multiplier with an approximate 4–2 compressor and error recovery unit is proposed.

Atop-down structure based An approximate multiplier is proposed in [17]. This approximated multiplier is divided into three blocks, based on number of partial products in each column, approximate compressors of sizes 4–2, 6–2 and 8–2 are selectively applied in the middle block. A grouped error recovery method is also added to improve the accuracy. Approximate adders are used to design a heterogeneous approximate multiplier is proposed in [18]. This multiplier uses all the three approximate adders, these adders are designed based on a genetic algorithm, which minimize the overall Mean Error Distance (MED) of the multiplier with a proper approximate adder combinations Jiangmin et al.in [25] proposed A transistor level XOR-XNOR based low power 4 - 2 compressor used in tree structured fast multipliers. a 4 - 2 and a novel 5 -2 compressor that work on low supply voltage of 0.6 V.have proposed in [24].In this work, a novel 4 - 2 compressor architecture is presented. The contributions of the work are listed below.

A new high speed area-efficient, low power 4 – 2 compressor is proposed. Overall error rate is 25% with equal number of +1 and -1 error difference. Dadda multiplier is designed with the proposed 4 -2 compressor .and compared with existing multipliers. The rest of the paper is organized as follows. A brief overview on the exact 4–2 compressor and the need for approximation in multipliers included in Section 2. The proposed approximate compressor describes in Section 3. The modified design of approximate Dadda multiplier describes in section 4. Simulation results for approximate compressors and approximate multipliers are discussed in Section 5. and finally conclusions in Section 6.

II. EXACT COMPRESSORS

In AI and DSP applications, one of most important arithmetic operation is multiplication. These applications require high speed multiplier architectures to involve high speed parallel operations with tolerable levels of accuracy. Introduction of approximation in multipliers leads to design of faster computations with minimal hardware complexity, delay and power, with accuracy in required levels.

In multiplication process Partial product summation is the one of speed limiting operation cause the propagation delay in adder structure. In order to minimize the propagation delay, compressors are introduced. Compressors calculate the sum and carry at each stage simultaneously. The resultant carry is added with a higher significant sum bit in the next level. This process is continued until the final product is produced.

A 4–2 compressor has five inputs (A1; A2; A3; A4; Cin) and three outputs (Sum; Carry; Cout).

All the input bits and outputs have the same binary weight. The compressor receives the input Cin from a previous block of order one binary bit lower in significance, and produces outputs Cout and Carry of order one binary bit higher in significance.

The general block diagram of exact 4–2 compressor is shown in Fig.1.

$$SUM = A1 \text{ XOR } A2 \text{ XOR } A3 \text{ XOR } A4 \text{ XOR } Cin \quad (1)$$

$$CARRY = Cin(A1 \text{ XOR } A2 \text{ XOR } A3 \text{ XOR } A4) + A4\ (\sim(A1 \text{ XOR } A2 \text{ XOR } A3 \text{ XOR } A4)) \quad (2)$$

$$Cout = A3\ (A1 \text{ XOR } A2) + A1(\sim(A1 \text{ XOR } A2)) \quad (3)$$

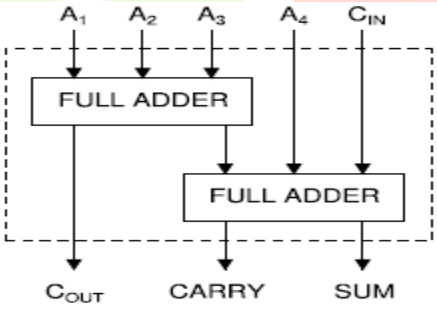

Fig.1. conventional 4-2 compressor

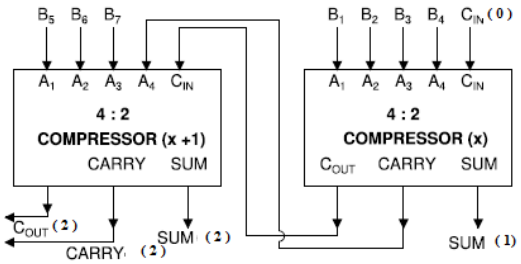

Fig.2.4-2 compressor chain

The approximate 4–2 compressor can be implemented with reducing the number of output bits to two. this approximate compressor is used in the implementation of multiplier, the reduction in the number of output bits effectively minimize the number of input bits of the succeeding compressors. Except the input combination ‘‘X4X3X2X1 = 1111”, in the remaining 15 cases the value of output bit Cout = 0. So in designing approximate compressors, Cout is not considered.

Table 2.1: Truth table for conventional 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

III. Proposed Approximate Compressor

The proposed high speed area-efficient 4-2 approximate compressor is proposed in this section. The compressor inputs are A1, A2, A3 and A4, outputs are CARRY' and SUM'. The input Cin and output Cout in the exact 4–2 compressor are completely ignored in the design of approximate 4–2 compressor.SUM can be generated using a multiplexer (MUX) based design approach.

Output of XOR gate(A1 XOR A2) acts as the select line for the MUX. When select line goes high, (A3 AND A4) is selected and when it goes low, (A3 OR A4) is selected. By introducing an error with error distance 1 in the truth table of the exact compressor, the proposed 4 -2 compressor , carry logic can be implemented with an OR gate. The logical expressions for realization of SUM' and CARRY' are given below.

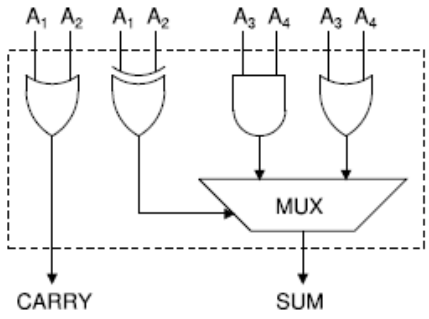

Fig.3. proposed 4-2 compressor

Table 3.1: Truth table of proposed 4-2 compressor

A_1	A_2	A_3	A_4	$CARRY$	SUM	ED
0	0	0	0	0	0	0
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	**0**	**1**	-1
0	1	0	0	**1**	**0**	+1
0	1	0	1	1	0	0
0	1	1	0	1	0	0
0	1	1	1	1	1	0
1	0	0	0	**1**	**0**	+1
1	0	0	1	1	0	0
1	0	1	0	1	0	0
1	0	1	1	1	1	0
1	1	0	0	1	0	0
1	1	0	1	1	1	0
1	1	1	0	1	1	0
1	1	1	1	**1**	**1**	-1

IV. Proposed Multiplier

In this section, the proposed design of an 8 × 8 approximate multiplier is presented. The multiplication operation can be divided into 3 parts as follows.

- Generation of partial products
- Arrange of partial products into two rows
- The computation of final result generally using adders

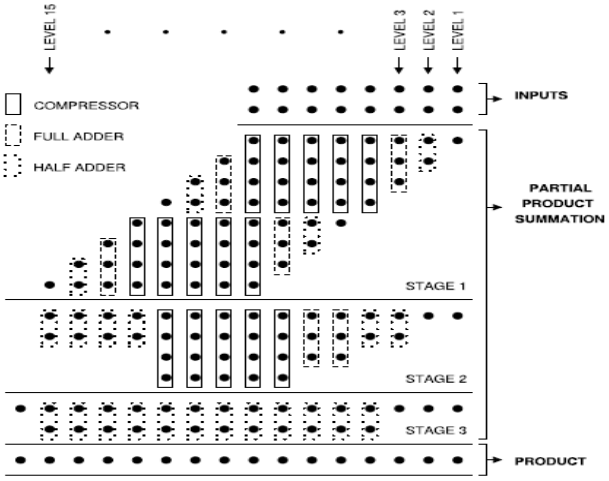

Fig 4.8×8 Approximate multiplier

The multiplier overall performance is mainly depends on the optimization of the second module. The 8 ×8 unsigned Dadda multiplier is designed using the proposed approximate compressors. The architecture of the Dadda multiplier designed using conventional 4–2 compressors is presented in [14].Each dot in the figure indicate a partial product obtained from AND gates. The reduction module contain half-adders, full adders and approximate proposed 4–2 compressors. The approximate compressors are indicated with rectangles.

The main objective of replacing all the conventional compressors with approximate compressors is to minimize the delay, power consumption and area significantly. If the approximate compressors are used at the least significant columns (rightmost columns in Fig.2.) then the performance of approximate multipliers increase in terms of accuracy. The overall error at the output of the approximate multiplier is minimized by rearranging the order of input bits to the approximate compressor.

V. SIMULATION RESULTS AND DISCUSSION

In this section proposed approximate compressor and multiplier is simulated and synthesis by Xilinx ISE 14,7 and vivado 2019.2 are discussed.

The proposed approximate compressor and optimized existing proposed compressor in [9] are simulated. For an effective comparison, Proposed approximate 4–2 compressor and existing 4-2 compressor designs are simulated and synthesized using Xilinx ISE 14.7 and Vivado 2019.2. The proposed approximate compressor based DADA multiplier and optimized existing proposed compressor based multiplier in [9] are simulated. For an effective comparison, Proposed approximate 4–2 compressor based multiplier and existing 4-2 compressor Multiplier designs are simulated and synthesized using Xilinx ISE 14.7 and Vivado 2019.2.

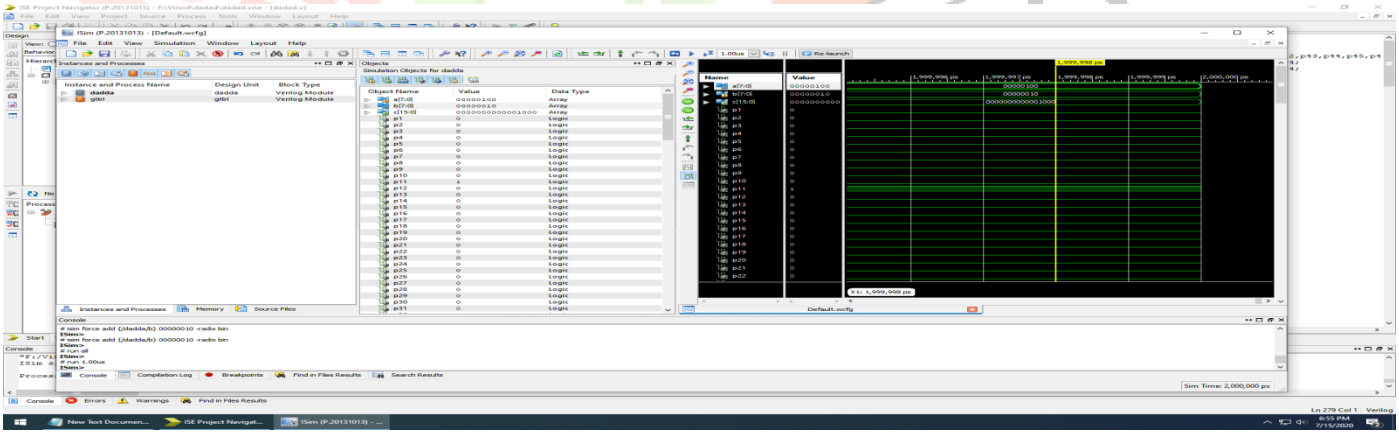

Fig.5. simulation result of existing multiplier using 4-2 compressor

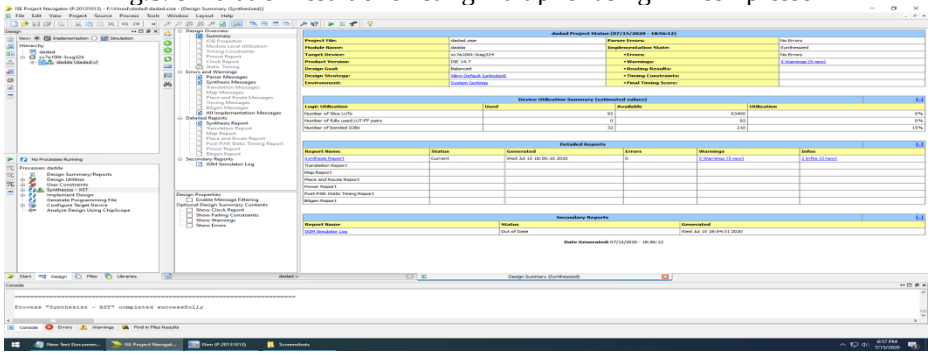

Fig.6. Area report of existing multiplier using 4-2 compressor

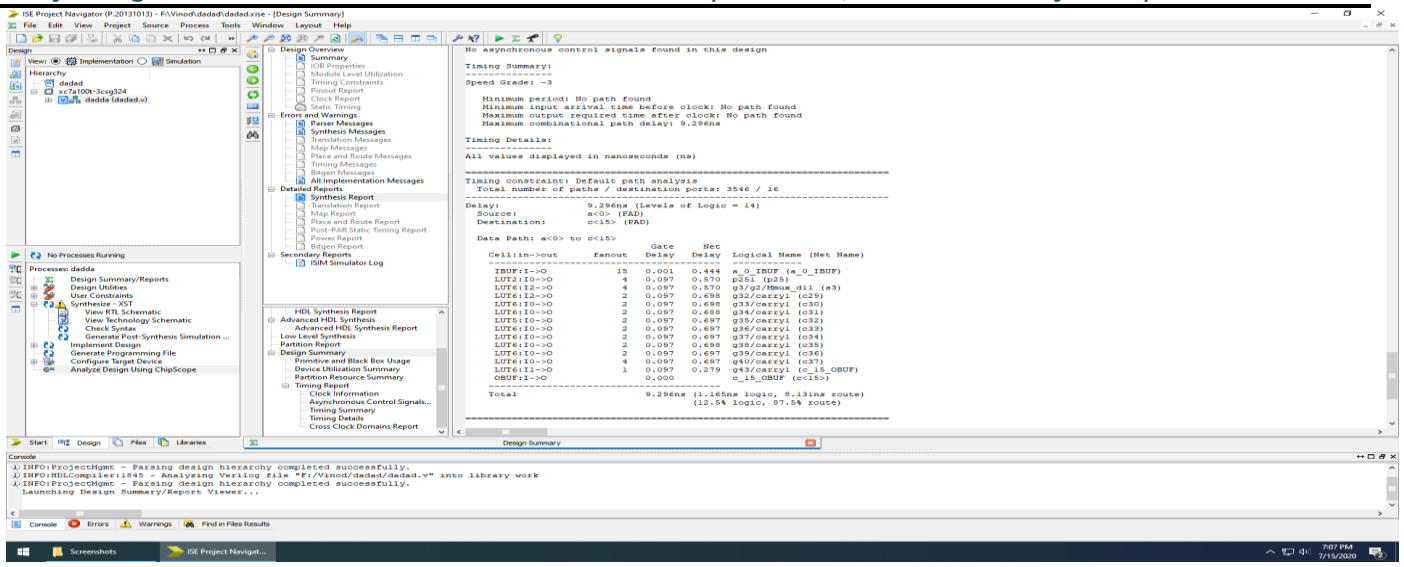

Fig.7. Delay report of existing multiplier using 4-2 compressor

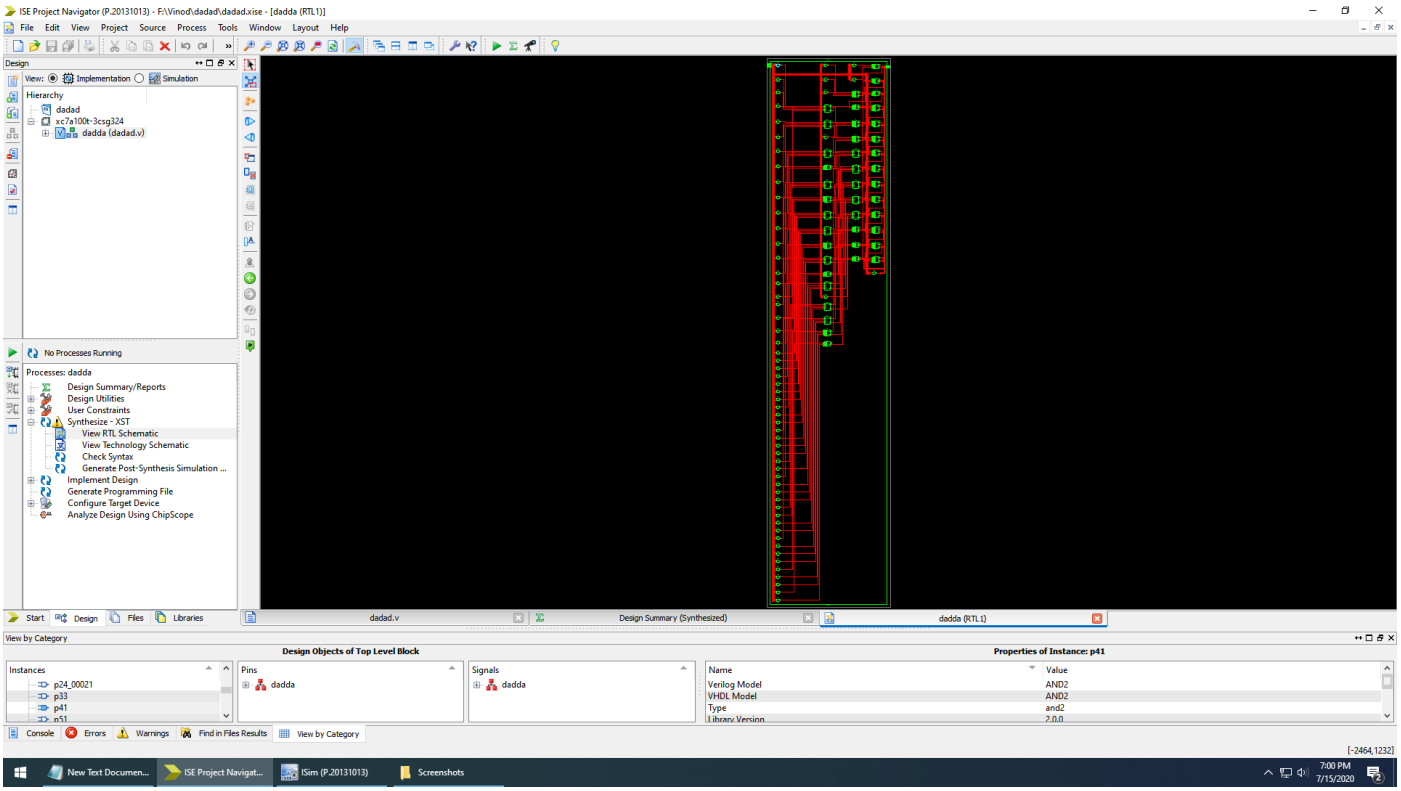

Fig.8. RTL view of existing multiplier using 4-2 compressor

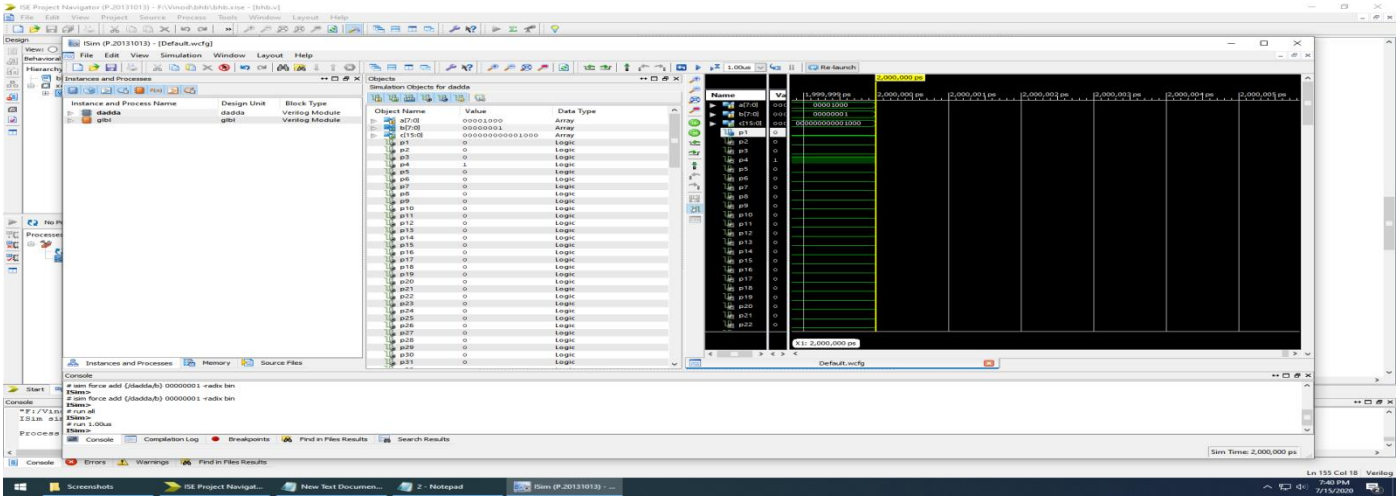

Fig.9. simulation result of proposed multiplier using 4-2 compressor

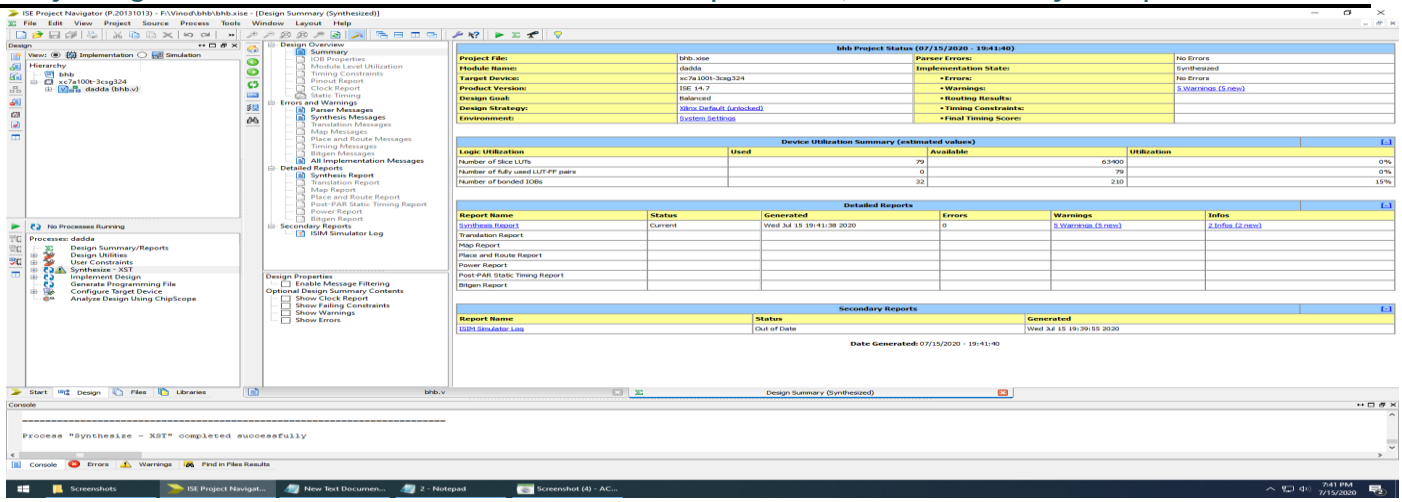

Fig.10. Area report of proposed multiplier using 4-2 compressor

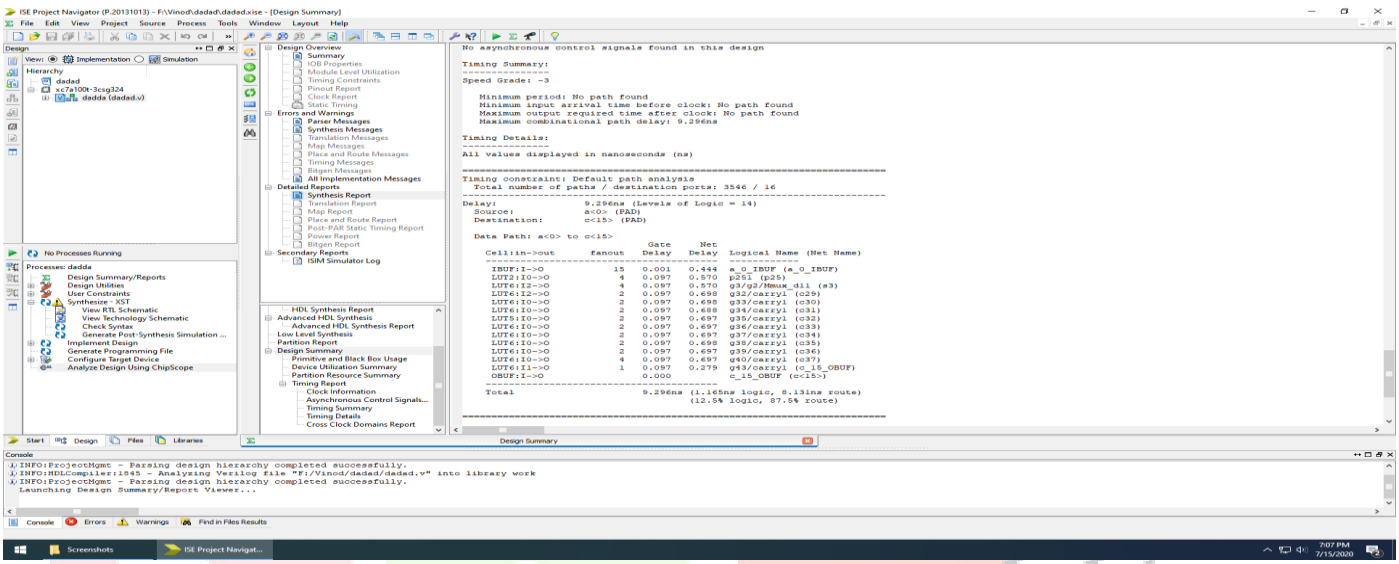

Fig.11. Delay report of proposed multiplier using 4-2 compressor

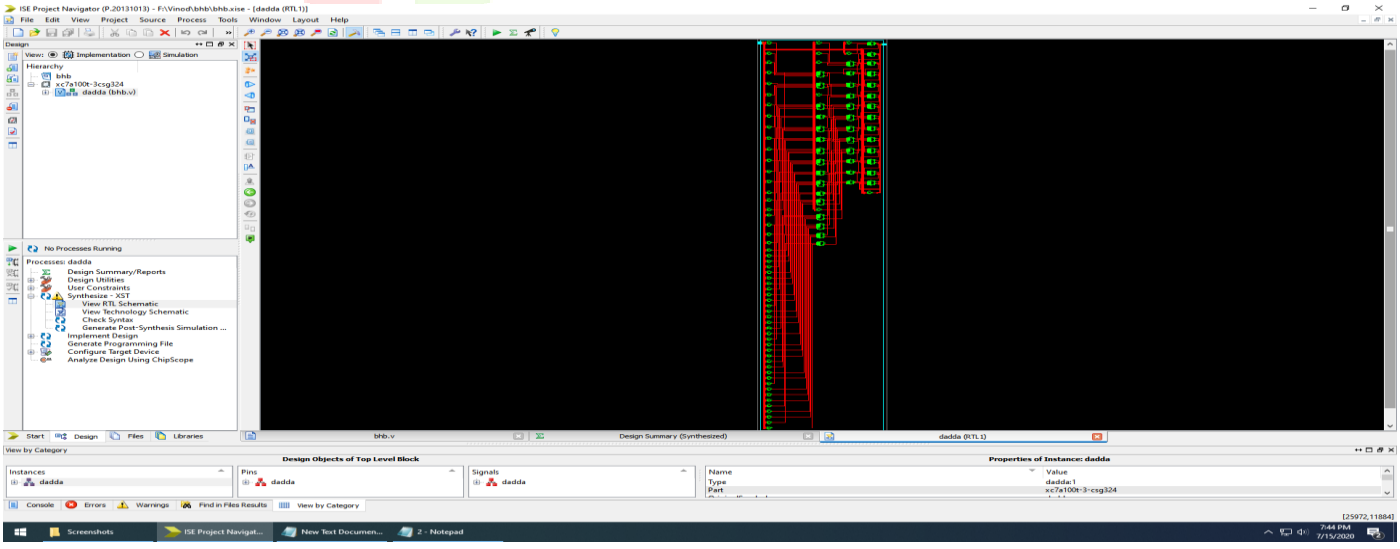

Fig.12. RTL view of proposed multiplier using 4-2 compressor

Table 5.1: Performance analysis of proposed and existing multipliers

Design metrics	Existing approximate multiplier [1]	Proposed approximate multiplier
No. of LUTs	95	79
Delay (ns)	10.5	8.3
ADP	997.5	655.7

From the table it is observed that proposed multiplier will take 79 LUTs for designing but existing multiplier take 95 LUTs, from that proposed multiplier will take 16 LUTs less than existing multiplier i.e. 21% area efficient. proposed multiplier will take 8.3ns delay but existing multiplier take 10.3ns delay, from that proposed multiplier is faster as compared to existing approximate multiplier i.e. 21% speed improved as compared to existing. proposed multiplier will take 655.7 but existing multiplier take 997.5 , from that proposed multiplier ADP is 341.8 less than existing multiplier i.e. 35% ADP is efficient as compared to existing approximate multiplier.

VI. Conclusion

A novel approximate 4 -2 compressor designs are presents in this paper. Firstly, a high speed area efficient compressor design is proposed, which attained a considerable reduction in area, delay and power when compared to other state-of-the-art approximate compressor designs. The proposed approximate multiplier design has accuracy with 25% error rate and equal positive and negative absolute error deviation of 1.The proposed approximate multiplier shows a significant improvement in terms of area, power consumption and delay as compared to the existing approximate multiplier.

In conclusion, this work has shown that multiplier can be implemented for approximate computing by an approximate design of a compressor; this proposed multiplier offers advantages in terms of design parameters compared to existing approximate multipliers, and in terms of accuracy metrics, area, delay and power consumption.

References

[1] K. Manikantta Reddy, M. H. Vasantha, Y. B. N. Kumar, and D. Dwivedi,``Design and analysis of multiplier using approximate 4-2 compressor,"AEU-Int. J. Electron. Commun., vol. 107, pp. 89_97, Jul. 2019.

[**2].**Radhakrishnan D, Preethy AP. Low power CMOS pass logic 4–2 compressor for high-speed multiplication. Proc of the 43rd IEEE Midwest Sympos Circuits and Systems 2000;3:1296–8.

[3] Gupta V, Mohapatra D, Raghunathan A, Roy K. Low-power digital signal processing using approximate adders. IEEE Trans Comput-Aid Des Integr CircSyst 2013;32(1):124–37.

[4] Muthulakshmi S, Dash Chandra Sekhar, Prabaharan SRS. Memristor augmented approximate adders and subtractors for image processing applications: an approach. AEU – Int J Electron Commun 2018;91:91–102.

[5] Liu C, Han J, Lombardi F. A low-power, high-performance approximate multiplier with configurable partial error recovery. Des Automat Test in Europe Conf Exhib (DATE), Dresden 2014:1–4.

[6] Narayanamoorthy S, Moghaddam HA, Liu Z, Park T, Kim NS. Energy-efficient approximate multiplication for digital signal processing and classification applications. IEEE Trans Very Large Scale Integration (VLSI) Syst 2015;23(6):1180–4.

[7] Hashemi S, Bahar RI, Reda S. DRUM: a dynamic range unbiased multiplier for approximate applications. In: IEEE/ACM Int. Conf. on Computer-Aided Design (ICCAD), Austin, TX; 2015. p. 418–25.

[8] Hsiao SF, Jiang MR, Yeh JS. Design of high-speed low-power 3–2 counter and 4–2 compressor for fast multipliers. Electron Lett 1998;34(4):341–3.

[9] Chang CH, Gu J, Zhang M. Ultra low-voltage low-power CMOS 4–2 and 5–2 compressors for fast arithmetic circuits. IEEE Trans Circ Syst I: Reg Pap 2004;51 (10):1985–97.

[10] Saha Aloke, Pal Rahul, Naik Akhilesh G, Pal Dipankar. Novel CMOS multi-bit counter for speed-power optimization in multiplier design. AEU – Int JElectron Commun Oct. 2018;95:189–98.

[**11]** Wang Z, Jullien GA, Miller WC. A new design technique for column compression multipliers. IEEE Trans Comput Aug. 1995;44(8):962–70.

[12] Gu J, Chang CH. Ultra low voltage, low power 4–2 compressor for high speed multiplications. In: Proc of Int Symp on Circuits and Systems; 2003. p. 321–324, vol 5.

[13] Parhami B. Computer arithmetic: algorithms and hardware designs. 2nded. New York: Oxford Univ. Press; 2010.

[14] Momeni A, Han J, Montuschi P, Lombardi F. Design and analysis of approximate compressors for multiplication. IEEE Trans Comput 2015;64(4):984–94.

[15] Akbari O, Kamal M, Afzali-Kusha A, Pedram M. Dual-quality 4:2 compressors for utilizing in dynamic accuracy configurable multipliers. IEEE Trans Very Large Scale Integr Syst 2017;25(4):1352–61.

[16] Ha M, Lee S. Multipliers with approximate 4–2 compressors and error recovery modules. IEEE Embed Syst Lett Mar. 2018;10(1):6–9.

[17] Guo Y, Sun H, Guo L, Kimura S. Low-cost approximate multiplier design using probability-driven inexact compressors. IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), Chengdu; 2018. p. 291–4.

[18] Alouani I, Ahangari H, Ozturk O, Niar S. A novel heterogeneous approximate multiplier for low power and high performance. IEEE Embed Syst Lett Jun. 2018;10(2):45–8.

[19] S. Ghosh, D. Mohapatra, G. Karakonstantis, and K. Roy, ``Voltage scalable high-speed robust hybrid arithmetic units using adaptive clocking," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 9, pp. 1301_1309, Sep. 2010.

[20] D. Baran, M. Aktan, and V. G. Oklobdzija, ``Multiplier structures for low power applications in deep-CMOS," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), Rio de Janeiro, Brazil, May 2011, pp. 1061_1064.

[21] S. Mittal, ``A survey of techniques for approximate computing," ACM Comput. Surv., vol. 48, no. 4, pp. 1_33, Mar. 2016.

[22] H. Jiang, C. Liu, L. Liu, F. Lombardi, and J. Han, ``A review classi_cation and comparative evaluation of approximate arithmetic circuits," ACM J. Emerg. Tech. Comput. Syst., vol. 13, no. 4, p. 60, 2017.

[23] J. Liang, J. Han, and F. Lombardi, ``New metrics for the reliability of approximate and probabilistic adders," IEEE Trans. Comput., vol. 62,no. 9, pp. 1760_1771, Sep. 2013.

[24] 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.

[25] 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.