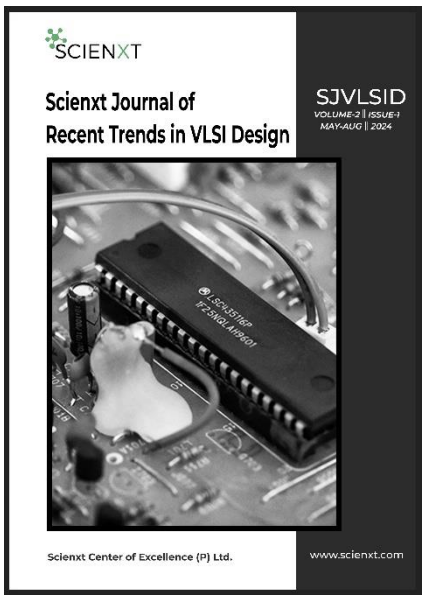


Scienxt Journal of Recent Trends in VLSI Design
Volume-2 || Issue-1 || May-Aug || Year-2024 || pp. 1-15

Highly efficient 4:2 compressor using carry based approximate full adder on digital circuit design

[*1]Aswathy E. R, [2]Dr. S. Swapna Kumar

[*1]Assistant Professor, Department of Electronics & Communication Engineering Vidya Academy of Science & Technology, India

[2] Head of the Department, Department of Electronics & Communication Engineering, Vidya Academy of Science & Technology, India

**Corresponding Author: Aswathy E. R*
Email: aswathy5314@vidyaacademy.ac.in

Abstract:

The 4:2 compressors have been widely used in multiplier realizations. For rapid digital arithmetic integrated circuits, we suggest a very efficient 4:2 compressor circuit. Four: 2 compressor circuits were built using a carry-based approximation full adder. In error-tolerant applications where precise processing units are not always essential, approximate computing can decrease design complexity while enhancing performance and power efficiency. They can be substituted with their approximate counterparts. A new design technique for carrying-based approximation of full adders has been modified to include variable probability terms. Most multimedia programs can glean useful information from slightly erroneous results. As a result, we do not need to generate precise results. This short introduces a novel gate level logic modification technique for approximating a full adder in order to take advantage of the relaxation of numerical exactness. For reducing the area, the sum term is changing to recommend the carry-based approximation adder (CBAA) instead of the considerable XOR operation in the typical full adder. The power consumption, area, and delay of the suggested 4:2 compressor circuit were compared to previously reported circuits, and the proposed circuit was found to have the lowest power consumption, least area, and least delay. It is recommended to employ a Proposed Compressor to increase efficiency while decreasing error rate. VHDL circuit construction is simulated by using Xilinx ISE 14.7.

Keywords:

Approximate computing, Carry based approximate adder, Area efficient, 4:2 Compressor, VHDL

1. Introduction:

Accurate, exact, and deterministic algorithms are used in the computation of many scientific and technical problems. However, since many signal/image processing and multimedia applications are error-tolerant and generate results good enough for human perception, precise and accurate computations are not always required [1]. A decrease in circuit complexity, and consequently in area, power, and delay, is critical to the functioning of a circuit in certain error-tolerant applications. For this reason, error-tolerant systems can benefit from the usage of approximation computing, which lowers accuracy while retaining speed and/or power consumption [2].

Due to its capacity to trade computation accuracy for lower computational complexity, approximate computing has drawn a lot of attention. This has resulted in digital designs that are faster, smaller, lighter, and more energy-efficient. Approximately correct outcomes that fall within a given error bound are frequently acceptable in various applications, including digital image, audio, and video processing, data analytics, machine learning, artificial intelligence, etc. [3-6]. The term approximate computing refers to both computer software and hardware [7, 8]. The two main areas of approximate computing in terms of hardware-level approximation are approximate circuits [9] and storage [10], with research on approximate circuits concentrating on approximate arithmetic circuits [11] [12] and approximate logic synthesis [13].

A compressor is a circuit that is mostly used in multipliers to add terms for partial products while reducing the operands. A standard M:N compressor generates an N-bit binary number from M equally weighted input bits. Also referred to as a complete adder, the 3:2 compressor is the most basic and commonly utilized compressor. It yields two outputs after summing together three inputs. In a similar vein, two tumbled 3:2 compressor circuits can likewise be used to construct a 4:2 compressor. As illustrated in fig.1, the traditional implementation of a 4:2 compressor consists of two serially coupled full adders.

Two full-adders (FA) are used in the typical implementation of a 4:2 compressor to add binary number cells. Two full adders are connected in series make up a 4:2 compressor. Rather than using another adder, we utilize a compressor adder, which has expansion of carry term. A compressor is a contemporary digital circuit that operates at fast pace and requires little gate design. With a quick processor and small size, this compressor becomes a vital instrument for quick multiplication adding techniques.

The five inputs of the precise compressors are X1, X2, X3, X4, andCin. Three outputs are produced: Carry,$Cout$, and Sum [14].

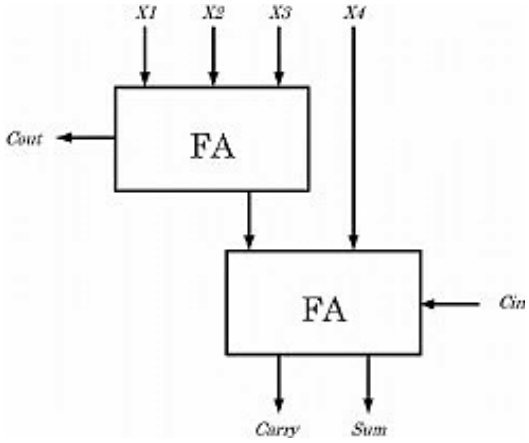


Figure.1: Conventional 4:2 Compressor using two full adder [14]

The Output of the conventional compressor is given as,

$$Sum = x1 \oplus x2 \oplus x3 \oplus x4 \oplus Cin \quad (1)$$

$$Cout = x1 \oplus x2\, x3 + x1x2 \quad (2)$$

$$Carry = (x1 \oplus x2 \oplus x3 \oplus x4)\, Cin + (x1 \oplus x2 \oplus x3)\, x4 \quad (3)$$

A compressor condenses an n-bit integer to two bits by using the right amount of repetition. Carry bit Cout moves up the hierarchy. For low power applications, full adder compressors have been used in arithmetic and digital signal processing (DSP) circuits. Mostly, multiplier architectures use compressors. The three components of a multiplier's design are partial product generation, partial product accumulation, and final addition. The partial products are the result of the multiplication function, which is carried out bit by bit in the first section. The most important component of multipliers is the second part, which improves performance [15].

The remaining paper is structured as follows: Section II addresses the related works of approximate computing in 4:2 compressor, concept of approximate arithmetic circuits whereas, section III explains the design methodology. Section III describes the structure of proposed approximate 4:2 compressors Part IV discusses the simulation results. Section V provides a summary of the conclusions.

2. Literature survey:

Error-tolerant applications that can tolerate faults, such as multimedia signal processing and data mining, may not always require exact computing units. Approximate computing, on the other hand, can improve performance and power efficiency while lowering design complexity. They can be replaced with their approximate equivalents. A compressor is a device that is typically used in multipliers to reduce the operators while adding terms of partial products. Adding binary integer cells with two full adders (FA) is a common implementation of 4:2 compressors. A 4:2 compressor consists of two full adders linked in series. Low carry propagation is the reason why compressor adders are preferred over other adders. A carry-based approximate adder can be used in applications that need less power. The proposed carry-based area and power measurements are contrasted with traditional ones.

For quick DMAICs, previous work on the 4:2 compressor circuits [16] is suggested. In order to realize multipliers, the 4-2 compressor has been used often. The circuit for a 4:2 compressor has been constructed using a novel exclusive OR (XOR) and exclusive NOR (XNOR) module. When compared to previous documented circuits, the power consumption, delay, and PDP of the suggested 4:2 compressor circuit show that it has the lowest power consumption and the shortest latency. SPICE, which is based on TSMC 0.18# m CMOS technology has been used to run simulations.

According to A. Momeni et al. [17], approximate computing is a compelling paradigm for digital processing at Nano meter dimensions. Particular significance is seen for computer arithmetic designs in imprecise computing. This article focuses on the design and analysis of two new approximate 4:2 compressors for use in a multiplier. When it comes to peak signal-to-noise ratio and average normalized error distance, two of the suggested multiplier designs are superior. When compared to an exact design, the findings show that the suggested designs significantly reduce power dissipation, latency, and transistor count.

To achieve precise results, standard digital computational hardware blocks with various architectures are described by Pawan Sonwane1 et al. [18]. This study addresses the construction of a low-power, imprecise 4:2 compressor employing approximation as a fundamental technique. Power consumption is the primary factor to take into account when constructing computationally demanding blocks. By using fewer transistors, the complex logic of the 4:2 compressors' switching capacitance and power consumption are made simpler.

An innovative approach for analyzing and implementing a full adder circuit utilizing VHDL technology is presented by Bobbinpreet Kaur et al. [19]. It displays the technology schematic diagrams and Register Transfer Level (RTL) schematic diagrams for the various VHDL architectural modeling styles, such as dataflow, behavioral, and structural modeling.

A novel paradigm for the energy-efficient design of circuits and systems was presented by Jie Han [20]. This is a quick overview of approximation computing and discusses the difficulties it faces in terms of its potential uses in computing systems that are error- and energy-resistant.

Suganthi Venkatachalam et al. [21], discusses a novel multiplier approximation design strategy. Differentiating probability terms are introduced by modifying the partial products of the multiplier. The accumulation of modified partial products is subject to variation in the logic complexity of approximation depending on their likelihood. The synthesized results of the suggested models achieve the highest peak signal to noise ratio.

Using the numerical exactness relaxation, Manickam Ramasamy et al. [22] describe an entirely new gate level logic modification technique for full adder approximation. To lessen area complexity, the total term of the regular full adder is altered by incorporating a carry-based approximation adder (CBAA) to replace the necessary XOR operation. To illustrate this idea, provide a number of imprecise approximate type complete adders with reduced gate complexity and utilize them to create approximate multi bit adders.

3. Design and methodology:

An approximate logic implementation, in contrast to conventional adder design, modifies certain truth table entries [22]. To suggest a new form of approximate adder, compute the total term from the carry term. In approximation for carry calculation, one OR gate and one AND gate substitute one XOR gate, two AND gates, and one OR gate. By using a NOT gate to flip the carry term, the sum term is determined. In more situations, this keeps the difference between the original and approximation numbers as close as feasible while further simplifying the situation. Carry-based approximation adders are useful in low-power applications. A carry-based adder is more efficient than a regular adder [22]. The findings show that the amount of memory used overall, area, and delay have all significantly decreased. Consequently, approximation architecture uses less space, needs less time, and is simpler to use in hardware structures.

Table. 1: Performance analysis of conventional and carry based approximate adders [23]

Type	***Delay (nS)***	***TMU (kB)***	***No of LUT***	***I/O Ports***
Conventional Adder	5.776	200308	2	5
Carry Based Approximate Adder I	5.753	200500	1	4
Carry Based Approximate Adder II	5.601	200308	0	2
Carry Based Approximate Adder III	5.753	199924	1	4
Carry Based Approximate Adder IV	5.753	199860	1	4

Table. 1 compares carry-based approximate adder II with carry-based approximate adder IV, demonstrating that carry-based approximate adder IV requires less space and time. The proposed design requires the use of carry-based approximation adder II and carry-based approximate adder IV [23]. This section proposes two Proposed Compressor designs. It makes intuitive sense to replace the exact full-adder cells with estimated full-adder cells 2 and 4 in order to create an approximate 4:2 compressor. On the other hand, these designs are exceedingly efficient; in terms of delay, memory usage, and area, they significantly outperform an accurate compressor. It was decided to propose two approximation compressors, which lowered the exact compressor.

3.1. Proposed compressor 1:

Proposed Compressor 1 is obtained using a carry-based approximate adder 2, which comprises two AND gates, two OR gates, and two inverters as shown in Fig. 2. The proposal's design 1 enables the carry to be simplified to Cin by altering the value of the other 8 outputs. By introducing compressor 1 to eliminate the vital XOR operation in the conventional compressor,

the total term of the conventional compressor is modified to reduce area complexity. The implementation of a proposed compressor I design consists of two serially coupled Carry base approximate full adder II. Compared to the conventional Compressor, proposed compressor 1 yields significant reduction in delay, number of look up tables, number of input output ports and total memory usage. This leads to an increase in the execution speed of the circuit.

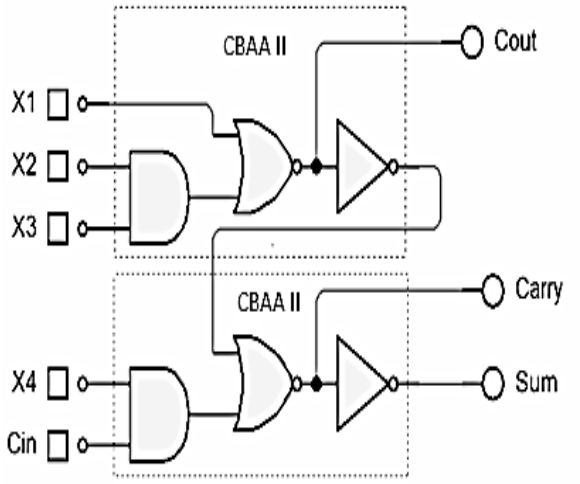


Figure. 2: Proposed Compressor1

3.2. Proposed compressor 2:

A carry-based approximate adder 4 with six AND gates, two OR gates, and two inverters is utilized to obtain it, as demonstrated in Fig. 3. Design 2 simplifies the carry to Cin by altering the values of the other 8 outputs. Minimizing area complexity is possible by proposing a compressor II that eliminates the essential XOR operation in the conventional compressor. The implementation of a proposed compressor II design consists of two serially coupled Carry base approximate full adder IV. Compared to the conventional Compressor, proposed compressor II yields significant reduction in delay, number of look up tables, number of input output ports and total memory usage. This leads to an increase in the execution speed of the circuit.

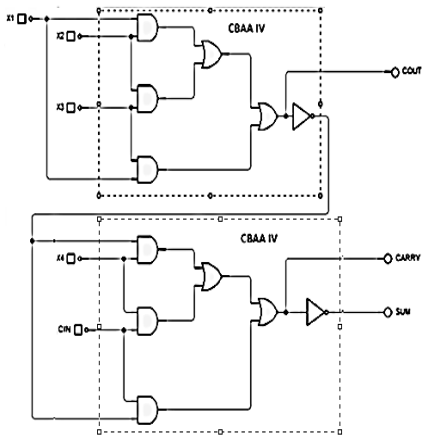


Figure. 3: Proposed compressor 2:

4. Result and analysis:

The simulation of both conventional and proposed compressors is done using Xilinx software. Fig. 4 depicts the simulation result of a conventional compressor. The implementation of Proposed Compressor I utilized Carry Based Adder II, while the implementation of Proposed Compressor II utilized Carry Based Adder IV. In Fig. 5 and Fig. 6, the simulation results and device utilization summary of the proposed Compressor I and II are depicted. The compressors proposed were constructed, tested, and simulated in the Xilinx environment.

In Fig. 4,which shows the simulation results of conventional compressor I, the inputs of the conventional compressor are X1,X2,X3,X4 and Cin . $Cout$,Carry and Sum are the output of the conventional design .In the device utilization summary shows that the number of LUTs are 4, number of slices are 2, number of bonded IOBs are 8 and the number of input output ports are 8.

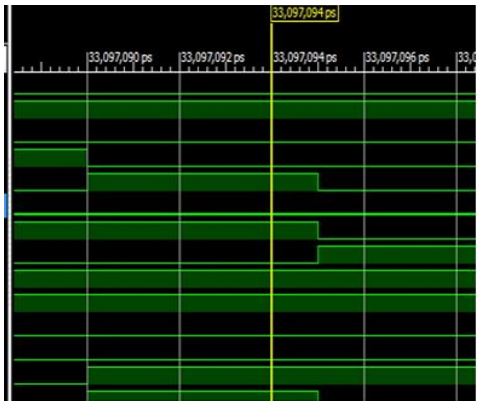


Figure.

```
Device utilization summary:
---------------------------

Selected Device : 3s100evq100-5

 Number of Slices:                        2  out of    960
 Number of 4 input LUTs:                  4  out of   1920
 Number of IOs:                           8
 Number of bonded IOBs:                   8  out of     66
```

Figure. 4: Device utilization summary and Simulation result of Conventional Compressor

In Fig 5, which shows the simulation results of conventional compressor I .The inputs of the conventional compressor are X1, X2, X3, X4, and in. $Cout$, Carry, and Sum are the outputs of

the conventional design . The device utilization summary shows that the number of LUTs is 2, the number of slices is 1, the number of bonded IOBs are 6 and the number of input-output ports is 6.

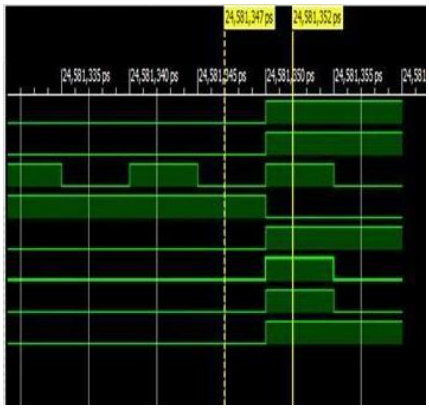


Figure. 5:

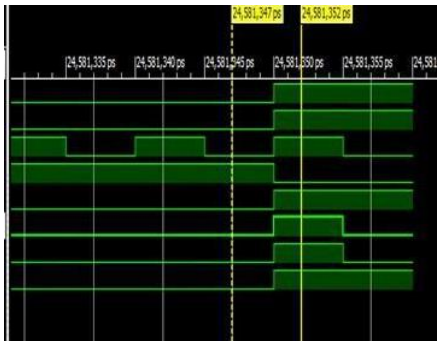
```
Device utilization summary:
---------------------------

Selected Device : 3s100evq100-5

 Number of Slices:                        1  out of    960
 Number of 4 input LUTs:                  2  out of   1920
 Number of IOs:                           6
 Number of bonded IOBs:                   6  out of     66
```

Figure. 5: Device utilization summary and Simulation result of Proposed Compressor1

In Fig 6, which shows the simulation results of conventional compressor I, the inputs of the conventional compressor are X1, X2, X3, X4, and Cin. $Cout$, Carry, and Sum are the outputs of the conventional design. The device utilization summary shows that the number of LUTs is 2, the number of slices is 1, several bonded IOBs are 6 and the number of input-output ports is 6.



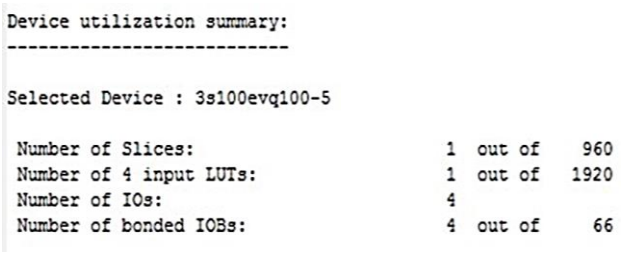


Figure. 6: Device utilization summary and Simulation result of Proposed Compressor2

The performance is compared based on the tradeoff parameters such as delay, Total memory usage, No of LUT, and No of I/O Ports, As compared to the conventional compressor is shown in Table 2. Implementation of the proposed compressors was done using Verilog HDL .Compared to the conventional Compressor, two proposed designs yield significant delay, total memory usage and area parameters reductions. In the first proposed design the reduction in delay was 12%; the reduction in LUTs is 50 %. The total memory usage reduction is 0.06%. The second proposed design reduced delay by 12%, the reduction of the number of input output ports, LUTs, has reached 25 %, 50% respectively and the reduction in total memory usage by 0.31%.

Table. 2: Performance Comparison of Conventional and Proposed Compressors

Type	Delay (nS)	TMU	No of LUT	No of I/O Ports
Conventional compressor	6.837	200500	4	8
Proposed Compressor I	6.054	200378	2	6
Proposed Compressor II	6.054	199860	2	6

Comparison of delay performance of conventional and Proposed Compressors shown in Fig. 7. Considering the Delay' column, it is found that the delay of proposed design I and proposed design II is equal and the same. The value of this delay is equal to 6.054 nS . Further, it is also observed that the delay of the proposed design I and proposed design II is found to be smaller compared with the conventional Compressor .The proposed design I and II shows an improvement in the delay reduction when compared to the delay-optimized cases. Improvements in delay reduction at the cost of smaller area and total memory usage.

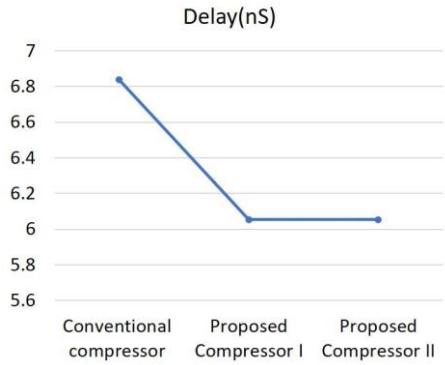


Figure. 7: Comparison of delay performance

In comparison to the conventional compressor, two proposed designs result in decreases in total memory usage and area parameters. The conventional design provides maximum memory usage of 200500 kB and proposed design 1 and 2 shows maximum memory usage of 200378kB and 199860 kB respectively. Comparison of Total memory usage of Conventional and Proposed Compressors shown in Fig. 8.

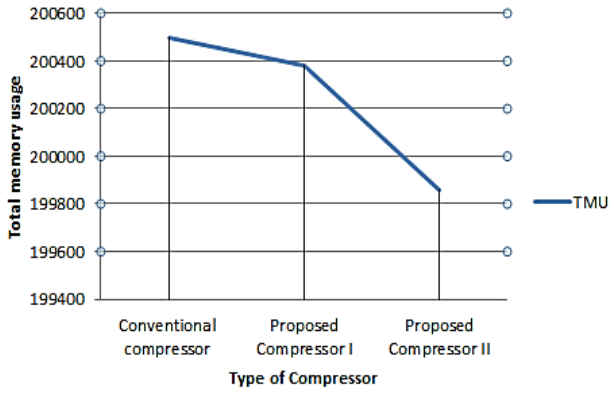


Figure. 8: Comparison of Total memory usage

As per Fig. 9, the Conventional compressors are likely to have moderate values of total area parameters. The proposed compressors are area-efficient. Conventional compressors are relatively slow, have high power hungry with moderate area parameters. Conversely, the proposed compressors I and II are fast and consume low area parameters. The proposed compressors I and II are more efficient in performance.

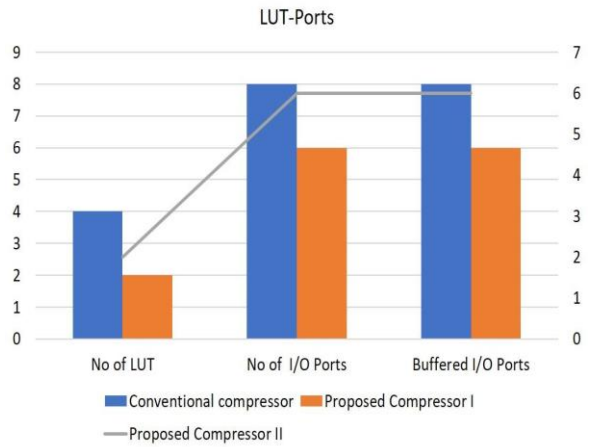


Figure. 9: Comparison of area parameters

5. Conclusion:

Exact computing is an emerging paradigm for computation at the Nano scale. Computer arithmetic has significant operational benefits for inexact computing; there is a lot of literature on approximate adders. The paper presents innovative approaches to designing two approximate 4:2 compressors utilizing a carry-based approximate adder. Comparing to an exact design, the proposed compressors demonstrate a significant decrease in device utilization, delay, and memory usage. The proposed 4:2 compressor designs can be used in applications with minimum loss in output quality while reducing significant delay and area. The current work focuses mainly on the area and utilizes delay reduction techniques that can be utilized to decrease power. In the future, the multipliers can be implemented with a variety of design configurations. A carry-based approximate adder was used to develop the proposed compressors. Two design configurations are present and are compared to the accurate conventional compressor. In terms of delay, total memory usage, number of LUTs, and number of I/O ports, the proposed compressor has been found to be superior. The focus of this work is on reducing power by reducing total memory usage and delay techniques. The multipliers can have different design configurations implemented in the future.

6. Future scope:

Different design configurations for the multipliers can be implemented in the future. The effectiveness of the circuit can be checked by calculating the parameters such as delay and area parameters.

7. Acknowledgment:

I am thankful for the assistance I received from the electronics and communication engineering .department.

5. References:

(1) J. Han and M. Orshansky, "Approximate Computing: An Emerging Paradigm for Energy-Efficient Design," in ETS'13, Avignon, France, May 27-31, 2013, pp. 1 – 6.

(2) R. Venkatesan, A. Agarwal, K. Roy, and A. Raghunathan, "MACACO: Modeling and analysis of circuits for approximate computing," in ICCAD 2011, pp. 667 – 673.

(3) M.A. Breuer, "Multi-media applications and imprecise computation," Proc. 8th Euromicro Conference on Digital System Design, pp. 2-7, 2005.

(4) H.-Y. Cheong, I.S. Chong, A. Ortega, "Computation error tolerance in motion estimation algorithms," Proc. International Conference on Image Processing, pp. 3289-3292, 2006.

(5) R. Nair, "Big data needs approximate computing: technical perspective," Communications of the ACM, vol. 58, pp. 104, 2015.

(6) M. Shafique, R. Hafiz, M.U. Javed, S. Abbas, L. Sekanina, Z. Vasicek, V. Mrazek, "Adaptive and energy-efficient architectures for machine learning: challenges, opportunities, and research roadmap," Proc. IEEE Computer Society Annual Symposium on VLSI, pp. 627-632, 2017.

(7) M. Shafique, R. Hafiz, S. Rehman, W. El-Harouni, J. Henkel, "Crosslayer approximate computing: from logic to architectures," Proc.53rd/EDAC/IEEE Design Automation Conference, pp.1-6, 2016.

(8) A. Sampson, W. Dietl, D. Fortuna, D. Gnanapragasam, L. Ceze, D. Grossman, "EnerJ: approximate data types for safe and general low power computation," ACM SIGPLAN Notices, vol. 46, pp. 164-174, 2011.

(9) K. Roy, A. Raghunathan, "Approximate computing: an energy-efficient computing technique for error resilient applications," Proc. IEEE Computer Society Annual Symposium on VLSI, pp. 473-475, 2015.

(10) A. Sampson, J. Nelson, K. Strauss, L. Ceze, "Approximate storage in solid-state memories," ACM Transactions on Computer Systems, vol. 32, pp. 1-9, 2014.

(11) Z. Yang, A. Jain, J. Liang, J. Han, F. Lombardi, "Approximate XOR/XNOR-based adders for inexact computing," Proc. IEEE International Conference on Nanotechnology, pp. 690-693, 2013.

(12) H. Jiang, C. Liu, N. Maheshwari, F. Lombardi, J. Han, "A comparative evaluation of approximate multipliers," Proc. IEEE/ACM International Symposium on Nanoscale Architectures, pp. 191-196, 2016.

(13) S. Venkataramani, A. Sabne, V. Kozhikkottu, K. Roy, A. Raghunathan, "SALSA: systematic logic synthesis of approximate circuits," Proc. Design Automation Conference, pp. 796-801, 2012.

(14) K. Prasad and K. K. Parhi, "Low-power 4-2 and 5-2 compressors," in Proc. of the 35th Asilomar Conf. on Signals, Systems and Computers, vol. 1, 2001, pp. 129–133.

(15) 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, pp. 341–343, 1998

(16) Sanjeev Kumar, Manoj Kumar,"4-2 Compressor Design with New XOR-XNOR Module", International Conference on Advanced Computing & Communication Technologies, IC-4,2014

(17) Momeni, J. Han, P. Montuschi, and F. Lombardi, "Design and analysis of Proposed Compressors for multiplication," IEEE Trans.Comput., vol. 64, no. 4, pp. 984–994, Apr. 2015.

(18) Pawan Sonwane. P. Malathi, Manish Sharma, "Design of Low Power Inexact 4:2 Compressors using Approximate Adder," IEEE International Conference on Computer, Communication and Control, IC4-2015

(19) Bobbinpreet Kaur, Girdhari Agarwal, Amandeep Kaur "Analysis and Implementation of a Full Adder Circuit using Xilinx Software'' International Conference on Advancements in Engineering and Technology (ICAET 2015)

(20) Jie Han "Introduction to Approximate Computing" IEEE 34th VLSI Test Symposium(VTS) Special session paper-2016

(21) Suganthi Venkatachalam and Seok-Bum Ko," Design of Power and Area Efficient Approximate Multipliers", IEEE Transactions on Very Large Scale Integration (VLSI) Systems (Volume: 25, Issue: 5, May 2017)

(22) Manickam Ramasamy, G.Narmadha,, S. Deivasigamani, "Carry based approximate full

adder for low power approximate computing", 7th International Conference on Smart Computing & Communications (ICSCC) , 2019

(23) Aswathy E R,Aksa David, "Implementation Of An Effective Acs Unit Using Carry Based Approximate Adder", International Research Journal of Modernization in Engineering Technology and Science,Volume:03/Issue:06/June-2021