ISSN: 2278-4632
Vol-13, Issue-08, August 2023

Design of High-Speed VLSI Architecture of Approximate Multiplier Using Majority Logic

U BHARADWAJA[1], Dr.S.RAVI CHAND [2]

[1]Assistant Professor, Nalla Narasimha Reddy Education Society's Group of Institutions - Integrated Campus, Hyderabad.

[2] Professor & HOD Department of ECE, Nalla Narasimha Reddy Education Society's Group of Institutions - Integrated Campus, Hyderabad

ABSTRACT

Quantum-dot cellular automata (QCA) is one of the most promising approaches to create ultralow-power and very high-speed digital circuits among the developing technologies that have lately been offered as replacements to the traditional CMOS. Although efficient QCA-based implementations for a number of binary and decimal arithmetic circuits have been shown, there is still room for significant advancement with the right use of the QCA technology's built-in logic gates. This project suggests an approximation multiplier. approximation computing (AC) is advantageous since it reduces the need for accuracy and delays. Many new nanotechnologies use the majority logic (ML) gate as their logic building block. These adders block erroneous carry-out signals to higher-order computation components to increase accuracy. The multiplier was implemented using partial product reduction (PPR) circuitry. It used the parallel approximation 6:3 compressor. QCA implementations evaluate adder designs. According to the experimental findings, there has been a major advancement over earlier designs. The carry look ahead adder is used to further refine the proposed design.

Key words: QCA majority gates, dadda algorithm, ripple carry adder, carry lookahead adder and approximation method.

1. INTRODUCTION

Traditional CMOS technologies have gradually been restricted in their ability to create Increasing circuit integration has led to the development of VLSI circuits. Despite advancements in semiconductor technology and energy-efficient design methodologies, the power dissipation of computing systems continues to be a severe issue. By sacrificing precision for less complexity and power consumption, approximation computing (AC), a novel computing paradigm at the nanoscale, presents a possible answer for the VLSI sector.

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

To balance the circuit's performance and accuracy, AC makes use because of the program's built-in tolerance for mistakes.

Therefore, AC can be implemented in a vast array of applications and architectures, including signal processing, image recognition, multimedia, and data analysis. A number of novel devices on the nanoscale, such as spin-wave, nanomagnet, and quantum cellular automata (QCA) technology, have been developed recently. Different from traditional Boolean logic, these strategies are grounded in an abstraction known as ML. Nanotechnology uses less energy on an intrinsic level than CMOS does. Additionally, compared to these conventional two-input Boolean logic processes, ML is a more evocative function. Thus, the suggested designs for approximation circuits are implemented in this paper using ML.

Arithmetic units like adders and multipliers are frequently utilised in computing systems. Thus, arithmetic circuit speeds and power consumption are related. big impact on how well computing systems operate. Despite the fact that academics have put out a number of designs for approximation circuits in transistor-based technologies, these designs lose their appeal when used in other non-transistor or technologies that employ alternative logic gates. The system displayed in, as an illustration, employs several XOR operations for carry creation and propagation. But when expressing Since ML operations are inefficient for two- or more-input XOR gates, we propose a pair of MLAFAs and MLAAMs are Approximate multipliers and complete adders based on machine learning, respectively. The following provides a description of these contributions.

1) By using the essential route time and precision of the 2- and 4-bit adders we propose provide a straightforward approach to the design of multibit approximation circuits. were both greatly improved. The proposed adders have a unique structure that makes Errors are less prone to accumulate in long computation sequences.

2) To create a straightforward and effective 8 8 multiplier, we suggest Using Wallace-based proportional partial product reduction (PPR) circuits, we'll show how to implement a roughly parallel 6:3 compressor. Compressing six partial products at once is possible with the proposed compressor, which also features a less complex compared to the conventional 4:2 compressor.

Applications Image processing makes use of arithmetic operations like adding and multiplying. Combining results from the peak signal-to-noise ratio (PSNR) and the

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

structural similarity index (SSIM), the findings are evaluated. Accelerators for machine learning that use minimal amounts of electricity in the form of neural networks are created using multipliers. We carry out research on MLAMs and Image processing using 8-bit multi-level adaptive fuzzy automata. Three aspects of Discussion of the experimental outcomes is provided. Both the MAE and NMED of the suggested designs are significantly smaller than the industry standard. are significantly lower than those of the previous attempts. Both The suggested 8-bit approximation adder decreases MAE by 48.1% and NMED by 50%. The number of logical levels in an effective design is minimized, inverters, and majority gates needed to implement logic. There is a decrease of 16.67% in majority gates, 50% in inverters, and 42.86% in logic levels, in an 8-bit approximation multiplier architecture.

After analyzing the image processing designs, SSIM and PSNR are enhanced. We discovered that there is only one design that produces an endlessly near copy of the original. The suggested the multiplier-based accelerator improved LeNet-5's accuracy on the MNIST dataset to 97.18 percent. when used for machine learning applications. The QCA technology is used to implement the suggested adders. The outcomes of simulations run showing that When utilising the To a large extent, When utilising the QCA Designer-E 2.2, the performance of QCA layouts is inversely related to the cost of logic implementation. design tool. Less logic levels, inverters, and majority gates mean QCA realization uses less power, requires fewer clocking phases, and occupies less space.

2. LITERATURE SURVEY

S. Mittal, "A survey of approximate computing techniques," As performance needs increase and resource budgets plateau, approximation computing—which trades off compute quality with effort—has grown to be not just appealing but also necessary. We give a survey of approximation computing (AC) approaches in this article. We cover ways for employing AC in different processors (including CPU, GPU, and FPGA), as well as programming frameworks for AC. We also cover techniques for locating approximal programmed parts and assessing output quality. To highlight their similarities and differences, we categories these strategies based on a few essential traits. The purpose of this article is to give researchers an understanding how AC approaches operate and encouraging further research in order to establish AC as the default computing model in future systems.

"Nanomagnet logic: An architectural level overview," M. Vacca et al. Accurate computing (AC) benefits from lowering the need for complete precision, which lowers power usage and space requirements. Many developing nanotechnologies use the majority logic

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

(ML) gate is

their primary logic construction component. In this work, multibit adders and multipliers—ML-based arithmetic circuits—are proposed. In order to improve accuracy in order to prevent incorrect carry-out signals from reaching higher-level processors, these adders are manufactured. components.

We used a special partial product reduction (PPR) circuitry to implement the suggestedmultiplier. This circuitry used a 6:3 compressor based on a parallel approximation. Quantum- dot cellular automata (QCA) implementation costs, error metrics, and layouts are examined toevaluate the adder designs. examined. Based on the experimental findings, it can be said that the designs are significantly better than earlier ML-based ones. Image processing and a neuralnetwork (NN) accelerator are both used to further assess the offered designs.

3. EXISTED METHODOLOGY

DADDA MULTIPLIER

Multipliers are essential for various applications as well as the current advanced flag handling. Many scientists' designers have attempted, and continue to attempt, to create multipliers with improved overall parameters such as speed, low power usage, less range, or acombination of these in a single multiplier. DADDA multiplier's underlying idea is based on the geometry of the underlying framework seen in below.

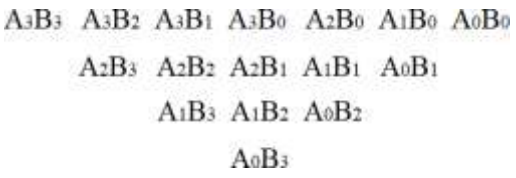

Fig1. Algorithm of dadda multiplier

Dadda multiplier is a reduction method that makes use of the reduced two-row Partial products with as few reduction steps as possible. The [3,2] and [2,2] counters were placed by Dadda in a way that put them on the Critical path's maximum possible length. For a multiplier and a factor that both have N bits, the partial product is N by N. These unfinished products are arranged in a matrix. Dadda underwent a number of transformations to reduce the height of these matrices to a matrix with only two rows.

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

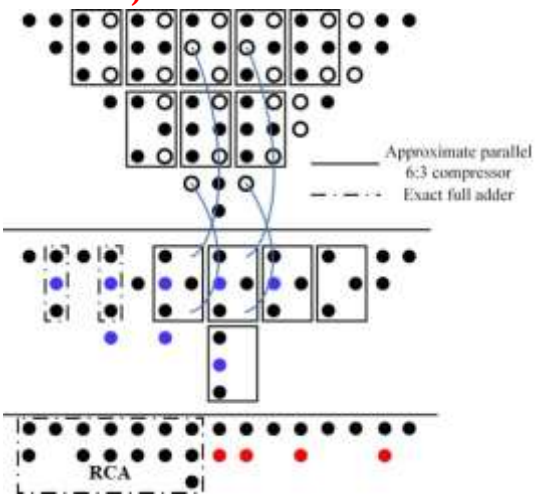

Fig2.Reduction process of an unsigned 8 × 8 multiplier existed PPR circuit

N-BIT BINARY ADDER

As was previously said, A small number of logic circuits are sufficient for the construction of 1-bit binary adders. However, suppose we needed to add two n-bit values. In this case, a Ripple Carry Adder would be constructed from n connected 1-bit full adders that are "cascaded" or connected. In a protracted binary addition, a "ripple carry adder" is made up of "n" cascaded 1-bit full adders, each of which represents a single weighted column. The carry signals cause a "ripple" to appear in the data from right to left (LSB to MSB). data stream. binary adder, hence the name ripple carry adder.

To "add" two 4-bit values, for instance, the first complete adder's outputs would be the addition's first place digit sum (S) and a carry-out bit, the latter of which would be used ascarry-in for the second binary adder. It is possible to add larger integers by connecting the carrier bit output of one full binary adder to the next full adder's input. Similarly, the second binary adder in the series generates a carried-out bit and a summated bit. A 4-bit adder is demonstrated in the code below.

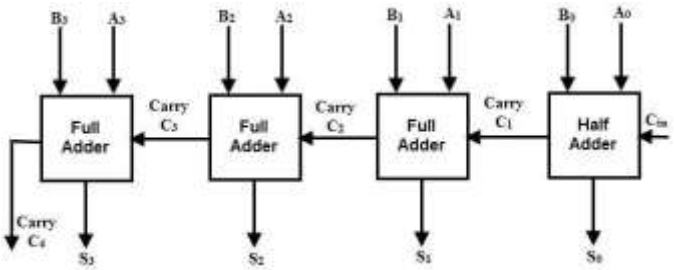

Fig 3. 4-bit binary parallel adder

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

4. PROPOSED METHODOLOGY

In this paper, we suggest a rough multiplier using the dadda multiplier algorithm and the Carry lookahead adder. We need to have a deeper knowledge of the previously created designs because we can see them in the current methodology. In the approach that was in place, the author employed a ripple carry adder to add reduction partial products to the design's final level. The propagation delay is greater because of our routing information for ripple carry adders; thankfully, this will have an impact only on the speed of the design. In order to reduce partial production at the final level of the multiplier design, we are going to suggest a Prefix parallel adder is used in this paper. In essence, parallel prefix adders are fast adders; as an illustration, we will use the carry lookahead adder. Compared to different parallel prefix adders, this CLA has a shorter propagation delay.

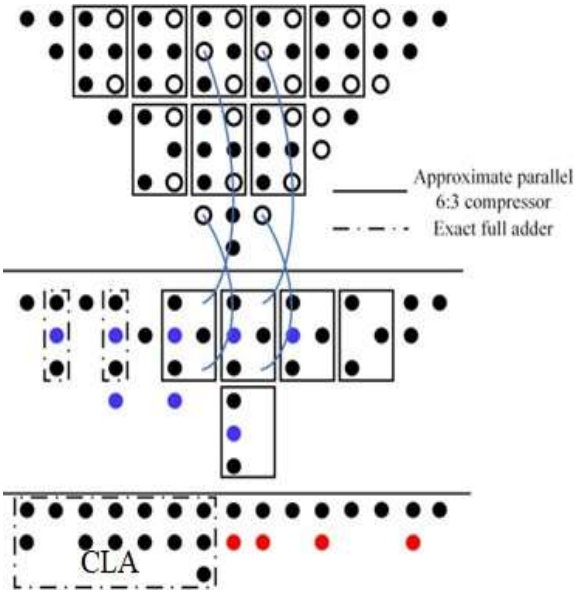

Fig4. Reduction process of an unsigned 8 × 8 multiplier proposed PPR circuit

CARRY LOOKAHEADADDER

By incorporating more complicated hardware, such as a carry look-ahead adder, the propagation delay can be decreased. This design appropriately modifies the ripple carry concept so that the adder's carry logic over fixed groups of bits is simplified to a two-level process. Let's have a serious chat about the layout. By breaking down CLA's overall operation into three separate components, its full functionality may be clearly understood.

1. Pre processing

Computation is required A The prefix form carry look-ahead adder is a parallel-processing.It

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

is typical in the B. pi = Ai *xor* Bi, gi = Ai *and* Bi

2. Carry look ahead network

This section is what gives CLA its unique identity and is the main factor in its outperforming other adders. Here, we calculate the carries for each individual bit. Intermediation signals like group propagate and create are employed.

Pi:j = Pi:k+1 *and Pk:j*

Gi:j = Gi:k+1 *or* (Pi:k+1 *andGk:j*)

3. Post processing

This final phase is shared by all members of this adder family (carry look forwards). It requires the calculation of byte totals.

Si = pi xor Ci-1

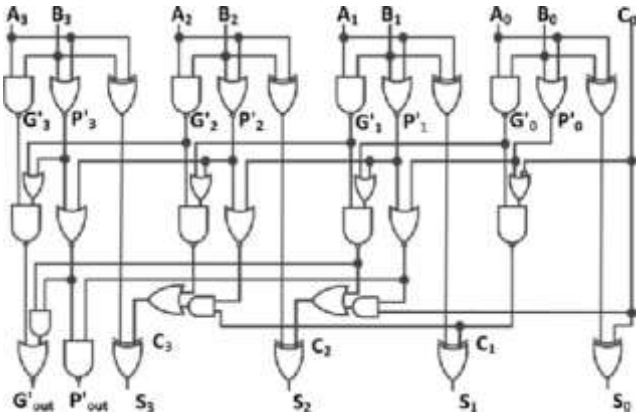

Fig5. 4 bit carry lookahead adder

5. RESULTS

RTL SCHEMATIC

An abbreviation for "register transfer level," which describes the overall layout of the system. It's a tool for gauging how close a building design comes tosome ideal that has yet to be realized. Data transformation into the Programming languages like verilog and vhdl are used to provide a working overview of an architecture. In order to facilitate study, the RTL schematic includes details on the internal connection blocks. The image below depicts the RTL schematic diagram of the planned architecture.

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

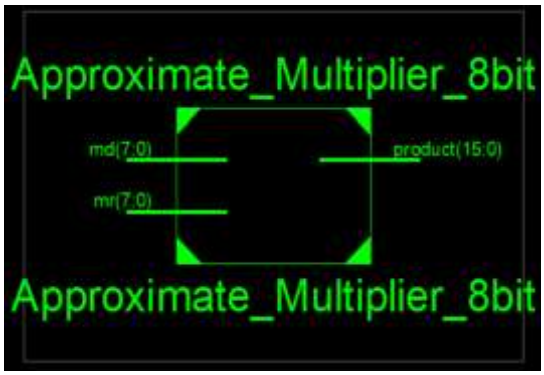

Fig 6. RTL Schematic of existed design

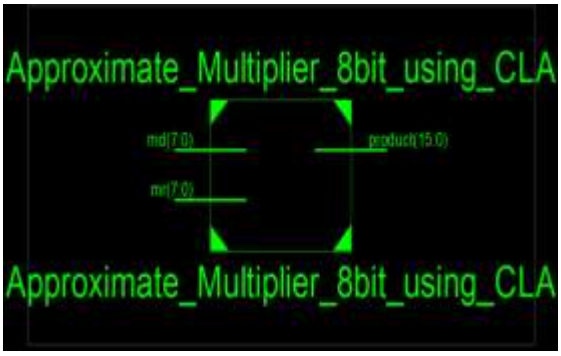

Fig 7. RTL Schematic of proposed design

TECHNOLOGY SCHEMATIC

The LUT format technology schematic is used to depict the architecture; the LUT is then used as the area parameter in VLSI design estimation. In FPGA, look-up tables (LUTs) serve as a square representation of the code's memory allocation.

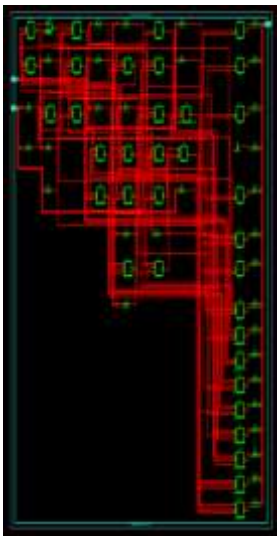

Fig 8. View Technology Schematic of existed design

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

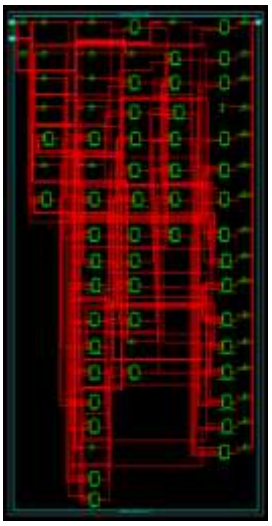

Fig 9. View Technology Schematic of proposed design

SIMULATION

The simulation represents the process, while the schematic is used to check the wiring and assembly. When the user toggles the tool from implantation mode to simulation mode, the simulation window opens on the main screen. The waveforms produced by the simulation are limited to the available window size. In this case, it is adaptable enough to supply multiple radix number systems.

Fig 10. Simulated Waveforms of existed design

Fig11. Simulated Waveforms of proposed design

PARAMETER

When comparing different architectures, VLSI takes into account three factors: area, delay, and power. The HDL language used in this case is verilog, and the tool XILINX 14.7 is used to obtain the area and delay-aware parameter.

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

Parameter	Approximate multiplier using RCA	Approximate multiplier using CLA
Delay (ns)	14.078	11.666

Table1. parameter comparison

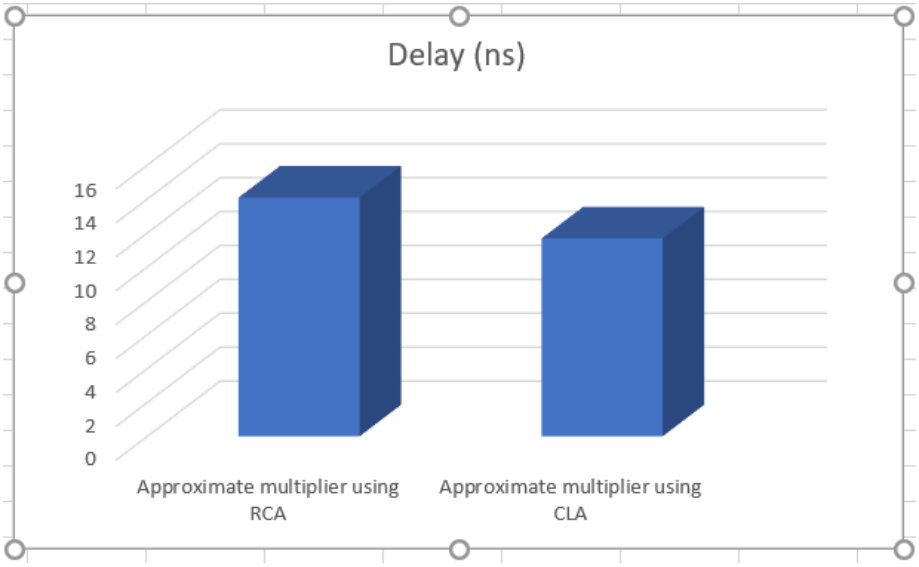

Fig 12. delay comparison bargraph

CONCLUSION

In this project presents the designs, analysis a novel approximate 6:3 compressor and a unique PPR circuit are proposed for the parallel compressor in multiplier using carry lookahead adder. They are able to reduce the delay without significantly degrading the quality of the multiplier. In addition, the proposed compressor is strongly generalizable. It requires only one majority gate with a constant of "1," which is an OR gate that is excellently implemented in other logic primitives. Compared with other existing approximate designs, there is a significant improvement in terms of delay.

REFERENCES

[1] Q. Xu, M. Todd, and S. K. Nam, "Approximate computing: A survey," IEEE Design Test, vol. 33, no. 1, pp. 8–22, Feb. 2016.

[2] S. Mittal, "A survey of techniques for approximate computing," ACM Comput. Surv., vol. 48, no. 4, pp. 1–33, 2016.

[3] S. Venkataramani, S. T. Chakradhar, K. Roy, and A. Raghunathan, "Approximate computing and the quest for computing efficiency," in Proc. 52nd Annu. Design Autom.

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

Conf., Jun. 2015, pp. 1–6.

[4] W. Liu, F. Lombardi, and M. Schulte, "A retrospective and prospective view of approximate computing," Proc. IEEE, vol. 108, no. 3, pp. 394–399, Mar. 2020.

[5] C. S. Lent and P. D. Tougaw, "A device architecture for computing with quantum dots," Proc. IEEE, vol. 85, no. 4, pp. 541–557, Apr. 1997.

[6] M. Vacca et al., "Nanomagnet logic: An architectural level overview," in Field-Coupled Nanocomputing. Cham, Switzerland: Springer, 2014, pp. 223–256.

[7] A. Khitun and K. L. Wang, "Nano scale computational architectures with spin wave bus," Superlattices Microstruct., vol. 38, no. 3, pp. 184–200, 2005.

[8] W. Liu, L. Qian, C. Wang, H. Jiang, J. Han, and F. Lombardi, "Design of approximate radix-4 booth multipliers for error-tolerant computing," IEEE Trans. Comput., vol. 66, no. 8, pp. 1435–1441, Aug. 2017.

[9] Z. Yang, A. Jain, J. Liang, J. Han, and F. Lombardi, "Approximate XOR/XNOR-based adders for inexact computing," in Proc. 13th IEEE Int. Conf. Nanotechnol. (IEEE-NANO), Aug. 2013, pp. 690–693.

[10] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 4, pp. 850–862, Apr. 2010.