ESP-IJCEET
ESP International Journal of Communication Engineering & Electronics Technology
ISSN: 2583-9217 / Volume 1 Issue 3 October 2023 / Page No: 23-33
Paper Id: IJCEET-V1I3P104 / Doi: 10.56472/25839217/ IJCEET-V1I3P104

Original Article

Area Efficient Nano Approximate an using QCA Designer

S. Jasmine[1], D. Ajitha[2]

[1,2]*SRM Madurai College for Engineering and Technology, Tamilnadu, India*

Received Date: 12 September 2023 Revised Date: 25 September 2023 Accepted Date: 05 October 2023

Abstract: *This study delves into the realm of approximate computing, where computational errors and imprecisions are embraced as acceptable trade-offs to achieve more efficient and resource-conserving solutions. The research focuses on the design of approximate adders within the framework of Quantum-dot Cellular Automata (QCA), a promising technology for ultra-low-power computation. Approximate adders are circuits that introduce controlled errors into addition operations to reduce resource consumption while maintaining adequate precision. Two distinct approaches are explored in this study: the utilization of threshold gates and the adaptation of Arithmetic Logic Units (ALUs) to create QCA-based approximate adders. These approaches aim to strike a balance between precision and computational resources, ultimately offering more energy-efficient solutions. Additionally, novel majority inversion-based approximate adder architecture is proposed, emphasizing significant area reduction. The study's findings shed light on the potential of approximate computing in QCA-based systems, paving the way for simplified, energy-efficient, and error-tolerant computational solutions.*

Keywords: *Digital Circuits, Locking Technique, Computing, Quantum-Dot.*

I. INTRODUCTION

In today's rapidly evolving world of computation, the pursuit of energy efficiency and resource optimization is paramount. While precision and accuracy have long been the driving forces behind computational methodologies, there is a growing recognition that not all applications necessitate uncompromising exactitude. In many scenarios, computational errors and imprecisions can be tolerated, yielding outcomes that are not only understandable but also significantly more efficient in terms of computational resources.

This recognition has given rise to the concept of "approximate computing," a paradigm that intentionally introduces controlled errors and imprecisions into computational processes to achieve computational tasks more efficiently. By reducing the precision demands, approximate computing opens up avenues for simplifying circuits, reducing energy consumption, minimizing delay, and conserving area in various computational systems [2], [3]. It offers a pragmatic approach to balance computational demands with available resources and has found application across multiple design levels, spanning from the transistor-level to algorithmic, architectural, and even software design.

In the context of hardware design, one of the key components where approximate computing principles can be effectively applied is the adder circuit. Adders are fundamental building blocks for arithmetic operations, and their efficiency can greatly impact the overall performance of computational systems. Quantum-dot Cellular Automata (QCA), an emerging technology that leverages the quantum properties of electrons for ultra-low-power computation, provides a promising platform for exploring approximate adder designs.

Approximate adders in the QCA domain are circuits designed to add two numbers with a certain degree of error, consciously sacrificing precision in exchange for resource optimization. This error-tolerant approach in QCA circuits can be achieved through the utilization of quantum gates like CNOT gates, Toffoli gates, and Fredkin gates. The primary objective is to create circuits that execute addition operations at a higher speed while demanding fewer resources.

This research delves into the realm of approximate computing within QCA technology, with a particular focus on the design of approximate adders. It explores various approaches, including the use of threshold gates and the adaptation of

Arithmetic Logic Units (ALUs), to achieve a delicate balance between precision and resource efficiency in QCA-based computational systems. Additionally, a novel majority inversion-based approximate adder architecture is proposed, aiming to significantly reduce the area requirements while maintaining an acceptable degree of precision.

Through these investigations, this study contributes valuable insights into the practical implementation of approximate computing principles in the context of QCA, offering a promising avenue for the development of simpler, more energy-efficient computational solutions in various application domains.

II. LITERATURE SURVEY

- Sanchez-Macian et al analyzed the impact of defects in Quantum-dot Cellular Automata (QCA) majority gates on image processing when using approximate adders. They evaluated variations in error distance metrics due to defects and explored strategies for defect mitigation, including combining approximate and exact adders and selectively introducing fault-tolerant majority gates in different bits.

- Perri et al introduced a design for approximate binary adders optimized for Quantum-dot Cellular Automata (QCA) technology. Their 16-bit architecture demonstrated significantly lower energy-delay-product and area occupancy compared to state-of-the-art competitors, with significantly improved accuracy.

- Bahar et al presented a single-layer binary discrete cosine transform (BinDCT) implemented in QCA, featuring low-complexity combinational and sequential logic elements. Their designs outperformed previous ones in terms of cell complexity, area coverage, and energy dissipation.

- Zhang et al proposed a one-bit approximate full adder based on majority logic and extended their design to multi-bit approximate full adders. They also demonstrated the application of these designs in Quantum-dot Cellular Automata (QCA).
- Aravinth et al discussed the development of an approximate adder using Quantum-dot Cellular Automata (QCA) technology. Their focus was on basic arithmetic processes, particularly binary digit addition, emphasizing simplicity and approximate accuracy.

- Sharma et al introduced a novel reversible circuit design acting as a full adder, even parity, and odd parity generator. Their design minimized garbage output, required few clock zones, and could be adapted for various logic gates.

- Tripathi et al planned an energy and cost-efficient 4-bit Ripple Carry Adder (RCA) using an efficient full adder design. They employed a triplet design technique, achieving a smaller QCA cell count compared to previous designs.

- Prasanna et al presented efficient combinational arithmetic circuits, including full adders, full subtractors, and comparators, designed using Quantum-dot Cellular Automata (QCA) technology.

- Kalpana et al designed full adders and full subtractors using QCA with a focus on minimal QCA cell count and power analysis.

- Swetha et al designed a QCA-based full adder logic circuit using a five-input majority gate. They also introduced a multilayer 1-bit Arithmetic Logic Unit (ALU) structure for implementing both logical and arithmetic operations.

- U. B. Joy et al developed a QCA-based Full Adder (FA) cell utilizing a combination of QCA majority gates and inverters. They compared the performance of their QCA FA cell with existing designs.

- Z. Chu et al presented multi-digit Binary Coded Decimal (BCD) adder designs based on three-input exclusive-OR (XOR3) and majority (MAJ) gates, which are commonly used in financial, commercial, and industrial computing.

- P. U. Rao et al implemented a BCD adder in QCA using five-input majority gates and compared its delay with existing systems using Xilinx software.

- Vanaraj et al compared different gate types used in adders and subtractors for their energy dissipation. They proposed novel designs for adders and subtractors using energy-efficient gates.

- B. Sen et al explored the design of fault-tolerant Quantum-dot Cellular Automata (QCA) tiles, including a fault-tolerant tile implementing majority logic and tiles for coupled majority-minority functions.

- M. Maity et al introduced a fault-tolerant reversible multiplexer (MUX) design using a parity-preserving Fredkin gate. Their design was efficient in power dissipation and fault tolerance.

- B. Sen et al designed a robust universal logic gate in Quantum-dot Cellular Automata (QCA) that realized majority and minority functions simultaneously with high fault tolerance, using alternative QCA tile structures.

- D. Bhowmik et al studied the fault-tolerance properties of the 5-input majority gate, which is essential for fault-tolerant Quantum Cellular Automata circuits.

- M. S. Latha Gade et al proposed an area-efficient fault-tolerant QCA reversible 2:1 MUX design with superior performance compared to existing architectures, evaluated for functional accuracy and energy dissipation.

- B. Sen et al focused on designing a reversible Arithmetic Logic Unit (ALU) in QCA based on the reversible QCA structure (RQCA), aiming to maximize throughput.

- Z. D. Patitz et al presented reliable QCA cell structures for designing single clock-controlled majority gates with tolerance to radius of effect-induced faults, important for carry look-ahead adders.

- M. Raj et al proposed a majority-logic-based self-checking feature to enhance fault tolerance in QCA adders.

- Roohi et al developed a novel Parity-Preserving Reversible Gate (PPRG) using Quantum-dot Cellular Automata (QCA) technology, emphasizing fault tolerance and reversibility.

- Sánchez-Macián et al analyzed the impact of QCA majority gate defects on image processing using approximate adders, exploring defect mitigation strategies.

- Jing Huang et al analyzed defect tolerance properties of QCA when employing tiles with molecular QCA cells, studying different input/output arrangements and their impact on defect tolerance.

- T. Sultana et al proposed a new 3-dot-based QCA architecture, including wiring logics, basic gates, XOR gates, and efficient QCA-based half adders and full adders.

- D. Tripathi et al focused on constructing an energy and cost-efficient 4-bit QCA RCA using an efficient full adder design and a triplet design technique.

- M. Javid et al introduced a robust QCA adder design, considering various types of defects in Quantum-dot Cellular Automata technology.

- M. Pathania et al presented a novel XOR gate for QCA, reducing cell count and area for half adder circuits, emphasizing efficiency.
- D. Abedi et al designed a QCA serial adder (QSA) with reduced cell count and area, using a coplanar clock-zone-based crossover design for QCA full adders.

III. PROPOSED SYSTEM

Computational errors and imprecisions can be effectively accommodated in certain applications, where the outcomes, although not entirely precise, remain comprehensible and beneficial for human understanding. Indeed, by intentionally reducing precision to a reasonable extent, various circuit parameters such as the number of devices, energy consumption, delay, and area can be minimized. Consequently, approximate computing stands as a viable solution for swift computation in applications tolerant of errors, leading to the creation of simpler circuits with improved energy efficiency [2], [3]. It's worth noting that approximate computing, as a design approach, can be applied across various levels of abstraction in the design process, encompassing aspects from transistor-level design to logic, algorithmic, architectural, and software design.

An approximate adder, within the context of Quantum-dot Cellular Automata (QCA), is a circuit designed to perform addition with a certain degree of error while conserving resources compared to a full-precision adder. Typically, the design of an approximate adder in QCA involves the utilization of quantum gates like CNOT gates, Toffoli gates, and Fredkin gates. The fundamental concept guiding the development of an approximate adder in QCA is to strike a balance between precision and simplicity, resulting in a circuit that is faster and demands fewer resources.

One approach to crafting an approximate adder in QCA entails the use of threshold gates, which can represent a range of values. These gates function by determining whether the input signal exceeds or falls below a predefined threshold value. By cascading multiple threshold gates, it becomes possible to construct a straightforward and efficient approximate adder.

Another method for designing an approximate adder in QCA involves incorporating an Arithmetic Logic Unit (ALU). An ALU is a digital circuit that carries out arithmetic and logical operations. In the realm of QCA, an ALU can be repurposed to create an approximate adder by truncating the precision of the input numbers.

In the broader scope, the design of an approximate adder in QCA necessitates a meticulous consideration of the trade-off between precision and computational resources. The objective is to fashion a circuit that delivers the requisite degree of precision while minimizing the resources essential for its implementation.

In this particular research endeavor, novel approximate adders based on majority inversion are proposed to achieve significant area reduction. The carry output is determined through majority functions involving three primary inputs, while the sum output is generated by inverting the carry, as illustrated in the figure. The approximated truth table for the proposed adder is provided in the table.

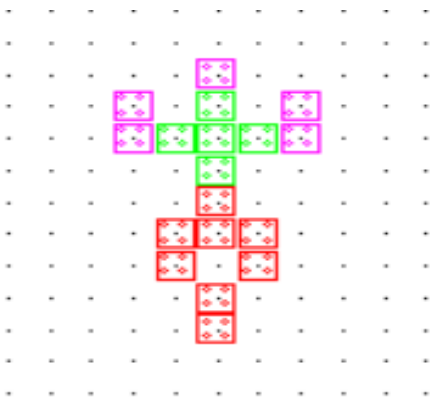

Figure 1: Proposed QCA Adder-1

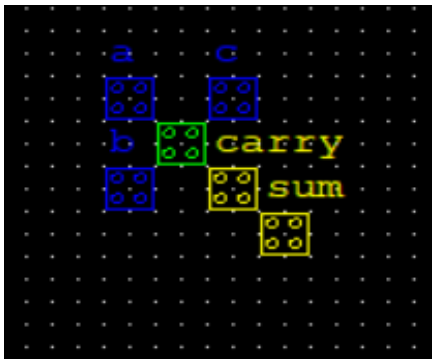

Figure 2: Proposed QCA Adder-2

Table 1: Proposed Adder Truth Table

A	B	Cin	Carry	sum
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

We have designed eight-bit approximate adders by cascading two four-bit approximate adders as shown in give below figure:

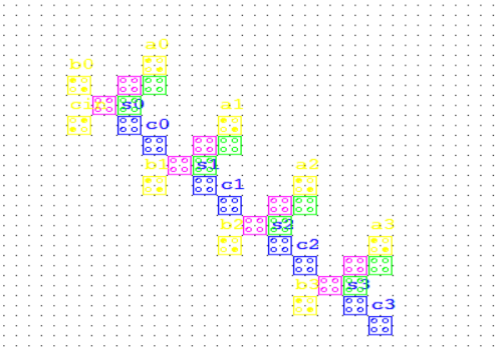

Figure 3: Eight-Bit Approximate Adders by Cascading Two Four-Bit

A. Designing a QCA layout for an approximate adder involves the following steps:

- Determine the size and precision of the input numbers to be added.
- Determine the required precision of the approximate adder.
- Determine the resources available for the QCA layout.
- Choose a suitable QCA cell to use as the basic building block for the layout.
- Choose a suitable QCA clocking scheme.
- Design the logic gates required for the approximate adder.
- Layout the logic gates using the chosen QCA cell and clocking scheme.

B. Here are some additional details on the above steps:

Step 1: Determine the size and precision of the input numbers to be added. The size of the input numbers will determine the number of qubits required to represent them. For example, if the input numbers are 8-bit integers, then 8 qubits will be required

to represent each input number. The precision of the input numbers will determine the degree of error that can be tolerated in the approximate adder.

Step 2: Determine the required precision of the approximate adder. The required precision of the approximate adder will depend on the application. For example, if the approximate adder is being used for a simple arithmetic operation, such as calculating the sum of two numbers, then a low degree of precision may be acceptable. However, if the approximate adder is being used for a more complex operation, such as a financial calculation, then a higher degree of precision may be required.

Step 3: Determine the resources available for the QCA layout. The resources available for the QCA layout will depend on the physical constraints of the system being used to implement the layout. For example, the size of the QCA chip, the number of qubits available, and the clocking scheme available will all impact the resources available for the layout.

Step 4: Choose a suitable QCA cell to use as the basic building block for the layout. QCA cells are the basic building blocks of a QCA layout. There are many different types of QCA cells available, each with different properties, such as fan-out, fan-in, and propagation delay. The choice of QCA cell will depend on the requirements of the approximate adder.

Step 5: Choose a suitable QCA clocking scheme. QCA circuits require a clocking scheme to synchronize the signals in the circuit. There are many different QCA clocking schemes available, such as the majority gate clocking scheme and the ripple gate clocking scheme. The choice of QCA clocking scheme will depend on the requirements of the approximate adder.

Step 6: Design the logic gates required for the approximate adder. The logic gates required for the approximate adder will depend on the chosen QCA cell and clocking scheme. Typically, the logic gates will include threshold gates, which are used to represent a range of values.

Step 7: Layout the logic gates using the chosen QCA cell and clocking scheme. Once the logic gates have been designed, they can be laid out using the chosen QCA cell and clocking scheme. The layout should be optimized for the available resources, such as the number of qubits and the clocking speed.

C. Here is QCA Layout Step for an Approximate Adder:

- The input numbers are represented by qubits, with one qubit per bit of precision.
- The qubits are input into threshold gates, which are used to represent a range of values.
- The output of the threshold gates is passed through a series of XOR gates to perform the addition.
- The result of the addition is output as qubits, with one qubit per bit of precision.

IV. RESULTS AND DISCUSSION

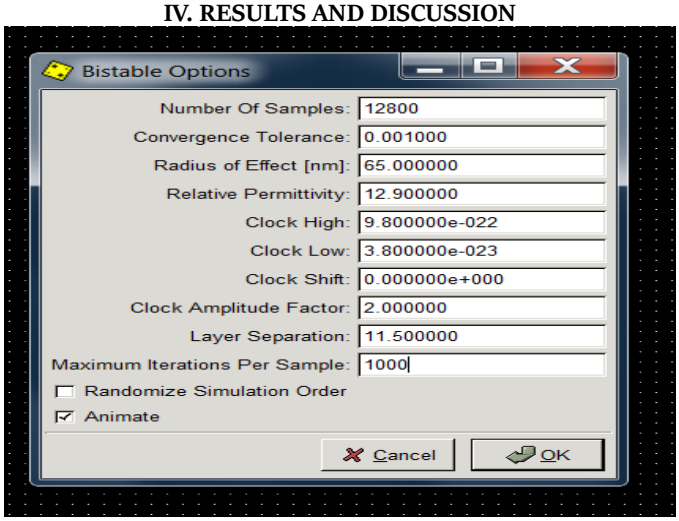

Figure 4: Simulation Parameters

In QCA Designer, can create bi-stable cells by selecting the appropriate option in the Cell Editor and configuring the cell's layout and behavior for approximate adder design as shown in figure 5.1. By using bi-stable cells in your designs, you can create circuits that can store and manipulate digital information, enabling a wide range of computational and communication applications

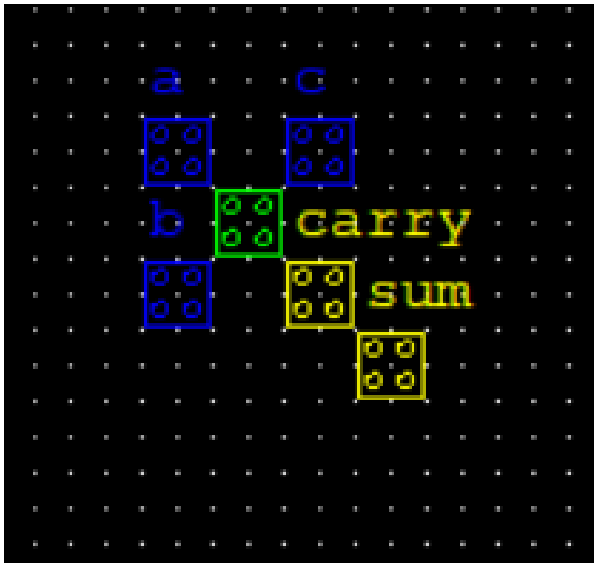

Figure 5: QCA Layout of Proposed Adders

Layout design of adders in QCA involves arranging QCA cells in a specific configuration to implement a desired function or circuit as shown in Figure 5.2. QCA Designer is a software tool that simplifies this process, allowing users to create custom layouts for QCA circuits. To begin the layout design process in QCA Designer, users can create a new project and select the appropriate QCA cell library for their application.

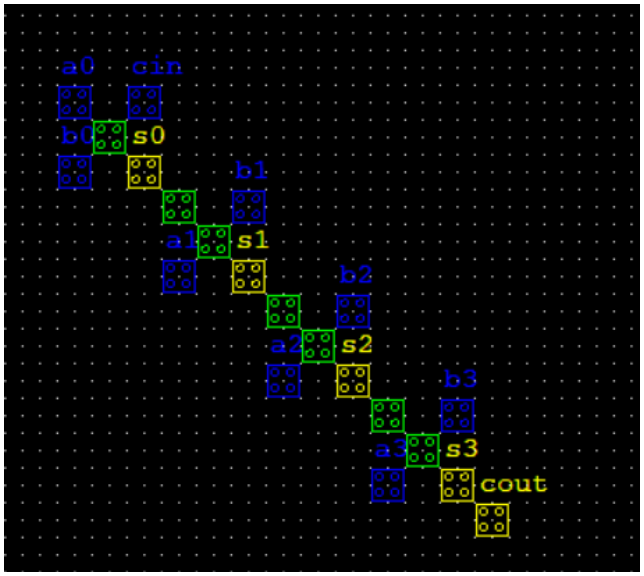

Figure 6: Proposed Multi Bit Adder

Layout design of proposed 4 bit adder in QCA involves arranging QCA cells in a specific configuration to implement a desired function or circuit as shown in Figure 5.3. QCA Designer is a software tool that simplifies this process, allowing users to create custom layouts for QCA circuits. To begin the layout design process in QCA Designer, users can create a new project and select the appropriate QCA cell library for their application.

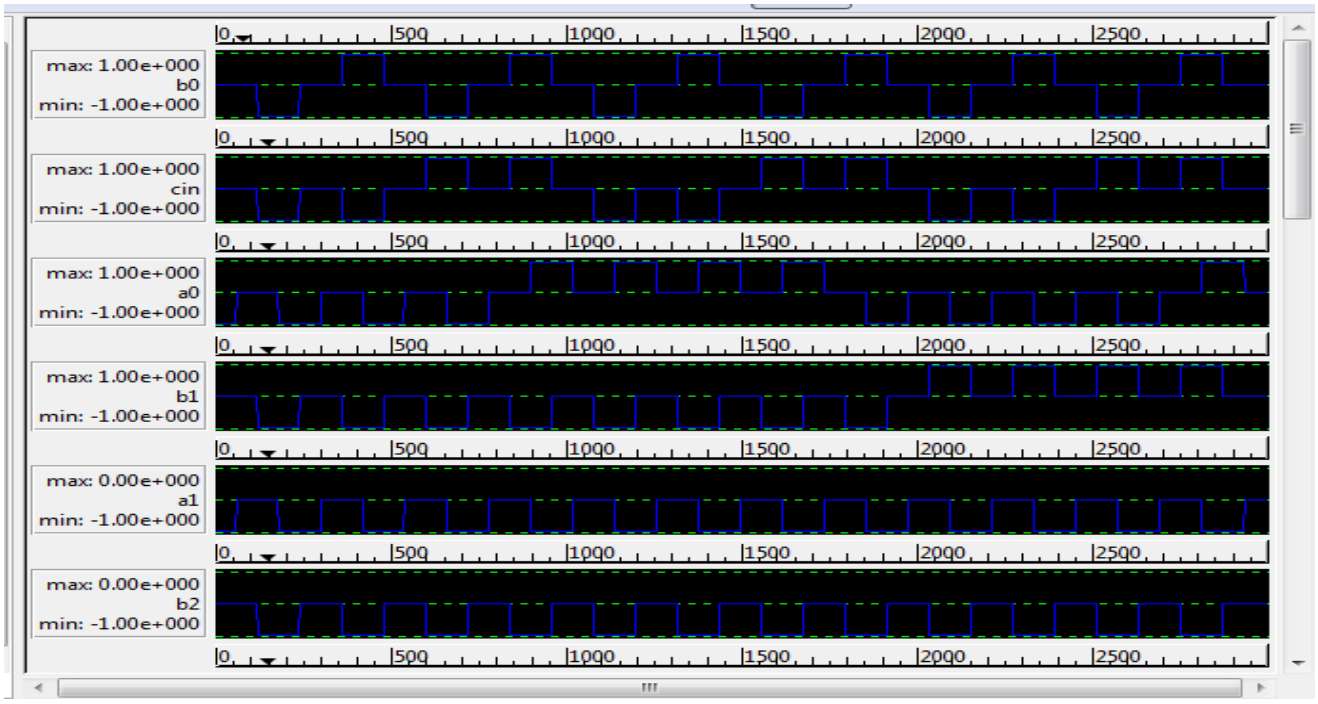

Figure 7: Input Sequence

We can verify input waveforms by simulating your QCA layout design. To verify input and output waveforms, select the "Simulation" tab. By verifying the input and output waveforms in QCA Designer, the layout of proposed design is functioning as intended and refine it as necessary.

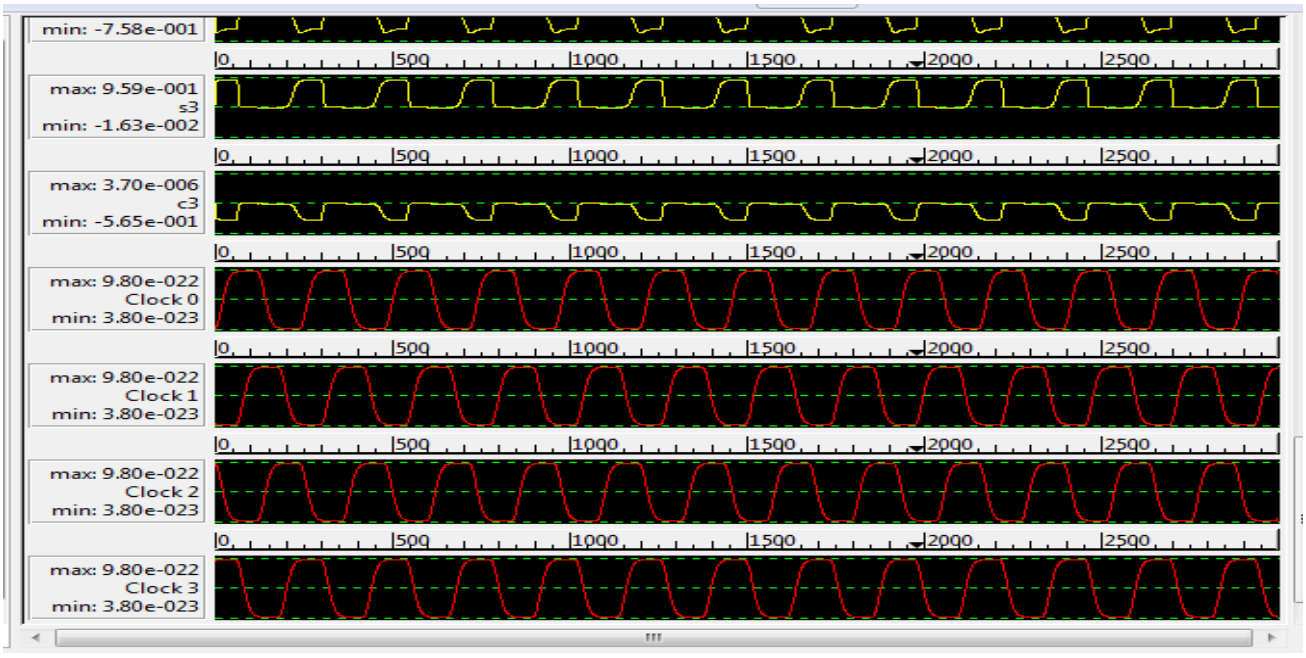

Figure 8: Output Sequence

We can verify output waveforms by simulating your QCA layout design. To verify input and output waveforms, select the "Simulation" tab. By verifying the input and output waveforms in QCA Designer, the layout of proposed design is functioning as intended and refine it as necessary.

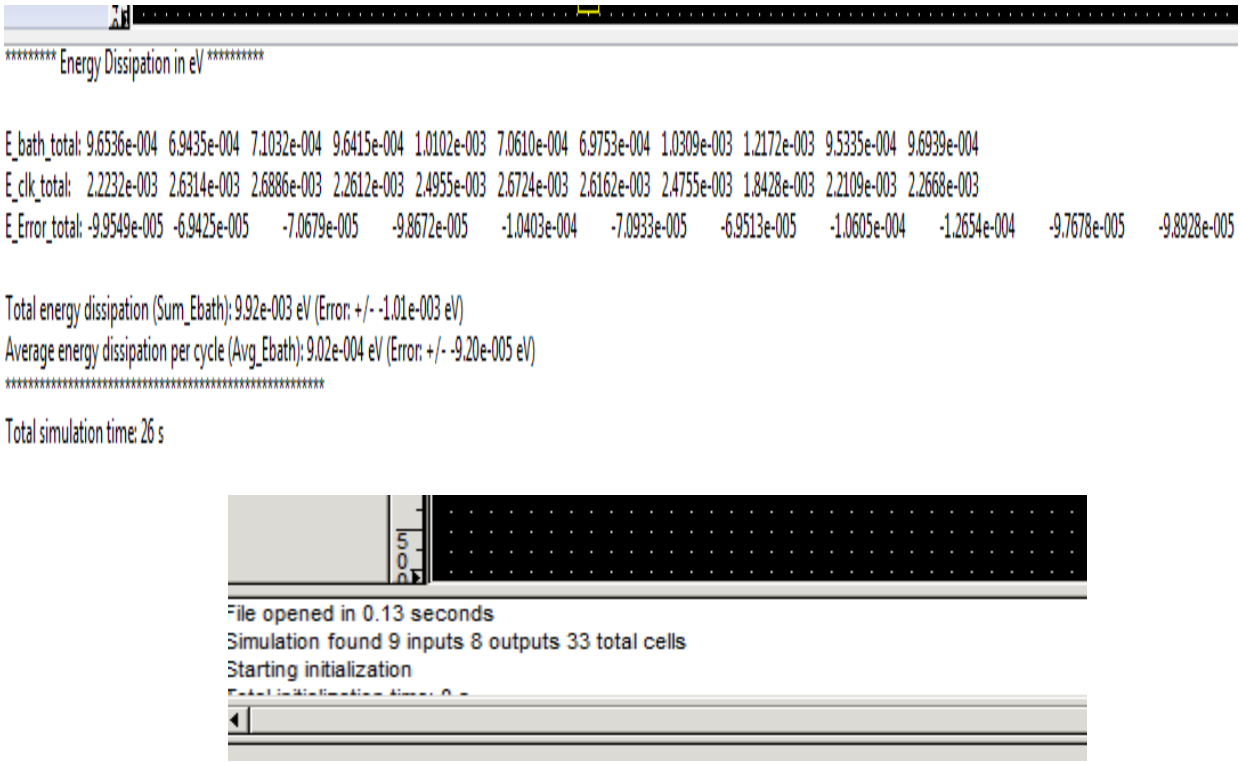

Figure 9: Performance Analysis

Table 2: Comparison Table

Parameter	Existing	Phase-1	Phase-2
Area -Dots	52	33	21
Delay -S	43	26	12
Energy -J	13.0	9.20	3.14

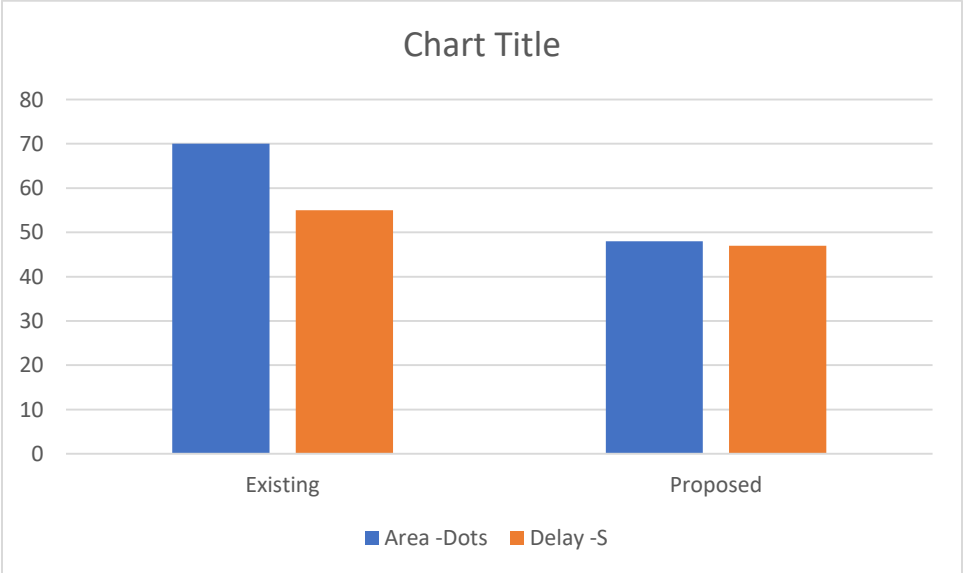

Figure 10: Performance Analysis

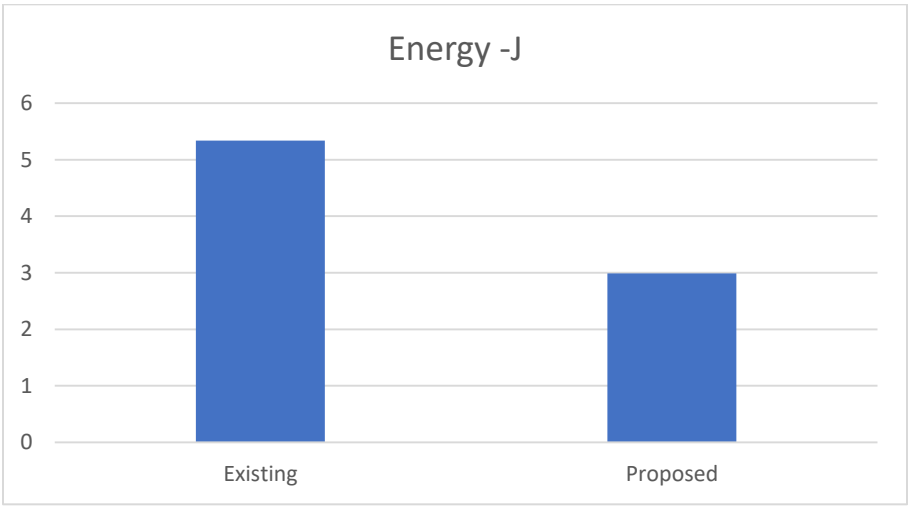

Figure 11: Performance Analysis

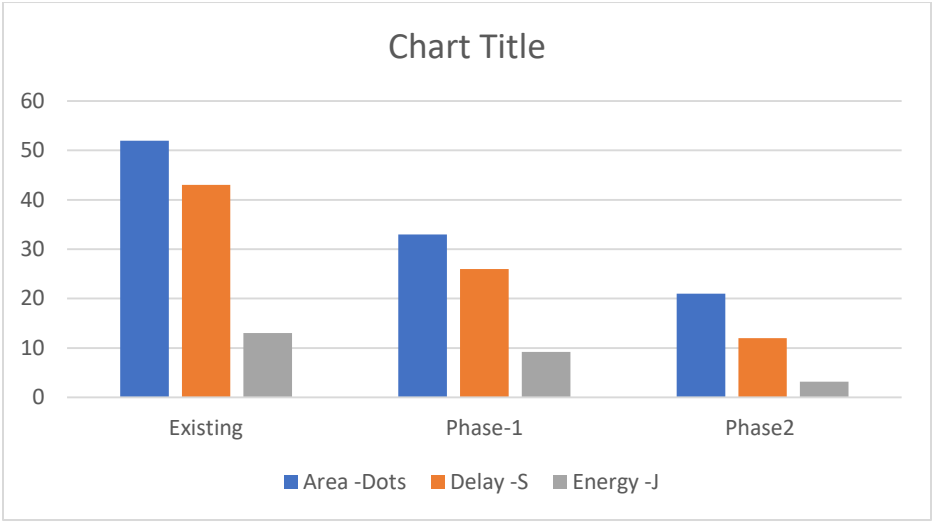

Figure 12: Performance Analysis

The number of dots and delay are important factors to consider when designing QCA circuits. In general, a circuit with fewer dots can be advantageous, as it can lead to reduced circuit complexity and increased speed. The proposed design shows minimum dot requirement and delay in comparison with existing design

V. CONCLUSION

In the pursuit of energy-efficient and resource-conserving computing solutions, this research has explored the design of approximate adders within the Quantum-dot Cellular Automata (QCA) paradigm. These adders introduce controlled errors into addition operations to achieve substantial reductions in resource utilization while maintaining an acceptable level of precision. Two prominent approaches were examined: employing threshold gates and adapting Arithmetic Logic Units (ALUs) for QCA-based approximate addition. These approaches demonstrated the trade-off between precision and computational resources, highlighting the potential for energy-efficient computing in error-tolerant applications. Moreover, the introduction of a novel majority inversion-based approximate adder architecture showcased a substantial reduction in area requirements, further advancing the state-of-the-art in approximate computing within QCA technology. This research contributes valuable insights into the practical implementation of approximate computing in QCA systems, offering a promising avenue for the development of simpler, more energy-efficient computational solutions in various application domains.

VI. REFERENCES

[1] A. T. Vanaraj, M. Raj and L. Gopalakrishnan, "Energy-Efficient Coplanar Adder and Subtractor in QCA," 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), 2020, pp. 539-544, doi: 10.1109/ICSSIT48917.2020.9214274.

[2] P. U. Rao and P. Niranjan, "Design of efficient BCD adder with five input majority generators for quantum cellular automata," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), 2020, pp. 1566-1573, doi: 10.1109/ICISS49785.2020.9316010.

[3] Z. Chu, Z. Li, Y. Xia, L. Wang and W. Liu, "BCD Adder Designs Based on Three-Input XOR and Majority Gates," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 68, no. 6, pp. 1942-1946, June 2021, doi: 10.1109/TCSII.2020.3047393.

[4] U. B. Joy, S. Chakraborty, S. Tasnim, M. S. Hossain, A. H. Siddique and M. Hasan, "Design of an Area Efficient Quantum Dot Cellular Automata Based Full Adder Cell Having Low Latency," 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2021, pp. 689-693, doi: 10.1109/ICREST51555.2021.9331135.

[5] K. Swetha, K. L. Krishna, J. V. S. Sowmya, D. S. Reddy, G. Pravallika and G. A. Kumar, "Area Efficient Multilayer Arithmetic Logic Unit Implementation in Quantum-dot Cellular Automata," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021, pp. 584-589, doi: 10.1109/ICICV50876.2021.9388584.

[6] S. Perri, F. Spagnolo, F. Frustaci and P. Corsonello, "Accuracy Improved Low-Energy Multi-Bit Approximate Adders in QCA," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 68, no. 11, pp. 3456-3460, Nov. 2021, doi: 10.1109/TCSII.2021.3077669.

[7] K. Kalpana, B. Paulchamy, S. Chinnapparaj, K. Mahendrakan and A. AbdulHayum, "A Novel design of Nano scale TIEO based single layer full adder and full subractor in QCA paradigm," 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021, pp. 575-582, doi: 10.1109/ICICCS51141.2021.9432098.

[8] S. Prasanna A, B. Madhava Reddy and R. S R, "Design of Combinational Arithmetic Circuits using Quantum Dot Cellular Automata," 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021, pp. 117-122, doi: 10.1109/ICOEI51242.2021.9453069.

[9] D. Tripathi and S. Wairya, "An Energy Dissipation and Cost Optimization of QCA Ripple Carry Adder," 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN), 2021, pp. 760-765, doi: 10.1109/SPIN52536.2021.9566068.

[10] S. Sharma and V. K. Sharma, "Design of Full Adder and Parity Generator Based on Reversible Logic," 2021 Emerging Trends in Industry 4.0 (ETI 4.0), 2021, pp. 1-4, doi: 10.1109/ETI4.051663.2021.9619268.

[11] B. Aravinth and L. J. A. Marcilin, "Implementation of coplanar approximate adders in QCA," 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 2016, pp. 680-684, doi: 10.1109/WiSPNET.2016.7566219.

[12] T. Zhang, W. Liu, E. McLarnon, M. O'Neill and F. Lombardi, "Design of Majority Logic (ML) Based Approximate Full Adders," 2018 IEEE International Symposium on Circuits and Systems (ISCAS), 2018, pp. 1-5, doi: 10.1109/ISCAS.2018.8350962.

[13] A. N. Bahar and K. A. Wahid, "Design and Implementation of Approximate DCT Architecture in Quantum-Dot Cellular Automata," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 28, no. 12, pp. 2530-2539, Dec. 2020, doi: 10.1109/TVLSI.2020.3013724.

[14] S. Perri, F. Spagnolo, F. Frustaci and P. Corsonello, "Accuracy Improved Low-Energy Multi-Bit Approximate Adders in QCA," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 68, no. 11, pp. 3456-3460, Nov. 2021, doi: 10.1109/TCSII.2021.3077669.

[15] A. Sanchez-Macian, A. Martin-Toledano, J. A. Bravo-Montes, F. Garcia-Herrero and J. A. Maestro, "Reducing the Impact of Defects in Quantum-Dot Cellular Automata (QCA) Approximate Adders at Nano Scale," in IEEE Transactions on Emerging Topics in Computing, doi: 10.1109/TETC.2021.3136204.