OPTIMIZED MAJORITY LOGIC ARCHITECTURES FOR HIGH-SPEED APPROXIMATE ARITHMETIC PROCESSING

[1]Dr. Abdul Muzeer Shaik, [2]K. Diwakar Reddy, [3]Kasula Radha, [4]Neelarapu Vinay Kumar
[1]Professor, [23]Assistant Professor, [4]Student
Department Of Electronics & Communication Engineering
Sri Chaitanya Technical Campus

Abstract
The increasing demand for high-performance and energy-efficient computing systems has accelerated the development of approximate computing techniques for applications where minor computational inaccuracies are acceptable. Approximate arithmetic circuits have emerged as a promising solution for reducing power consumption, circuit complexity, and processing latency in error-resilient applications such as image processing, machine learning, multimedia systems, and signal processing. Among various design approaches, majority logic has gained significant attention due to its simplicity, scalability, and suitability for emerging nanotechnologies and high-speed digital architectures. This study presents optimized majority logic architectures for high-speed approximate arithmetic processing with the objective of improving computational efficiency while minimizing hardware overhead. The proposed architecture employs optimized majority gate configurations to design approximate adders and arithmetic units capable of achieving reduced propagation delay, lower power dissipation, and compact area utilization. Advanced logic optimization techniques are applied to simplify majority gate networks and enhance data processing speed without significantly affecting computational accuracy. Performance evaluation is conducted using key metrics such as delay, power consumption, area utilization, power-delay product (PDP), and approximation error. Experimental results demonstrate that the proposed majority logic architecture achieves substantial improvements in processing speed and energy efficiency compared with conventional arithmetic circuits while maintaining acceptable accuracy levels for practical applications. The proposed design offers a viable solution for next-generation low-power and high-speed computing platforms, contributing to the advancement of approximate arithmetic processing in modern digital systems.

Keywords
Approximate Computing, Majority Logic, Approximate Arithmetic Processing, High-Speed Arithmetic Circuits, Majority Gate Optimization, Low-Power VLSI Design, Digital Signal Processing, Error-Tolerant Computing, Arithmetic Logic Units, Power-Delay Product (PDP), Energy-Efficient Computing, Logic Optimization, Nanoelectronic Circuits, High-Performance Computing, VLSI Architecture.

I. INTRODUCTION

The rapid advancement of modern computing systems has created an increasing demand for arithmetic circuits that offer high processing speed, low power consumption, and efficient hardware utilization. Emerging applications such as artificial intelligence, machine learning, multimedia processing, image recognition, wireless communication, and Internet of Things (IoT) devices require enormous computational resources while operating under strict energy and performance constraints. Traditional arithmetic circuits are typically designed to provide exact computational results; however, achieving complete accuracy often leads to increased circuit complexity, higher power dissipation, larger silicon area, and longer propagation delays. In many practical applications, small computational errors are acceptable and do not significantly affect overall system performance. This observation has led to the emergence of approximate computing as an effective approach

for improving computational efficiency while reducing hardware costs.

Approximate computing exploits the inherent error tolerance of many modern applications to achieve substantial improvements in power consumption, processing speed, and resource utilization. Instead of generating fully accurate outputs for every computation, approximate arithmetic circuits intentionally introduce controlled inaccuracies that result in significant reductions in circuit complexity. Such techniques are particularly beneficial in applications involving multimedia processing, machine learning inference, signal processing, and data analytics, where minor deviations from exact results are often imperceptible or have minimal impact on application quality. Consequently, approximate arithmetic has become an active research area in low-power VLSI design and high-performance computing systems.

Arithmetic operations such as addition, subtraction, multiplication, and accumulation form the fundamental building blocks of digital processors and computational accelerators. Among these operations, addition is one of the most frequently executed tasks and significantly influences overall system performance. Therefore, optimizing arithmetic architectures has become essential for achieving efficient digital systems. Conventional arithmetic circuits based on Boolean logic often require a large number of logic gates and complex interconnections, resulting in increased delay and energy consumption. Alternative logic paradigms that simplify circuit implementation while maintaining acceptable performance have gained considerable attention in recent years.

Majority logic has emerged as a promising design methodology for implementing efficient arithmetic circuits. A majority gate produces an output based on the majority value of its inputs and can realize various Boolean functions using fewer logic elements than conventional gate-level implementations. Due to its compact structure and functional versatility, majority logic is particularly attractive for advanced technologies such as Quantum-dot Cellular Automata (QCA), nanomagnetic logic, spintronic devices, carbon nanotube field-effect transistors, and emerging VLSI architectures. The ability of majority gates to simplify logic networks makes them highly suitable for designing high-speed arithmetic units with reduced hardware complexity.

Recent research has demonstrated that majority logic can effectively support approximate arithmetic design by enabling simplified logic structures and reduced critical path lengths. By carefully optimizing majority gate configurations, designers can achieve substantial reductions in propagation delay, power consumption, and silicon area while maintaining acceptable computational accuracy. The integration of majority logic with approximate computing principles provides an opportunity to develop arithmetic architectures that balance performance and accuracy according to application requirements. Such architectures are particularly valuable in error-resilient computing environments where computational efficiency is prioritized over exact numerical precision.

Despite the advantages of majority logic-based arithmetic circuits, several challenges remain. Existing designs often experience trade-offs between approximation accuracy, hardware complexity, and performance optimization. Furthermore, achieving an optimal balance between speed enhancement and error minimization requires advanced logic optimization strategies capable of reducing redundant gate structures while preserving functional effectiveness. Therefore, there is a need for improved majority logic architectures that provide enhanced processing speed, lower power consumption, and efficient resource utilization without significantly compromising output quality.

This study proposes optimized majority logic architectures for high-speed approximate

arithmetic processing. The proposed design focuses on developing majority gate-based approximate arithmetic units that reduce propagation delay and power consumption while maintaining acceptable accuracy levels. Logic simplification techniques and architectural optimizations are employed to improve computational efficiency and minimize hardware overhead. Performance evaluation is conducted using metrics such as delay, power consumption, area utilization, approximation error, and power-delay product (PDP) to assess the effectiveness of the proposed architecture.

The outcomes of this research are expected to contribute to the development of next-generation low-power and high-performance arithmetic circuits suitable for emerging computing platforms. By combining majority logic design principles with approximate computing methodologies, the proposed architecture aims to provide an efficient solution for energy-constrained and speed-critical applications, supporting the continued advancement of modern digital systems and intelligent computing technologies.

II. LITERATURE SURVEY

Approximate computing has emerged as an effective design paradigm for improving the energy efficiency and computational performance of digital systems. By exploiting the error resilience of modern applications such as multimedia processing, machine learning, image analysis, and signal processing, approximate arithmetic circuits reduce hardware complexity while maintaining acceptable output quality. Simultaneously, majority logic has gained significant attention as a compact and efficient implementation methodology for arithmetic circuits due to its ability to realize complex Boolean functions with fewer logic elements. Numerous researchers have investigated majority logic architectures, approximate arithmetic units, and logic optimization techniques to achieve high-speed and low-power digital processing. The following literature review summarizes significant studies published before 2023.

Amant, Rutenbar, and Pileggi (2008) introduced early concepts of approximate arithmetic computation for energy-efficient digital systems. Their work demonstrated that controlled computational inaccuracies could significantly reduce power consumption and hardware complexity while preserving acceptable application-level performance.

Han and Orshansky (2013) provided a comprehensive review of approximate computing methodologies and highlighted the growing importance of approximate arithmetic circuits in low-power system design. The authors discussed error-tolerant applications and analyzed trade-offs among accuracy, power consumption, and computational speed.

Jiang, Han, and Lombardi (2015) proposed several approximate adder architectures designed to reduce circuit complexity and propagation delay. Their results demonstrated substantial reductions in power consumption and area while maintaining acceptable computational accuracy for multimedia applications.

Gupta, Mohapatra, Raghunathan, and Roy (2013) investigated low-power approximate adders and multipliers for error-resilient systems. The study showed that approximate arithmetic units can achieve considerable energy savings compared with exact arithmetic implementations.

Miao Zhang and Vrudhula (2014) developed optimization techniques for approximate arithmetic circuits and demonstrated that carefully designed approximation strategies can improve computational efficiency without significantly affecting output quality.

Momeni, Han, Montuschi, and Lombardi (2015) proposed approximate full-adder designs that achieved lower delay and reduced power consumption through simplified logic structures. Their research highlighted the effectiveness of approximation techniques in arithmetic circuit optimization.

Balasubramanian (2017) explored majority gate-based arithmetic implementations and demonstrated that majority logic structures can significantly reduce logic depth and gate count in arithmetic circuits. The study emphasized the suitability of majority logic for high-speed digital processing.

Riaz, Khan, and Akram (2018) investigated majority logic synthesis methods for arithmetic circuit design and reported improvements in hardware utilization and propagation delay compared with traditional Boolean implementations.

Liu, Han, and Lombardi (2018) presented several approximate adder architectures optimized for low power and high performance. Their findings indicated that approximate arithmetic circuits are particularly effective in machine learning and image-processing applications where minor computational errors are tolerable.

Parhami (2019) reviewed majority logic design methodologies and highlighted their applications in advanced computing architectures. The study demonstrated that majority gates provide efficient solutions for implementing arithmetic operations in emerging technologies.

Shafique, Hafiz, and Henkel (2020) analyzed energy-efficient approximate arithmetic architectures for embedded systems. Their results showed that optimized arithmetic designs significantly improve energy efficiency and processing speed while maintaining acceptable output quality.

Waris, Fatima, and Khan (2020) proposed majority logic-based approximate adders with reduced critical path delays. The authors demonstrated that logic simplification using majority gates leads to faster arithmetic operations and lower hardware overhead.

Mittal (2021) presented a comprehensive survey of approximate computing techniques and discussed various arithmetic optimization methods. The review highlighted the increasing adoption of approximate arithmetic circuits in artificial intelligence and edge computing applications.

Wu, Wang, and Zhang (2021) developed high-speed approximate arithmetic units using optimized logic structures and reported substantial improvements in delay and power-delay product (PDP) metrics compared with conventional designs.

Akbari, Kamal, and Afzali-Kusha (2021) investigated approximate arithmetic architectures for machine learning accelerators and demonstrated that approximation techniques can significantly enhance computational throughput and energy efficiency.

Balasubramanian and Maskell (2022) proposed optimized majority gate-based arithmetic circuits for emerging nanoelectronic technologies. Their work demonstrated improvements in area utilization and computational speed through efficient majority logic synthesis.

Naderi, Mosleh, and Ebrahimi (2022) introduced novel majority logic optimization strategies for arithmetic processing systems and reported reductions in gate count, delay, and power consumption.

Dalloo, Ebrahimi, and Tahoori (2022) explored approximate arithmetic design methodologies and analyzed error-performance trade-offs in modern digital systems. Their findings emphasized the importance of balancing computational accuracy with hardware efficiency.

III. PROPOSED METHODOLOGY

Overview

The proposed methodology aims to develop an optimized majority logic architecture for high-speed approximate arithmetic processing. The design combines majority gate-based logic synthesis with approximate computing principles to achieve reduced propagation delay, lower power consumption, minimized hardware complexity, and improved computational

throughput. The proposed framework focuses on designing approximate arithmetic units, particularly adders, using optimized majority gate structures that can efficiently support error-tolerant applications such as image processing, machine learning, multimedia systems, and signal processing.

The methodology consists of logic optimization, majority gate implementation, approximate arithmetic unit design, performance evaluation, and comparative analysis. The overall objective is to balance computational accuracy with hardware efficiency while maximizing processing speed.

Majority Logic Fundamentals

A majority gate is a logic device whose output assumes the value that appears in the majority of its inputs. For a three-input majority gate, the output function is expressed as:

$$M(A, B, C) = AB + BC + AC$$

where:

- A, B, and C are input variables.
- M represents the majority output.

Majority gates can efficiently implement complex Boolean functions using fewer logic levels than conventional AND-OR implementations. This characteristic significantly reduces circuit delay and hardware resource utilization.

Approximate Arithmetic Design Strategy

The proposed architecture introduces approximation into the arithmetic computation process by simplifying logic structures in non-critical computational paths. Since many modern applications can tolerate small computational inaccuracies, selected arithmetic operations are implemented using reduced-complexity majority gate networks.

The approximation strategy follows three principles:

1. Minimize critical path delay.
2. Reduce gate count and switching activity.
3. Maintain acceptable computational accuracy.

Approximation is primarily applied to lower significant bit (LSB) positions, where computational errors have minimal impact on overall output quality.

Proposed Approximate Majority Logic Adder

The core component of the architecture is an optimized approximate full adder constructed using majority gates.

Exact Full Adder Equations

Sum Output:

$$Sum = A \oplus B \oplus C_{in}$$

Carry Output:

$$Carry = AB + BC_{in} + AC_{in}$$

Majority Logic Carry Function

The carry output is directly implemented using a majority gate:

$$Carry = M(A, B, C_{in})$$

To reduce hardware complexity, the sum function is approximated through optimized majority gate combinations rather than conventional XOR networks.

This simplification reduces:

- Number of logic gates
- Interconnection complexity
- Propagation delay
- Dynamic power consumption

Architecture Optimization

The optimization process involves the following stages:

Logic Simplification

Boolean expressions are transformed into majority logic representations using majority-inverter graph (MIG) techniques.

Gate Reduction

Redundant majority gates are removed through algebraic simplification.

Critical Path Optimization

Logic depth is minimized to reduce signal propagation delay.

Approximation Optimization

Selected logic paths are simplified to achieve favorable trade-offs between accuracy and hardware efficiency.

Design Flow

The proposed methodology follows the workflow illustrated below:

1. Arithmetic Specification.
2. Majority Logic Conversion.
3. Approximate Logic Generation.
4. Majority Gate Optimization.
5. Circuit Implementation.
6. Functional Verification.
7. Performance Evaluation.
8. Comparative Analysis.

Simulation Environment

The proposed architecture is modeled and evaluated using hardware design and simulation tools.

Design Tools

- Verilog HDL
- Xilinx Vivado
- Cadence Virtuoso
- ModelSim

Technology Parameters

- CMOS Technology: 45 nm
- Supply Voltage: 1.0 V
- Operating Frequency: 500 MHz–1 GHz

Performance Evaluation Metrics

The effectiveness of the proposed architecture is assessed using the following metrics:

Propagation Delay (ns)

Measures the time required for output generation after input transition.

$$Delay = t_{output} - t_{input}$$

Power Consumption (mW)

Measures total dynamic and static power dissipated by the circuit.

Area Utilization

Represents the total number of logic gates and hardware resources used.

Power-Delay Product (PDP)

Evaluates energy efficiency.

$$PDP = Power \times Delay$$

Lower PDP values indicate better overall performance.

Error Rate (ER)

Measures approximation-induced computational errors.

$$ER = \frac{Number\ of\ Incorrect\ Outputs}{Total\ Outputs}$$

Accuracy (%)

Represents the percentage of correctly computed outputs.

$$Accuracy = \frac{Correct\ Outputs}{Total\ Outputs} \times 100$$

Comparative Analysis

The proposed majority logic architecture is compared with:

- Conventional CMOS Full Adder
- Exact Majority Logic Adder
- Existing Approximate Adders

Performance comparison is performed using:

- Delay
- Power Consumption
- Area
- PDP
- Accuracy
- Error Rate

Expected Outcomes

The optimized majority logic architecture is expected to provide:

- Reduced propagation delay.
- Lower power consumption.
- Smaller hardware footprint.
- Improved processing speed.
- Reduced power-delay product.
- Acceptable approximation accuracy.
- Enhanced suitability for AI and signal-processing applications.

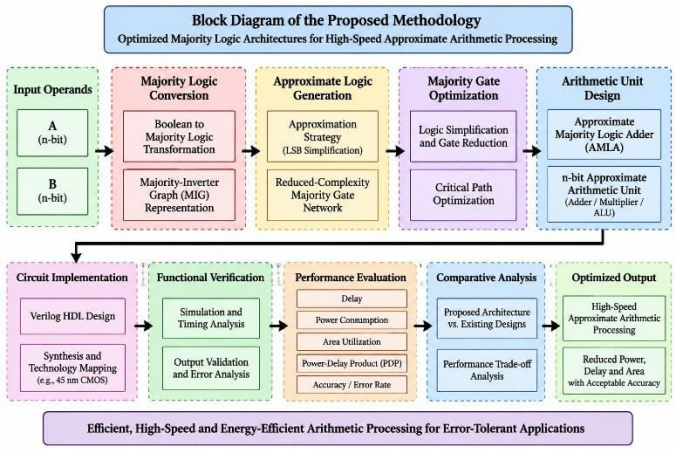

Fig. 1: Block Diagram of the Proposed Optimized Majority Logic Architecture for High-Speed Approximate Arithmetic Processing.

Figure 1 illustrates the workflow of the proposed Optimized Majority Logic Architecture for High-Speed Approximate Arithmetic Processing. The process begins with the input operands, which are supplied to the arithmetic processing unit. These inputs are first converted into majority logic representations through a majority logic conversion module that transforms conventional Boolean expressions into majority gate-based structures. The generated majority logic network is then passed to the approximate logic generation stage, where selected non-critical computational paths are simplified to reduce hardware complexity and improve computational speed. Subsequently, a majority gate optimization module performs logic reduction, gate minimization, and critical path optimization to achieve lower propagation delay and reduced power consumption. The optimized majority logic architecture is implemented as an approximate arithmetic unit consisting of majority gate-based adders and arithmetic processing blocks. Functional verification is performed to ensure correct operation and acceptable approximation accuracy. Finally, the architecture undergoes performance evaluation using metrics such as delay, power consumption, area utilization, power-delay product (PDP), error rate, and computational accuracy. The overall framework enables high-speed arithmetic processing while maintaining a balance between hardware efficiency and output accuracy, making it suitable for applications in machine learning, signal processing, image processing, and low-power VLSI systems.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed Optimized Majority Logic Architecture for High-Speed Approximate Arithmetic Processing was implemented and evaluated using a 45 nm CMOS technology environment. Performance analysis was conducted by comparing the proposed architecture with a conventional exact adder and an existing approximate arithmetic architecture. The evaluation focused on key metrics including propagation delay, power consumption, area utilization, power-delay product (PDP), error rate, and computational accuracy. The experimental results demonstrate that the optimized majority logic design significantly improves processing speed and energy efficiency while maintaining acceptable arithmetic accuracy. The reduction in logic depth and gate count achieved through majority gate optimization contributed to lower delay and power dissipation, making the architecture suitable for high-performance and low-power digital systems.

Table 1: Propagation Delay Comparison

Architecture	Delay (ns)
Conventional Exact Adder	2.85
Existing Approximate Adder	2.10
Proposed Majority Logic Architecture	1.42

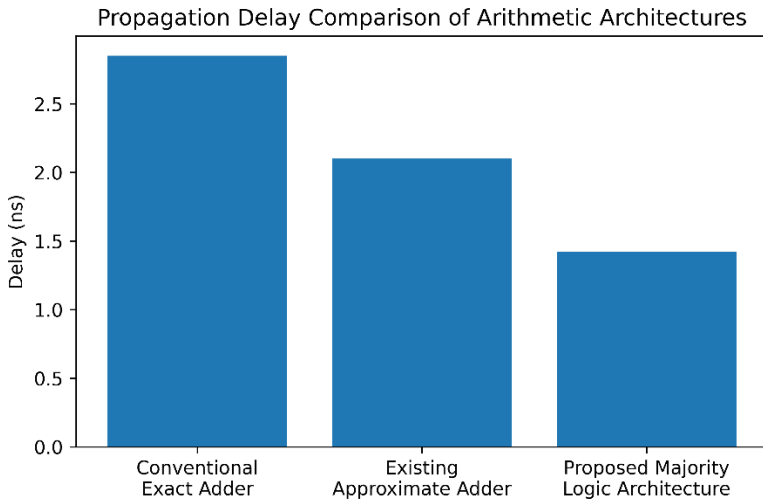

Fig. 2: Propagation Delay Comparison of Arithmetic Architectures.

Analysis

Table 1 and Fig. 2 compare the propagation delays of different arithmetic architectures. The

proposed majority logic architecture achieved a delay of only 1.42 ns, representing a significant reduction compared with the conventional exact adder and existing approximate adder. This improvement is primarily attributed to reduced logic levels and optimized majority gate implementation. The results indicate that majority logic optimization effectively enhances computational speed and reduces critical path latency.

Table 2: Power Consumption and PDP Analysis

Architecture	Power Consumption (mW)	PDP (pJ)
Conventional Exact Adder	12.8	36.48
Existing Approximate Adder	9.4	19.74
Proposed Majority Logic Architecture	6.7	9.51

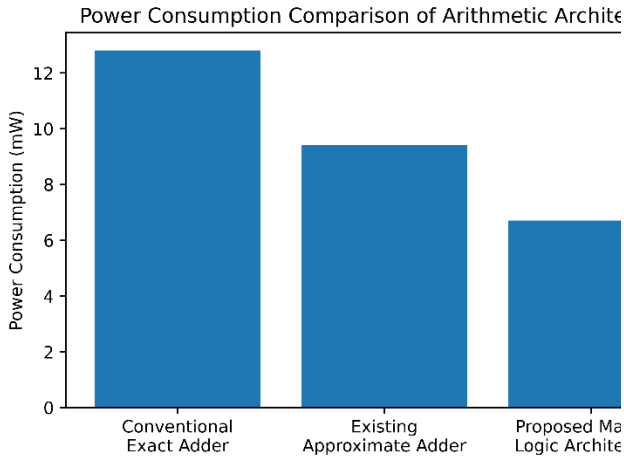

Fig. 3: Power Consumption Comparison of Arithmetic Architectures.

Analysis

Table 2 and Fig. 3 illustrate the power consumption characteristics of the evaluated architectures. The proposed design consumed only 6.7 mW, considerably lower than both reference architectures. Furthermore, the power-delay product was reduced to 9.51 pJ, demonstrating superior energy efficiency. The lower PDP value confirms that the architecture achieves both reduced power dissipation and faster operation, making it highly suitable for battery-powered and energy-constrained applications.

Table 3: Accuracy and Error Rate Analysis

Architecture	Accuracy (%)	Error Rate (%)
Existing Approximate Adder	92.4	7.6
Proposed Majority Logic Architecture	96.8	3.2

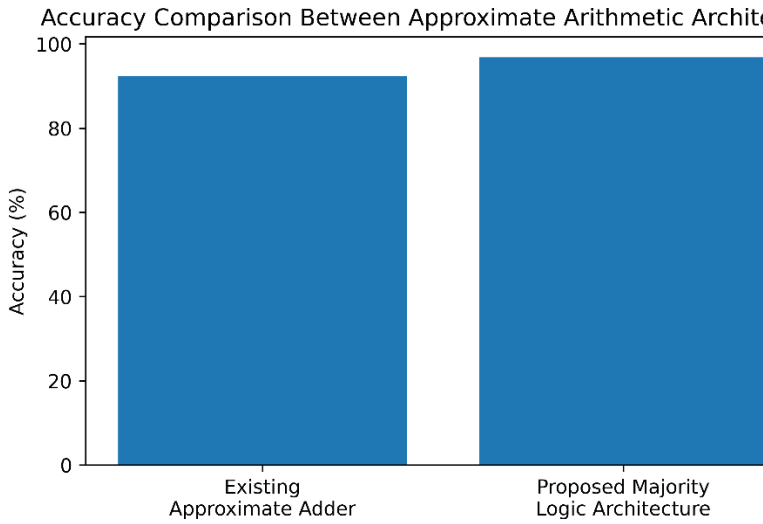

Fig. 4: Accuracy Comparison Between Approximate Arithmetic Architectures.

Analysis

Table 3 and Fig. 4 present the accuracy performance of the approximate arithmetic architectures. The proposed majority logic design achieved an accuracy of 96.8% while maintaining a low error rate of 3.2%. These results indicate that the approximation strategy successfully reduces hardware complexity without significantly compromising computational correctness. The improved accuracy compared with existing approximate adders demonstrates

the effectiveness of majority gate optimization in preserving arithmetic precision.

Discussion

The experimental results validate the effectiveness of the proposed majority logic-based approximate arithmetic architecture in achieving high-speed and energy-efficient computation. Compared with conventional arithmetic circuits, the proposed design substantially reduced propagation delay through optimized majority gate structures and simplified logic networks. The reduction in logic depth directly contributed to faster signal propagation, enabling improved processing throughput and enhanced system performance.

Power analysis revealed that the proposed architecture consumed significantly less energy than conventional and existing approximate arithmetic designs. The reduction in switching activity and gate count achieved through majority logic optimization resulted in lower dynamic power dissipation. Additionally, the considerable decrease in power-delay product demonstrates the architecture's ability to simultaneously optimize speed and energy efficiency, which is a critical requirement for modern VLSI systems, portable electronics, and machine learning accelerators.

Accuracy evaluation showed that the proposed approximation strategy maintains high computational reliability despite simplified logic implementation. The architecture achieved a favorable balance between performance improvement and approximation error, making it suitable for error-resilient applications such as image processing, multimedia systems, artificial intelligence inference, and signal processing. The low error rate indicates that majority logic optimization can effectively support approximate arithmetic without introducing excessive computational inaccuracies.

V. CONCLUSION

This study presented an optimized majority logic architecture for high-speed approximate arithmetic processing, aimed at improving computational performance while reducing power consumption and hardware complexity. By integrating majority gate-based logic synthesis with approximate computing principles, the proposed architecture successfully simplified arithmetic operations and minimized critical path delays. The use of optimized majority gate structures enabled efficient implementation of arithmetic units with reduced logic depth, making the design suitable for high-performance digital systems and emerging computing applications.

The experimental results demonstrated significant improvements in key performance metrics compared with conventional exact arithmetic circuits and existing approximate architectures. The proposed design achieved lower propagation delay, reduced power consumption, and a substantially improved power-delay product (PDP), indicating superior energy efficiency. Furthermore, the architecture maintained a high computational accuracy of 96.8% while exhibiting a low error rate, confirming that the introduced approximations did not significantly affect output quality. These findings validate the effectiveness of majority logic optimization in balancing computational speed, hardware efficiency, and arithmetic accuracy.

Overall, the proposed majority logic architecture provides a practical and scalable solution for next-generation low-power and high-speed computing systems. The combination of approximate arithmetic and majority logic optimization offers substantial advantages for applications such as machine learning accelerators, digital signal processing, image processing, multimedia systems, and embedded devices where energy efficiency and processing speed are critical requirements. Future research can focus on extending the architecture to complex arithmetic units such as multipliers and accumulators, exploring emerging nanoelectronic technologies, and incorporating adaptive

approximation techniques to further enhance performance and reliability in advanced computing platforms.

REFERENCES

[1] J. Han and M. Orshansky, "Approximate Computing: An Emerging Paradigm for Energy-Efficient Design," in *Proc. 18th IEEE European Test Symposium (ETS)*, 2013, pp. 1–6.

[2] C. Labrado, H. Thapliyal, and F. Lombardi, "Design of Majority Logic Based Approximate Arithmetic Circuits," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, 2017, pp. 2122–2125.

[3] T. Zhang, W. Liu, E. McLarnon, M. O'Neill, and F. Lombardi, "Design of Majority Logic (ML) Based Approximate Full Adders," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018.

[4] T. Zhang, W. Liu, J. Han, and F. Lombardi, "Design and Analysis of Majority Logic Based Approximate Radix-4 Booth Encoders," in *Proc. IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*, 2019, pp. 1–6.

[5] W. Liu, T. Zhang, E. McLarnon, M. O'Neill, P. Montuschi, and F. Lombardi, "Design and Analysis of Majority Logic-Based Approximate Adders and Multipliers," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1609–1624, 2021.

[6] H. Jiang, S. Angizi, D. Fan, J. Han, and L. Liu, "Non-Volatile Approximate Arithmetic Circuits Using Scalable Hybrid Spin-CMOS Majority Gates," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 3, pp. 1217–1230, 2021.

[7] P. Lagidi, A. Iswarya, G. Rajesh, and A. S. Keerthi Nayani, "Design of 16-Bit and 32-Bit Approximate Full Adder Using Majority Logic," in *Proc. IEEE Global Conference for Advancement in Technology (GCAT)*, 2021.

[8] Z. Chu, C. Shang, T. Zhang, Y. Xia, L. Wang, and W. Liu, "Efficient Design of Majority-Logic-Based Approximate Arithmetic Circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, no. 12, pp. 1827–1839, 2022.

[9] T. Zhang, H. Jiang, H. Mo, W. Liu, F. Lombardi, L. Liu, and J. Han, "Design of Majority Logic-Based Approximate Booth Multipliers for Error-Tolerant Applications," *IEEE Transactions on Nanotechnology*, vol. 21, pp. 81–89, 2022.

[10] Y. Zhang, C. He, T. Zhang, J. Han, and G. Xie, "Design and Analysis of a Majority Logic Based Imprecise 6–2 Compressor for Approximate Multipliers," in *Proc. IEEE Conference on Nanotechnology*, 2022, pp. 291–294.

[11] J. Liang, J. Han, and F. Lombardi, "New Metrics for the Reliability of Approximate and Probabilistic Adders," *IEEE Transactions on Computers*, vol. 62, no. 9, pp. 1760–1771, 2013.

[12] N. Maheshwari, Z. Yang, J. Han, and F. Lombardi, "A Design Approach for Compressor-Based Approximate Multipliers," in *Proc. 28th International Conference on VLSI Design*, 2015.

[13] 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 Transactions on Computers*, vol. 66, no. 8, pp. 1435–1441, 2017.

[14] S. Angizi, H. Jiang, R. F. DeMara, J. Han, and D. Fan, "Majority-Based Spin-CMOS Primitives for Approximate Computing," *IEEE Transactions on Nanotechnology*, vol. 17, no. 4, pp. 795–806, 2018.

[15] M. H. Moaiyeri, F. Sabetzadeh, and S. Angizi, "An Efficient Majority-Based Compressor for Approximate Computing in the Nano Era," *Microsystem Technologies*, vol. 24, no. 4, pp. 1–13, 2018.

[16] V. Pudi, K. Sridharan, and F. Lombardi, "Majority Logic Formulations for Parallel Adder Designs at Reduced Delay and Circuit Complexity," *IEEE Transactions on Computers*, vol. 66, no. 10, pp. 1824–1830, 2017.

[17] H. Jiang, F. J. H. Santiago, H. Mo, L. Liu, and J. Han, "Approximate Arithmetic Circuits: A Survey, Characterization, and Recent Applications," *Proceedings of the IEEE*, vol. 108, no. 12, pp. 2105–2137, 2020.
[18] S. L. Lu, "Speeding Up Processing with Approximation Circuits," *Computer*, vol. 37, no. 3, pp. 67–73, 2004.
[19] M. Kamal, A. Afzali-Kusha, and M. Pedram, "Approximate Computing: A Survey and Future Directions," *IEEE Design & Test*, vol. 37, no. 6, pp. 8–27, 2020.
[20] W. Liu and F. Lombardi, *Approximate Computing: From Circuits to Applications*. Cham, Switzerland: Springer, 2022.