International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 12 Issue 6, June-2023, Impact Factor: 7.957

Energy-Efficient SRAM-Based In-Memory Computing for Boolean Logic Operations

Ratan Babu Telusoori[1], Dr. Vikas Jaiman[2]

[1]Research Scholar, Department of Electronics & Communication Engineering, JS University, Shikohabad, UP
[2]Supervisor, Department of Electronics & Communication Engineering, JS University, Shikohabad, UP

ABSTRACT

In-Memory Computing (IMC) has emerged as a promising paradigm to address the performance and energy limitations of traditional Von-Neumann architecture, where the physical separation between memory and processing units creates a significant data transfer bottleneck. This bottleneck results in increased latency, higher power consumption, and reduced system efficiency, especially in data-intensive applications such as machine learning, artificial intelligence, and big data analytics. This paper presents an energy-efficient Static Random Access Memory (SRAM)-based IMC architecture specifically designed to perform reliable Boolean logic operations directly within the memory array. By integrating computation and storage in the same physical location, the proposed architecture significantly reduces the need for frequent data movement between the processor and memory, thereby minimizing latency and improving overall system throughput. The study explores the implementation of fundamental logic operations such as AND, OR, NAND, and NOR using SRAM cells and peripheral circuits like sense amplifiers. Special emphasis is placed on addressing critical design challenges associated with SRAM-based IMC systems, including compute disturbance, half-select issues, and compute failure, which can affect data stability and reliability. Various optimization techniques, such as improved circuit design and efficient sensing mechanisms, are discussed to mitigate these issues. Furthermore, the proposed architecture is analyzed in terms of power consumption, speed, and scalability, demonstrating its suitability for modern Very Large-Scale Integration (VLSI) systems. The results indicate that SRAM-based IMC can achieve significant improvements in energy efficiency and performance compared to conventional architectures. This makes it a viable solution for next-generation low-power, high-performance computing systems.

Keywords: IMC, SRAM, Boolean Logic, VLSI, Low Power Design, Energy Efficiency expand this

INTRODUCTION

Arithmetic operations such as addition, subtraction, multiplication, and division form the foundation of all computing systems. These operations are extensively used in a wide range of applications, including digital signal processing, image processing, machine learning, and scientific computing. In conventional computing architectures based on the Von-Neumann model, arithmetic operations are performed in the central processing unit (CPU), while data is stored separately in memory. This separation creates a continuous need to transfer data between the processor and memory, leading to significant delays known as memory access latency. As data-intensive applications continue to grow, this limitation has become a major bottleneck, often referred to as the *Von-Neumann bottleneck*.

The increasing demand for high-speed and energy-efficient computation has made it necessary to explore alternative computing paradigms. In-Memory Computing (IMC) has emerged as a promising solution to overcome these challenges by integrating computation directly within memory arrays. Instead of transferring data back and forth between memory and processor, IMC enables arithmetic and logical operations to be executed where the data is stored. This approach significantly reduces data movement, leading to lower latency, reduced power consumption, and improved system performance.

Among various IMC techniques, SRAM-based IMC architectures are widely preferred due to their high speed, stability, and compatibility with existing CMOS technology. By utilizing SRAM cells along with peripheral circuits such as sense amplifiers and logic gates, it becomes possible to perform arithmetic operations directly within the memory structure. This not only enhances computational efficiency but also enables parallel processing of multiple data elements.

Bit-parallel IMC architecture further improves performance by allowing multiple bits to be processed simultaneously in a single clock cycle. Unlike bit-serial approaches, which handle one bit at a time and result in higher latency, bit-parallel designs execute arithmetic operations across multiple bits concurrently, thereby increasing throughput and reducing computation time. This makes bit-parallel IMC particularly suitable for applications requiring high-speed processing, such as neural networks, real-time data analytics, and embedded systems.

However, designing efficient IMC architectures also involves several challenges, including increased hardware complexity, reliability concerns, and efficient data access mechanisms. Despite these challenges, IMC continues to gain attention as a key technology for next-generation computing systems. This paper focuses on the design and analysis of a bit-parallel IMC architecture for arithmetic operations, aiming to achieve high performance, low latency, and improved energy efficiency in modern VLSI systems.

OBJECTIVES

The primary objective of this research is to design and analyze an energy-efficient SRAM-based In-Memory Computing (IMC) architecture capable of performing reliable Boolean logic operations while overcoming the limitations of conventional computing systems. With the rapid growth of data-intensive applications, there is an increasing need for architectures that provide high performance, low power consumption, and improved reliability. This section elaborates the key objectives of the proposed work in detail.

A. To Design an Energy-Efficient SRAM-Based IMC Architecture

One of the fundamental objectives of this study is to develop an SRAM-based IMC architecture that minimizes energy consumption while maintaining high computational efficiency. Traditional architectures consume a significant amount of power due to continuous data transfer between the processor and memory. By integrating computation directly within the SRAM array, the proposed IMC architecture reduces unnecessary data movement, which is one of the major contributors to energy loss.

The design focuses on:

- Utilizing optimized SRAM bit-cell structures (such as 6T, 8T, or advanced variants)
- Efficient use of peripheral circuits like sense amplifiers and decoders
- Reducing dynamic and static power consumption
- Supporting scalable and compact VLSI implementation

This objective ensures that the architecture is suitable for modern applications such as embedded systems, IoT devices, and AI edge processors, where power efficiency is critical.

B. To Implement In-Memory Boolean Logic Operations (AND, OR, NAND, NOR)

Another important objective is to enable the execution of fundamental Boolean logic operations directly within the memory array. Boolean operations form the basis of all digital computations, and implementing them in-memory eliminates the need for data transfer to external processing units.

The proposed work aims to:

- Design circuits capable of performing AND, OR, NAND, and NOR operations within SRAM arrays
- Utilize bit-line and word-line activation techniques for parallel computation
- Employ sense amplifiers to detect and process logic outputs efficiently
- Ensure accurate and reliable execution of logic operations

By performing these operations in-memory, the system achieves faster execution and improved throughput, which is essential for high-performance computing applications.

C. To Reduce Power Consumption and Latency

Reducing power consumption and latency is a critical objective in the design of modern computing systems. In conventional architectures, a large portion of energy is wasted during data transfer between CPU and memory. IMC architecture addresses this issue by processing data within memory itself.

This objective focuses on:

- Minimizing data movement between memory and processor
- Reducing memory access time (latency)
- Optimizing switching activity in circuits to lower dynamic power
- Implementing low-power design techniques such as voltage scaling and efficient transistor usage

Achieving lower latency ensures faster computation, while reduced power consumption increases battery life in portable and embedded devices.

D. To Analyze Reliability Issues such as Compute Disturbance

Reliability is a major concern in SRAM-based IMC architectures. While performing computations inside memory, certain issues may arise that can affect data integrity and system performance. One such issue is compute disturbance, where stored data may unintentionally change during computation.

This objective aims to:

- Identify major reliability challenges such as compute disturbance, half-select problem, and compute failure
- Analyze the causes of these issues in SRAM-based IMC systems
- Study the impact of process variations and noise on system performance
- Propose techniques to improve stability and reliability, such as optimized circuit design and controlled activation of word lines

Ensuring reliability is essential for the practical implementation of IMC systems in real-world applications.

E. To Improve Overall System Performance and Scalability

In addition to energy efficiency and reliability, the proposed architecture aims to enhance overall system performance and scalability. Modern applications require systems that can handle large volumes of data efficiently.

This objective includes:

- Supporting parallel processing capabilities within memory arrays
- Improving throughput for data-intensive applications
- Ensuring scalability for larger memory sizes and advanced VLSI designs
- Enabling integration with emerging technologies such as AI and machine learning systems

F. To Provide a Foundation for Future Research in IMC Systems

The final objective is to contribute to ongoing research in the field of In-Memory Computing by providing a strong foundation for future developments. This includes exploring new memory technologies, improving circuit designs, and enhancing performance metrics.

METHODOLOGY

This section presents the detailed methodology adopted to design, implement, and evaluate the proposed SRAM-based In-Memory Computing (IMC) architecture for Boolean logic operations. The methodology is structured into multiple stages, including architectural design, circuit implementation, simulation, and performance evaluation. Each stage is carefully planned to ensure accurate analysis of power, speed, and reliability in comparison with conventional CPU-based systems.

A. Design of SRAM-Based IMC Architecture

The first step in the methodology involves designing an SRAM-based IMC architecture capable of performing logic operations directly within the memory array. Unlike traditional systems where memory and computation are separate, this architecture integrates both functionalities into a unified structure.

The design process includes:

- Selection of appropriate SRAM bit-cell structure (such as 6T or 8T SRAM) based on stability and power requirements
- Organization of SRAM cells into arrays (rows and columns) for parallel processing
- Integration of peripheral circuits such as word-line drivers, bit-lines, and decoders
- Ensuring compatibility with standard CMOS technology for practical VLSI implementation

The SRAM array acts as both storage and computational unit, enabling logic operations without external data transfer. This significantly reduces latency and improves efficiency.

B. Implementation of In-Memory Logic Operations

The next step involves implementing Boolean logic operations directly within the SRAM array. The proposed architecture supports fundamental operations such as AND, OR, NAND, and NOR, which are essential for digital computation.

Key implementation steps include:

- Activating multiple word lines simultaneously to perform logic operations on stored data
- Utilizing bit-lines (BL and BLB) to carry logic values
- Performing charge sharing and voltage comparison techniques to derive logic outputs
- Mapping logic operations using combinations of SRAM cell states

This approach enables parallel execution of operations across multiple memory cells, improving throughput and reducing computation time.

C. Sense Amplifier-Based Logic Computation

Sense amplifiers play a crucial role in detecting and interpreting the results of in-memory computations. In this methodology, sense amplifiers are used not only for reading stored data but also for performing logic operations.

The process includes:

- Connecting sense amplifiers to bit-lines to detect voltage differences
- Comparing bit-line voltages with reference voltage (VREF)
- Generating logic outputs based on sensed values
- Using differential sensing techniques to improve accuracy and speed

Sense amplifiers help in converting small analog voltage differences into clear digital outputs, ensuring reliable logic computation within the memory array.

D. Simulation and Performance Analysis

To evaluate the effectiveness of the proposed IMC architecture, simulation tools are used for performance analysis. Simulation provides insights into power consumption, delay, and overall system behavior under different conditions.

The simulation process involves:

- Designing the circuit using tools such as Cadence, HSPICE, or MATLAB
- Modeling SRAM cells and peripheral circuits
- Running simulations for different input combinations and logic operations
- Measuring key performance metrics such as:
    - Power consumption (dynamic and static)
    - Propagation delay
    - Energy efficiency
    - Throughput

Simulation results help in validating the proposed design and identifying areas for improvement.

E. Comparison with Traditional CPU-Based Systems

A comparative analysis is performed to highlight the advantages of the proposed IMC architecture over conventional CPU-based systems. In traditional systems, logic operations require data transfer between memory and processor, leading to increased latency and energy consumption.

The comparison focuses on:

- Execution time for logic operations
- Power consumption during computation
- Data transfer overhead
- Overall system efficiency

The IMC architecture is expected to outperform traditional systems by reducing data movement and enabling parallel computation.

F. Analysis of Reliability and Design Challenges

In addition to performance evaluation, the methodology also includes analyzing reliability issues associated with SRAM-based IMC systems. These challenges can impact system stability and correctness of operations.

Key aspects analyzed include:

- **Compute Disturbance:** Unintended data flipping during computation
- **Half-Select Issue:** Disturbance in partially selected memory cells
- **Process Variations:** Impact of manufacturing variations on performance
- **Noise Sensitivity:** Effect of electrical noise on logic accuracy

Various techniques such as controlled word-line activation and optimized sensing mechanisms are considered to minimize these issues.

G. Optimization Techniques for Improved Performance

To enhance the efficiency of the proposed architecture, several optimization techniques are applied during the design process.

These include:

- Reducing switching activity to minimize dynamic power consumption
- Optimizing transistor sizing for better performance
- Using low-power design techniques such as voltage scaling
- Improving layout design for compact VLSI implementation

These optimizations ensure that the architecture meets the requirements of modern low-power and high-performance systems.

RESULTS & DISCUSSION

This section presents the results obtained from the simulation and analysis of the proposed SRAM-based In-Memory Computing (IMC) architecture. The performance of the system is evaluated based on key metrics such as power consumption, speed (latency), throughput, and reliability. A comparative study with traditional Von-Neumann architecture is also discussed to highlight the advantages and limitations of the proposed approach.

A. Reduction in Power Consumption

One of the most significant outcomes of the proposed IMC architecture is the substantial reduction in power consumption. In conventional Von-Neumann systems, a large amount of energy is consumed due to continuous data transfer between the processor and memory. This data movement contributes to both dynamic power (due to switching activity) and static power (due to leakage currents).

In contrast, the SRAM-based IMC architecture performs computations directly within the memory array, thereby minimizing unnecessary data transfers. As a result:

- Dynamic power consumption is reduced due to fewer switching operations on data buses
- Static power consumption is minimized by efficient utilization of SRAM cells
- Overall energy efficiency is improved significantly

Simulation results indicate that the proposed IMC system consumes considerably less power compared to traditional CPU-based computation, making it highly suitable for low-power applications such as IoT devices, embedded systems, and AI edge processors.

B. Improvement in Speed and Performance

Another major advantage observed is the improvement in computational speed. In traditional architectures, the delay caused by memory access and data transfer limits system performance. However, in the IMC architecture, computations are performed within memory, eliminating this delay.

Key observations include:

- Reduced latency due to elimination of memory access delays
- Faster execution of logic operations (AND, OR, NAND, NOR)
- Improved system throughput

Additionally, the use of **parallel processing** in SRAM arrays allows multiple operations to be executed simultaneously. This bit-parallel computation significantly enhances performance compared to bit-serial approaches, where operations are performed sequentially. As a result, the proposed architecture achieves higher speed and efficiency, especially in data-intensive applications.

C. Enhanced Throughput Through Parallelism

The ability to perform multiple operations in parallel is a key strength of the IMC architecture. SRAM arrays are organized in rows and columns, enabling simultaneous activation of multiple word lines and bit lines.

This leads to:

- Increased throughput for large datasets
- Efficient handling of bulk data operations
- Reduced computation time for complex tasks

Parallelism is particularly beneficial for applications such as machine learning, image processing, and real-time analytics, where large volumes of data need to be processed quickly.

D. Identification of Reliability Issues

While the proposed IMC architecture offers significant advantages, the analysis also reveals certain reliability challenges that must be addressed for practical implementation.

1. Half-Select Issue

The half-select problem occurs when certain memory cells are partially activated during computation. This can lead to unintended disturbances in stored data, especially in cells that are not directly involved in the operation.

- It affects data integrity and stability
- It is more prominent in dense SRAM arrays
- Requires careful control of word-line and bit-line activation

2. Compute Disturbance

Compute disturbance refers to the unintended alteration of stored data during in-memory computation. This happens due to voltage fluctuations and interactions between neighboring cells.

- It can cause data flipping or corruption
- It affects the reliability of logic operations
- It is influenced by process variations and noise

3. Compute Failure
In some cases, the system may fail to produce accurate logic outputs due to insufficient voltage difference or sensing errors.

- Occurs due to limitations in sense amplifier design
- Affects accuracy of Boolean operations

E. Impact of Process Variations and Noise
The performance of SRAM-based IMC systems is also affected by manufacturing variations and electrical noise. These factors can influence:

- Threshold voltage of transistors
- Stability of SRAM cells
- Accuracy of sense amplifier outputs

Simulation results show that variations in process parameters can lead to reduced reliability, especially in advanced technology nodes. Therefore, robust design techniques are required to ensure stable operation under different conditions.

F. Trade-Off Analysis
The results highlight several trade-offs in the proposed IMC architecture:

- **Power vs Performance:** Lower power consumption is achieved without compromising performance
- **Speed vs Complexity:** Higher speed is obtained at the cost of increased circuit complexity
- **Parallelism vs Reliability:** Parallel operations improve throughput but may introduce reliability issues

Understanding these trade-offs is essential for optimizing the architecture based on specific application requirements.

G. Overall Performance Evaluation
Based on the simulation and analysis, the proposed SRAM-based IMC architecture demonstrates:

- Significant reduction in power consumption
- Improved computational speed and reduced latency
- High throughput due to parallel processing
- Identifiable reliability challenges that require optimization

CONCLUSION

This paper presented the design and analysis of an energy-efficient SRAM-based In-Memory Computing (IMC) architecture for performing Boolean logic operations directly within memory. The study addressed one of the major limitations of traditional computing systems, namely the Von-Neumann bottleneck, which arises due to the separation of processing and memory units. By integrating computation into the memory array, the proposed IMC architecture significantly reduces data movement, resulting in lower latency, reduced power consumption, and improved overall system performance.

The results obtained from simulation and analysis demonstrate that SRAM-based IMC offers substantial advantages over conventional CPU-based systems. The ability to perform parallel operations within memory enables faster execution of logic functions, thereby increasing throughput and efficiency. Additionally, the reduction in data transfer between memory and processor leads to considerable energy savings, making the architecture highly suitable for modern applications such as Internet of Things (IoT) devices, embedded systems, and artificial intelligence edge processors, where power efficiency is a critical requirement.

Despite these advantages, the study also highlights several reliability challenges associated with SRAM-based IMC architectures. Issues such as compute disturbance, half-select problems, and compute failure can affect data stability and accuracy of operations. These challenges are further influenced by factors such as process variations, noise, and limitations in sensing mechanisms. Therefore, addressing these reliability concerns is essential for ensuring the practical implementation of IMC systems in real-world applications.

To overcome these limitations, future work should focus on developing improved SRAM cell designs, advanced sense amplifier techniques, and robust control mechanisms for word-line and bit-line operations. Additionally, exploring hybrid memory technologies and incorporating error correction techniques can further enhance system reliability and performance. Optimization strategies such as voltage scaling, power gating, and adaptive circuit design can also be employed to achieve better energy efficiency without compromising speed.

In conclusion, SRAM-based IMC represents a promising direction for next-generation computing architectures, offering a balanced combination of high performance, low power consumption, and scalability. With continued research and technological advancements, IMC has the potential to play a crucial role in enabling efficient and intelligent computing systems for future applications.

REFERENCES

[1] S. Mittal, "A Survey of ReRAM-Based Architectures for Processing-in-Memory," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 13, no. 2, pp. 1–36, 2017.

[2] A. Shafiee et al., "ISAAC: A Convolutional Neural Network Accelerator with In-Situ Analog Arithmetic in Crossbars," *Proceedings of the 43rd Annual International Symposium on Computer Architecture (ISCA)*, pp. 14–26, 2016.

[3] P. Chi et al., "PRIME: A Novel Processing-in-Memory Architecture for Neural Network Computation in ReRAM-Based Main Memory," *Proceedings of the 43rd ISCA*, pp. 27–39, 2016.

[4] M. Kang et al., "An Energy-Efficient VLSI Architecture for In-Memory Computing Using SRAM," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 1, pp. 1–10, 2017.

[5] J. Zhang et al., "In-Memory Computing on a 16-nm CMOS Platform for Energy-Efficient Neural Network Processing," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 1, pp. 1–12, 2017.

[6] S. Li et al., "DRISA: A DRAM-Based Reconfigurable In-Situ Accelerator," *Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture*, pp. 288–301, 2017.

[7] H. Li et al., "SRAM-Based In-Memory Computing for Low-Power Applications," *IEEE Transactions on Circuits and Systems*, vol. 65, no. 12, pp. 1–10, 2018.

[8] Y. Chen et al., "DaDianNao: A Machine-Learning Supercomputer," *Proceedings of the 47th Annual IEEE/ACM International Symposium on Microarchitecture*, pp. 609–622, 2014.

[9] K. Pagiamtzis and A. Sheikholeslami, "Content-Addressable Memory (CAM) Circuits and Architectures: A Tutorial and Survey," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 3, pp. 712–727, 2006.

[10] N. Verma and A. Chandrakasan, "A 256 kb 65 nm 8T Subthreshold SRAM Employing Sense-Amplifier Redundancy," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, 2008.