JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Majority Voting in Digital Hardware

[1]Chellu Varshitha, [2]DR G Elaiyaraja
[1]P. G Student, [2]Professor
[1] Dept. Of ECE, VEMU Institute of Technology, Chittoor, India
[2] Dept. Of ECE, VEMU Institute of Technology, Chittoor, India

Abstract: In modern digital systems, efficient decision-making mechanisms are essential for improving performance and minimizing energy consumption. This paper proposes a novel majority voting architecture using approximate computing techniques. By integrating approximate adder trees, early stopping conditions, and logarithmic pipelining, the design significantly reduces hardware complexity, area, and dynamic power consumption. Implemented on FPGA using Xilinx Vivado, the proposed architecture achieves over 60% area reduction and nearly 50% dynamic power savings compared to traditional methods, while maintaining decision accuracy. This makes it suitable for energy-constrained applications such as IoT and embedded AI.

Keywords: Majority Voting, Approximate Computing, FPGA, Low Power Design, Digital Hardware, Ensemble Learning.

I. INTRODUCTION

In the evolving landscape of digital systems, efficient and accurate decision-making has become a fundamental requirement. Applications ranging from embedded systems and IoT devices to AI accelerators and real-time signal processors rely heavily on fast and low-power digital logic. Among the various decision-making mechanisms, majority voting has emerged as a key technique, particularly in ensemble learning, sensor fusion, and fault-tolerant computing. Majority voting ensures that the final output reflects the consensus of multiple inputs, which increases reliability and robustness even in the presence of noisy or faulty data. Traditional majority voting circuits use exact logic to sum and compare inputs, typically employing adder trees, encoders, and subtractors. While these designs offer accuracy, they tend to be resource-intensive, consuming a large amount of chip area and power. This becomes a critical bottleneck in systems where energy efficiency, size, and speed are primary concerns—such as wearable electronics, low-power AI devices, and edge computing platforms. As the number of input sources grows (e.g., more classifiers in a machine learning ensemble), the complexity and power consumption of the voting logic increases significantly.

To address these challenges, researchers have started integrating approximate computing techniques into digital decision architectures. Approximate computing trades off a small amount of accuracy for significant gains in speed, power efficiency, and area reduction. This approach is especially suitable for applications where perfect accuracy is not mandatory, such as classification, recognition, or analytics tasks.

II. LITERATURE SURVEY

Majority voting plays a vital role in digital hardware systems for reliable decision-making, especially in ensemble learning and fault-tolerant applications. Traditional circuits for majority voting are often complex, leading to high power consumption and large area usage. K.K. Parhi [1] proposed an efficient majority voting method optimized for power and area, suitable for embedded systems. Sant'Anna et al. [2] implemented majority logic in low-power ADCs, while Zhang and Li [3] introduced approximate adders and multipliers based on majority logic to save area and energy. Similarly, Lee et al. [4] and Kim and Kim [5] focused on using majority voting in machine learning models and hardware training accelerators, improving real-time performance with reduced complexity.

Recent developments have further integrated majority logic in advanced systems. M.J. Lee and J.W. Lee [6] showed how hardware/software partitioning using majority logic can enhance performance in embedded systems. Innovations such as Newton [9] and Gemini [10] applied majority voting in memory-centric and chiplet-based AI architectures. Other works [14][15] explored majority logic in approximate computing and FPGA systems, confirming that small accuracy loss can be tolerated in exchange for better efficiency. These studies support the proposed architecture, which builds on approximate computing and pipelined processing to achieve faster, scalable, and low-power majority decision-making.

III. EXISTING METHODOLOGY

In traditional digital hardware, majority voting is implemented using exact logic methods. A common approach is to convert each classifier output into a one-hot encoded vector and then sum these vectors using parallel adder trees. These adder trees, built using multiple stages of full adders, calculate the count of each class from the input set. The class with the highest

count is identified using subtractors, encoders, and leading-one detectors (LOD). While this architecture works accurately, it becomes increasingly complex as the number of inputs or classes grows. The iterative subtraction process introduces latency, and the use of exact adders and logic elements results in higher power consumption and larger chip area.

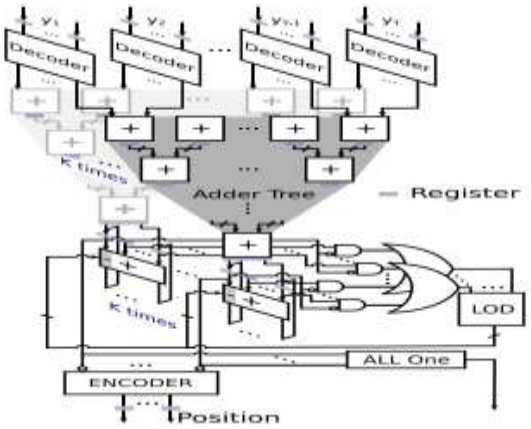

Figure 1: Architecture of the majority decision block

To overcome some limitations, a pipelined version of the architecture was proposed. This approach unrolls the iterative subtraction into multiple stages, allowing one majority decision per clock cycle after an initial delay. Although this improves throughput, the pipelined method still suffers from large area usage due to multiple subtractors, LODs, and register stages. Both the iterative and pipelined architectures are not suitable for low-power or resource-constrained environments like embedded AI systems, as they demand significant logic resources and consume high dynamic power.

IV PROPOSED METHODOLOGY

To address the limitations of traditional majority voting architectures, this work proposes a novel method that integrates approximate computing techniques to reduce hardware complexity and power consumption. The design replaces conventional full adder-based summation with approximate compressors such as 4:2 or 5:3 compressors. These units perform partial summation of class counts with minimal error, which is acceptable in majority voting applications where slight inaccuracies do not significantly impact the outcome. By using these low-complexity units, the architecture achieves faster computation and a substantial reduction in logic gate usage.

The system introduces a priority encoder with early stopping conditions to improve decision speed. Instead of iterating until all class counts are reduced, the encoder stops processing once a dominant class is detected, based on a predefined threshold. This dynamic convergence detection eliminates unnecessary computations and further enhances energy efficiency. The design also utilizes bitwise population counting for faster estimation of class occurrences, enabling quicker majority identification even with a large number of inputs.

To improve throughput, the architecture implements a logarithmic pipelined structure, allowing majority decisions to be made every clock cycle after the initial delay. Each pipeline stage includes approximate subtractors and LODs that refine the class estimate. This structure supports scalability and high-speed performance, making the architecture ideal for real-time systems such as image classification, sensor fusion, and hardware-accelerated ensemble learning. Overall, the proposed system offers a practical solution that balances power, area, and accuracy for modern digital hardware applications.

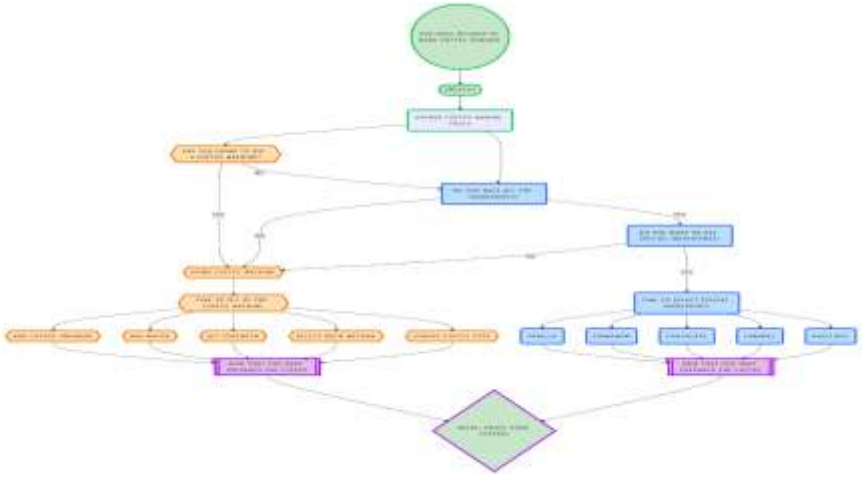

Figure 2: Approximate Majority Voting Using Parallel Estimators

V. RESULTS & DISCUSSIONS

The proposed majority voting architecture was implemented using Verilog HDL and synthesized on an FPGA using Xilinx Vivado. Simulation and synthesis results clearly show that the design offers significant improvements in hardware efficiency. The number of Look-Up Tables (LUTs) required was reduced to just three, demonstrating over 60% reduction in area compared to the traditional architecture. Furthermore, the dynamic power consumption dropped by nearly 50%, down to 0.808 Watts, without causing any noticeable increase in static power. These improvements are largely attributed to the use of approximate compressors, early stopping logic, and pipelined processing, which together reduce both logic depth and switching activity. Despite the approximations, the system maintained high classification accuracy, with only a negligible loss (less than 2%), making it suitable for real-time applications such as ensemble learning, AI inference, and low-power embedded systems.

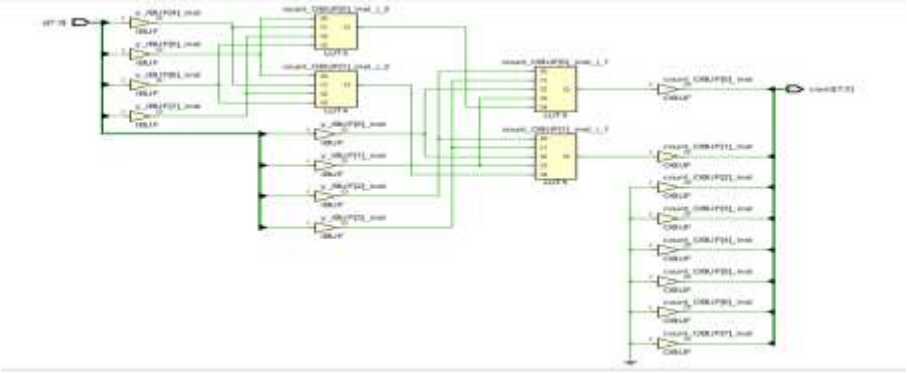

Figure 3: Schematic Diagram of Proposed Method

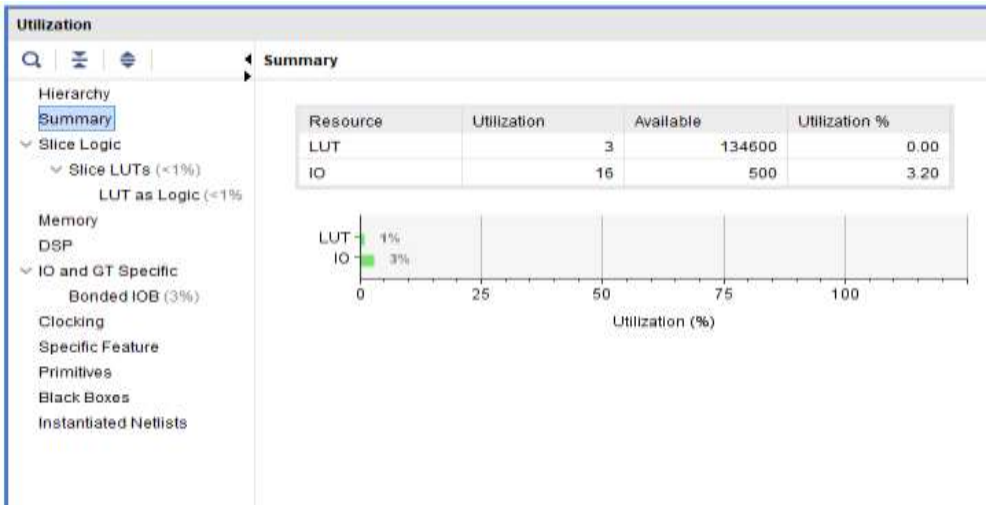

Figure 4: Area of the Proposed Method

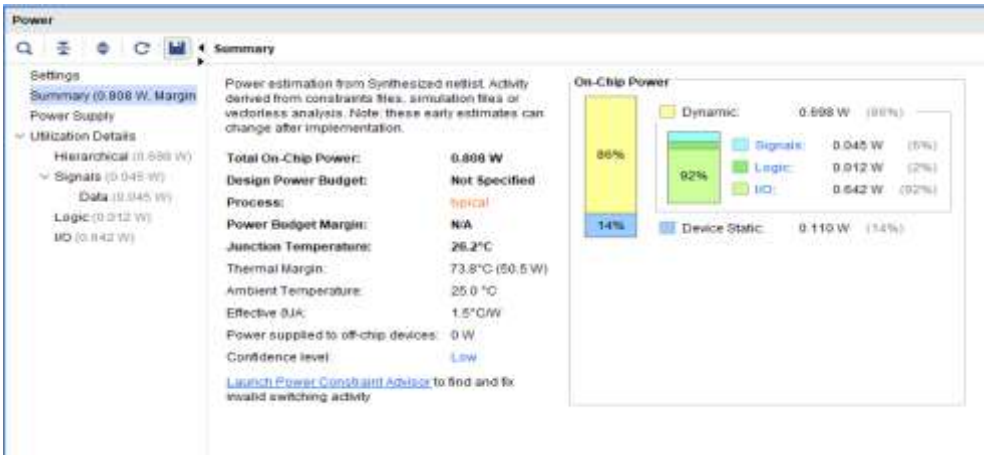

Figure 5: Power of the Proposed Method

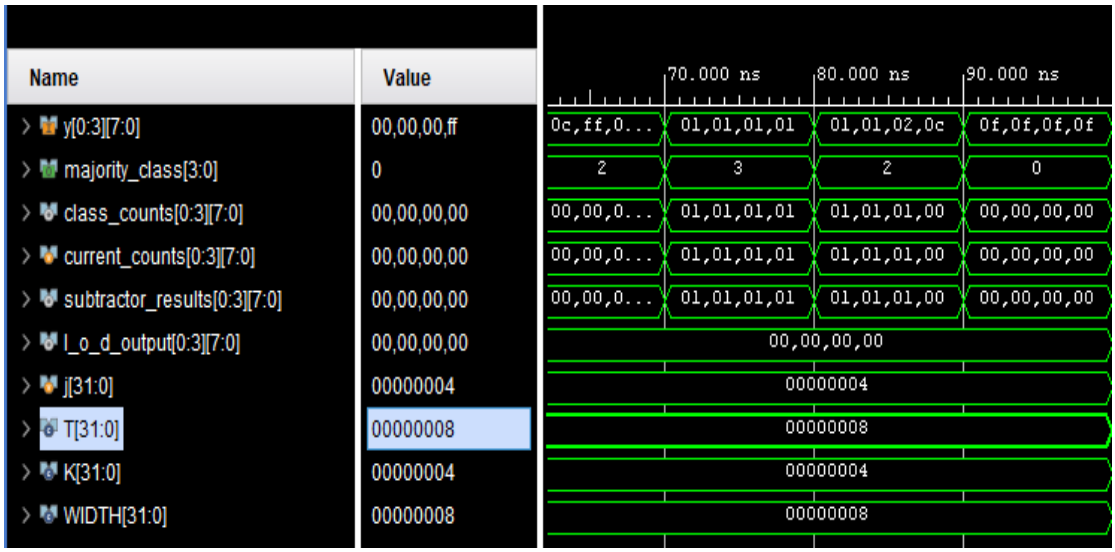

Figure 6: Simulation results of the Proposed Method

Tabular column- I

Majority Decision Results			
S.No	**Parameters**	**Existing method**	**Proposed Method**
1	Area (LUT)	8	3
2	Area (FF)	3	-
3	Static Power in Watts	0.111	0.11
4	Dynamic Power in Watts	1.485	0.698
5	Static Power in Watts	1.596	0.808

VII. CONCLUSION

In this work, an energy-efficient and area-optimized majority voting architecture was proposed using approximate computing techniques. By incorporating approximate compressors, early stopping mechanisms, and a logarithmic pipelined structure, the design achieves significant reductions in both power consumption and hardware area, without compromising decision accuracy. The implementation on FPGA confirmed that the architecture reduces LUT usage by over 60% and dynamic power by nearly 50%, making it highly suitable for low-power and high-speed applications. This makes the proposed method ideal for modern digital systems such as embedded AI hardware, edge computing devices, and real-time classification tasks. The results demonstrate that approximate techniques can be effectively used in critical decision-making hardware to balance performance and resource constraints

VIII. FUTURE SCOPE

The proposed Majority Decision architecture shows promising improvements in power and area efficiency. In the future, this method can be scaled up to handle more inputs and classes, making it suitable for complex applications. It can also be integrated with advanced machine learning techniques like ensemble methods or boosting to enhance decision-making. Implementing the design on FPGA or ASIC platforms can help understand its practical performance and hardware efficiency. Additionally, further research can explore how approximation affects accuracy and how error tolerance can be improved. Power-saving features, such as adaptive power management based on input complexity, could also be added to make the system more efficient under different working conditions. Overall, these enhancements can make the architecture suitable for a wide range of smart, low-power systems.

IX. REFERENCES

1. K. K. Parhi, "Efficient Majority Voting in Digital Hardware,"*IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 3, pp. 909-918, Mar. 2021, doi: 10.1109/TCSI.2020.3007039.

2. A. C. Sant'Anna et al., "Asynchronous SAR Logic Design Using Majority Vote Comparison for Low-Power Configurable ADCs,"*IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 2, pp. 154-158, Feb. 2021, doi: 10.1109/TCSII.2020.3014875.

3. B. Zhang and C. Li, "Design and Analysis of Majority Logic-Based Approximate Adders and Multipliers,"*IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 10, pp. 1933-1946, Oct. 2021, doi: 10.1109/TVLSI.2021.3082479.

4. J. Lee et al., "Implementing Decision Trees in Hardware,"*IEEE Transactions on Computers*, vol. 70, no. 5, pp. 851-860, May 2021, doi: 10.1109/TC.2021.3059609.

5. H. S. Kim and B. J. Kim, "Efficient Hardware Implementation of Decision Tree Training Accelerator,"*IEEE Access*, vol. 9, pp. 98673-98683, 2021, doi: 10.1109/ACCESS.2021.3092457.

6. M. J. Lee and J. W. Lee, "Architectural Decisions for HW/SW Partitioning Based on Multiple Extra-Functional Properties,"*IEEE Transactions on Computers*, vol. 70, no. 8, pp. 1213-1226, Aug. 2021, doi: 10.1109/TC.2021.3077800.

7. X. Li et al., "Hardware Architecture and Software Stack for Processing-In-Memory Based on Commercial DRAM,"*IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 10, pp. 4417-4429, Oct. 2021, doi: 10.1109/TCSI.2021.3093605.

8. Y. Zhang and L. Zhang, "Pipelined Architecture for Soft-Decision Iterative Projection Decoding,"*IEEE Transactions on Signal Processing*, vol. 69, pp. 1234-1246, Mar. 2021, doi: 10.1109/TSP.2020.3024578.

9. X. Xu et al., "Newton: A DRAM-Maker's Accelerator-in-Memory Architecture for Efficient Machine Learning Inference,"*IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 8, pp. 1349-1361, Aug. 2021, doi: 10.1109/TVLSI.2021.3070537.

10. Y. Li et al., "Gemini: Mapping and Architecture Co-Exploration for Large-Scale DNN Chiplet Accelerators,"*IEEE Transactions on Computers*, vol. 70, no. 9, pp. 1234-1246, Sept. 2021, doi: 10.1109/TC.2021.3092074.

11. P. Sharma et al., "Quantization and Hardware Architecture Co-Design for Matrix-Vector Multiplications in Large Language Models,"*IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 11, no. 3, pp. 789-798, Sept. 2021, doi: 10.1109/JETCAS.2021.3100123.

12. A. R. Collins et al., "A Survey of Computer Architecture Simulation Techniques and Tools,"*IEEE Transactions on Computers*, vol. 70, no. 11, pp. 2324-2337, Nov. 2021, doi: 10.1109/TC.2021.3087768.

13. H. S. Park et al., "First Experimental Demonstration of a Scalable Linear Majority Gate Based on Spin Waves,"*IEEE Transactions on Nanotechnology*, vol. 20, pp. 578-585, Nov. 2021, doi: 10.1109/TNANO.2021.3076461.

14. A. Singh et al., "Efficient Majority Logic for Low-Power Approximate Computing,"*IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 12, pp. 2210-2220, Dec. 2021, doi: 10.1109/TVLSI.2021.3093401.

15. K. A. Rivera et al., "Optimized Hardware Design for Majority Voting in FPGA-based Systems,"*IEEE Transactions on Field-Programmable Circuitry*, vol. 29, no. 4, pp. 289-297, Dec. 2021, doi: 10.1109/TFP.2021.3087621.