



DESIGN AND IMPLEMENTATION OF LOW POWER APPROXIMATE COMPRESSOR FOR BALANCED ERROR ACCUMULATION IN MAC OPERATOR

¹Ajith Kumar G, ²Dr. Vijaya Prakash A M,

¹M. Tech Student, ²Principal & Head (EEVDT), Dean-Infrastructure,

¹Department of Electronics and Communication Engineering,

¹Bangalore Institute of Technology, Bengaluru, India

Abstract: This manuscript discusses the design of Low Power Approximate Compressor for Balanced Error Accumulation in MAC operators used in VLSI and Digital Signal Processing applications. Approximate 4:2 compressors are used in the partial product reduction stage of multiplier design to reduce hardware complexity, switching activity, and propagation delay while maintaining acceptable computational accuracy. The proposed 8-bit MAC architecture is designed using System Verilog and verified using Xilinx Vivado Design Suite. The performance analysis of the proposed architecture is evaluated in terms of timing performance, area utilization, LUTs, and slices. Experimental results show that the proposed approximate compressor-based MAC architecture reduces hardware complexity and improves speed performance compared to conventional exact MAC designs. Therefore, the proposed design is suitable for low power VLSI system applications.

Keywords - Approximate Computing, MAC Operator, Approximate Compressor, Low Power Design, VLSI, Digital Signal Processing.

I. INTRODUCTION

Nowadays, low power and high-speed digital systems are becoming important in VLSI applications. Multiply and Accumulate (MAC) units are widely used in digital signal processing, image processing, multimedia systems, and machine learning applications. Since MAC operations are repeatedly performed in these systems, they consume more power and increase hardware complexity. Therefore, designing an efficient MAC architecture with reduced power consumption and improved speed has become an important research area. Approximate computing is one of the effective techniques used to reduce power consumption, propagation delay, and circuit complexity in error-tolerant applications. Instead of producing completely accurate outputs, approximate circuits generate near accurate results with lower hardware requirements. In many multimedia and signal processing applications, small computational errors can be accepted without significantly affecting the final output quality. In multiplier architectures, the partial product reduction stage contributes more to area and delay. Approximate 4:2 compressors are widely used in this stage to simplify logic operations and improve overall multiplier performance. However, repeated accumulation operations may increase error propagation in MAC architectures. To overcome this problem, balanced error accumulation techniques are used to minimize error generation during accumulation

operations. In this work, an approximate compressor-based 8-bit MAC architecture is proposed for low power VLSI applications. The proposed design is implemented using SystemVerilog and verified using Xilinx Vivado Design Suite. The performance of the proposed architecture is analyzed based on timing performance, area utilization, LUTs, and slices. The obtained results show improvement in hardware efficiency and speed compared to conventional exact MAC architectures.

II. LITERATURE REVIEW

M.N. Bhavana et al. "A Low-Power High-Speed Approximate Multiplier for MAC Applications" proposed an approximate multiplier design to improve the performance of MAC units used in digital signal processing systems. The work focused on reducing hardware complexity and improving arithmetic speed by utilizing approximate multiplication techniques. The authors explained that MAC units are one of the most important arithmetic blocks in DSP systems because they continuously perform multiplication and accumulation operations. In the proposed architecture, approximate arithmetic techniques were applied to reduce area utilization, propagation delay, and power consumption. The design achieved better arithmetic efficiency and reduced hardware overhead compared to exact multiplier architectures. Simulation results obtained using Cadence EDA tools showed improvements in area, power, and delay performance for low-power VLSI applications[1].

L. Thulasimani et al. "Performance Analysis of Interleaved MAC Design for Various Adder Architecture" proposed an interleaved approximate MAC architecture to reduce power, area, and delay in signal processing applications. The work focused on balanced error accumulation using positive and negative approximate 4-2 compressors. Different approximate multipliers were interleaved to compensate errors generated during accumulation operations. The proposed architecture reduced arithmetic complexity and improved hardware efficiency compared to existing cost-aware and accuracy-aware architectures. Experimental results showed improvements in power consumption, propagation delay, and area utilization while maintaining acceptable computational accuracy for DSP applications[2].

D. Anantha Lakshmi et al. "Static Segmentation based Approximate MAC for High Performance" proposed a modified approximate MAC architecture using static segmentation techniques to improve hardware efficiency. The proposed design utilized segmented multiplication and a carry propagate adder to simplify arithmetic operations and reduce circuit complexity. Multiplexers were used for partial product selection and data flow control. An error compensation mechanism was also introduced to maintain acceptable computational accuracy while reducing power consumption and area utilization. Simulation results showed that the proposed MAC architecture achieved considerable reductions in hardware complexity and switching activity compared to traditional MAC units[3].

Thejaswini P et al. "Energy Efficient Approximate MACs" proposed several approximate MAC architectures for high-performance and energy-efficient computing systems. The proposed MAC units utilized approximate adders and compressors to reduce arithmetic complexity and minimize the number of partial product reduction stages. The work mainly focused on edge devices and image processing applications where low-power operation and low latency are important design requirements. Approximate computing techniques were applied to reduce power consumption and hardware area while maintaining acceptable computational accuracy. The proposed architectures achieved significant improvements in area utilization and energy efficiency compared to conventional exact MAC units. The experimental results also demonstrated good image quality performance using PSNR and SSIM metrics for image smoothing applications[4].

Dr. Vidyasaraswathi H N et al. "Optimized MAC Unit using Binary Carry Select Adder and Counter-Based Modular Wallace Tree Multiplier" proposed an optimized MAC architecture for embedded systems using a counter-based modular Wallace tree multiplier and binary carry select adder. The proposed design focused on improving arithmetic speed while reducing power consumption in MAC operations. Carry propagation length was minimized to reduce delay and energy dissipation during multiplication and accumulation processes. The architecture achieved better performance, reduced hardware complexity, and improved energy efficiency compared to traditional MAC structures[5].

Vaddi Rajkumar et al. “Design and Analysis of Energy Efficient Approximate Multipliers For Image Processing Using Verilog HDL Programming” proposed several approximate multiplier architectures for image processing and AI applications. The proposed designs utilized recursive, bit-wise, and hybrid approximation techniques to improve hardware efficiency. Approximate computing methods were applied to reduce area, power consumption, propagation delay, and power-delay product. The proposed architectures achieved considerable improvements in energy efficiency while maintaining acceptable image quality and computational accuracy. Experimental results also demonstrated better PSNR and SSIM performance for image smoothing and edge detection applications[6].

Wei Tung et al. “Low-Power High-Accuracy Approximate Multiplier Using Approximate High-Order Compressors” proposed a high-speed and low-power approximate multiplier architecture using high-order compressors. The proposed design utilized different compressor structures depending on the significance of partial products. Accurate compressors were used for higher significance bits while approximate compressors were used for lower significance bits to reduce hardware complexity and power consumption. The architecture also introduced high-order approximate compressors such as 8:2 compressors to minimize carry chain logic and improve arithmetic speed. The proposed multiplier achieved significant reduction in propagation delay and dynamic power dissipation while maintaining very low normalized mean error distance values. Experimental analysis demonstrated that the architecture achieved both low-power operation and high computational accuracy, making it suitable for signal processing and low-power VLSI systems[7].

III. EXISTING SYSTEM

The existing MAC architecture uses an exact compressor to perform multiplication and accumulation operations. The exact compressor produces accurate outputs by generating precise sum and carry values. Due to its high accuracy, it is widely used in digital signal processing and arithmetic circuits. However, the exact compressor requires more logic resources, which increases power consumption and delay. As modern applications demand high speed and low power operation, the conventional exact compressor-based MAC architecture becomes less efficient. Therefore, there is a need for an improved design that can reduce power consumption and delay while maintaining acceptable accuracy.

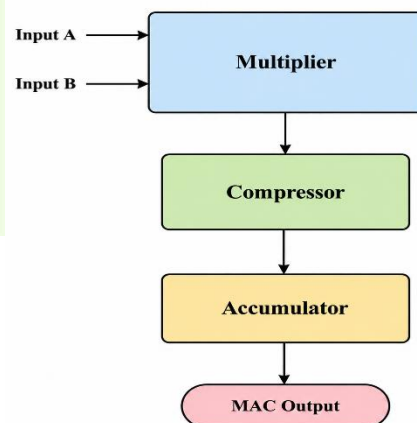


Fig. 1. Conventional MAC Architecture

IV. PROPOSED METHODOLOGY

A. Overview

The proposed methodology focuses on improving the performance of the MAC operator by utilizing approximate computing techniques. The architecture employs approximate multipliers along with an interleaved accumulation mechanism to reduce hardware complexity and improve operational efficiency. By balancing the generated errors during multiplication and accumulation, the proposed design achieves better performance while maintaining acceptable accuracy.

B. Working Principle

The proposed MAC architecture consists of positive and negative approximate multipliers, a 2:1 multiplexer, a phase toggle control unit, and an interleaved accumulator. The approximate multipliers generate multiplication results with different error characteristics. Based on the control signal, the multiplexer selects one of the multiplier outputs and forwards it to the accumulator. The accumulated result is then used to generate the final MAC output. The phase toggle control helps in balancing error accumulation during successive operations.

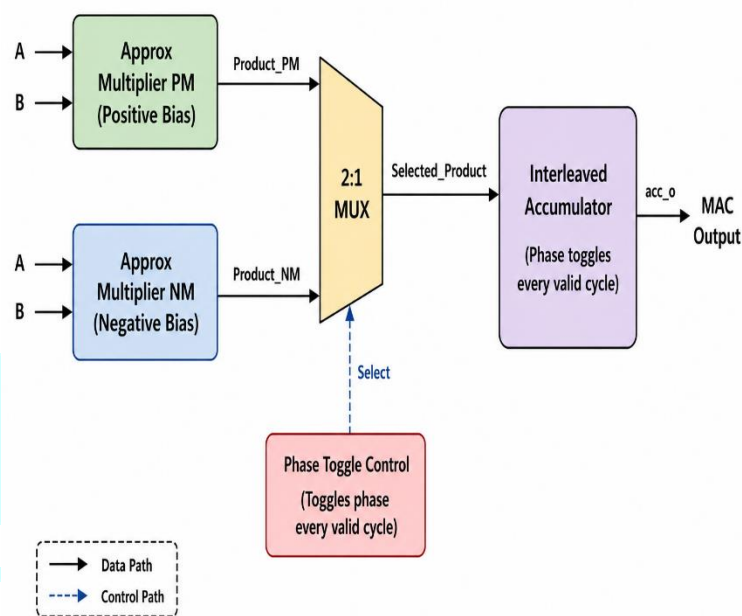


Fig. 2. Proposed MAC Architecture

Fig. 2 shows the proposed MAC architecture. The design consists of two approximate multipliers, namely the positive multiplier (PM) and negative multiplier (NM), a 2:1 multiplexer, a phase toggle control unit, and an interleaved accumulator. The approximate multipliers generate multiplication results with opposite error characteristics. The 2:1 multiplexer selects one of the multiplier outputs based on the control signal generated by the phase toggle unit. The selected output is then applied to the interleaved accumulator, where the multiplication results are accumulated to produce the final MAC output. This approach helps in balancing the accumulated error while improving the overall performance of the MAC operator.

C. Approximate Compressor Design

The proposed MAC architecture utilizes an approximate compressor in the partial product reduction stage of multiplication. The approximate compressor simplifies the logic operations involved in generating sum and carry outputs, thereby reducing hardware complexity, power consumption, and propagation delay. Unlike an exact compressor, the proposed compressor allows a small computational error in exchange for improved hardware efficiency. The generated outputs are then processed by the approximate multipliers and accumulated using the interleaved accumulation mechanism to achieve balanced error accumulation.

D. Mathematical Model

The operation of the proposed MAC architecture is based on multiplication and accumulation. In a conventional MAC unit, the output is obtained by multiplying the input operands and adding the result to the accumulated value. The mathematical expression of the MAC operation is given by:

$$\text{MAC} = (A \times B) + \text{Acc}$$

where A and B represent the input operands, and Acc represents the accumulated value from the previous operation. The multiplication result is added to the accumulated value to generate the final MAC output.

In the proposed architecture, two approximate multipliers are used to generate multiplication results with positive and negative error characteristics. The Positive Multiplier (PM) produces an output with a small positive error, whereas the Negative Multiplier (NM) produces an output with a small negative error. These outputs can be mathematically represented as follows:

$$PM = (A \times B) + e$$

$$NM = (A \times B) - e$$

where e represents the approximation error introduced during the multiplication process. The output selected by the 2:1 multiplexer is then accumulated to generate the final MAC output. By alternating between PM and NM outputs, the proposed architecture helps in balancing the accumulated error and improves the overall efficiency of the MAC operator. The balanced use of PM and NM outputs helps in reducing long-term error accumulation during successive MAC operations.

V. RESULTS

A. RTL Schematic

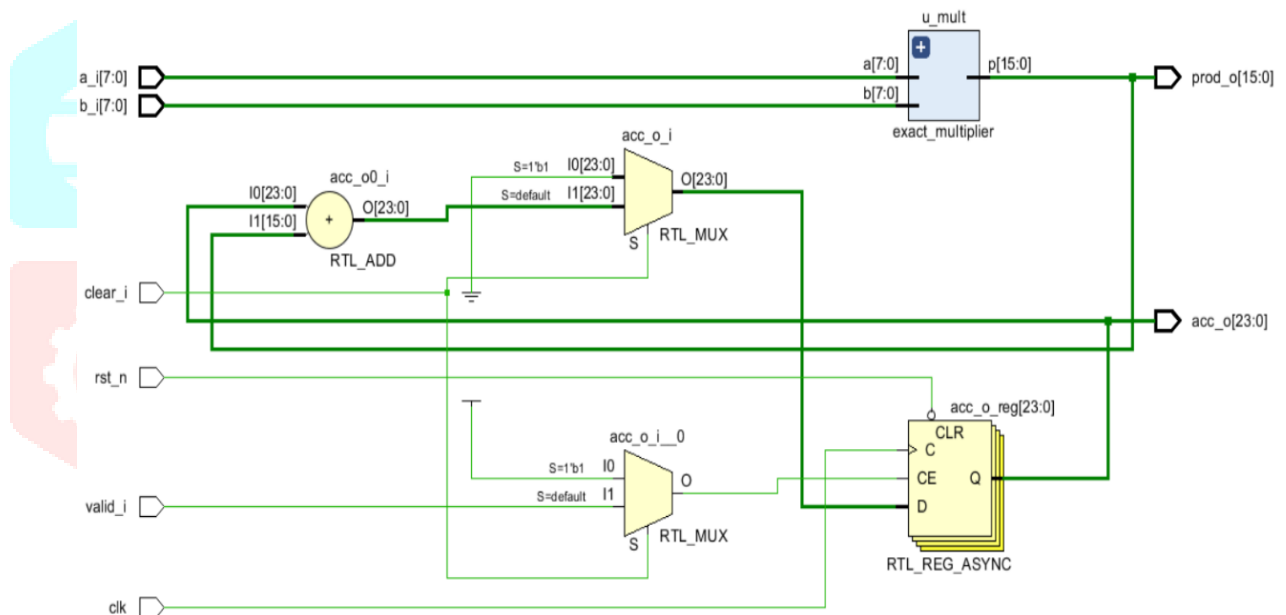


Fig. 3. RTL Schematic of Exact MAC

Fig. 3 shows the RTL schematic of the exact MAC design. The schematic illustrates the functional blocks involved in performing multiplication and accumulation operations.

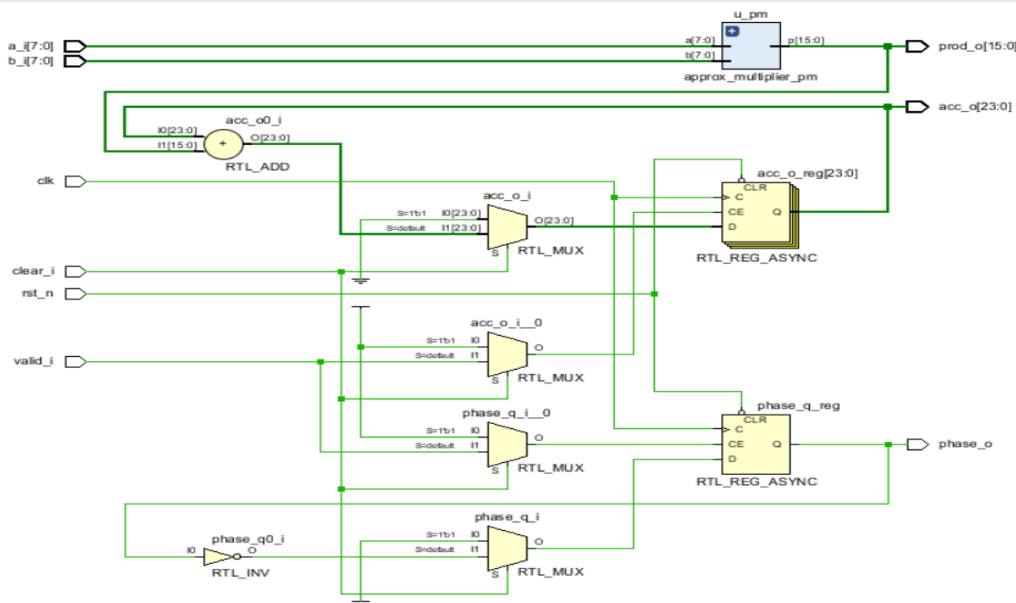
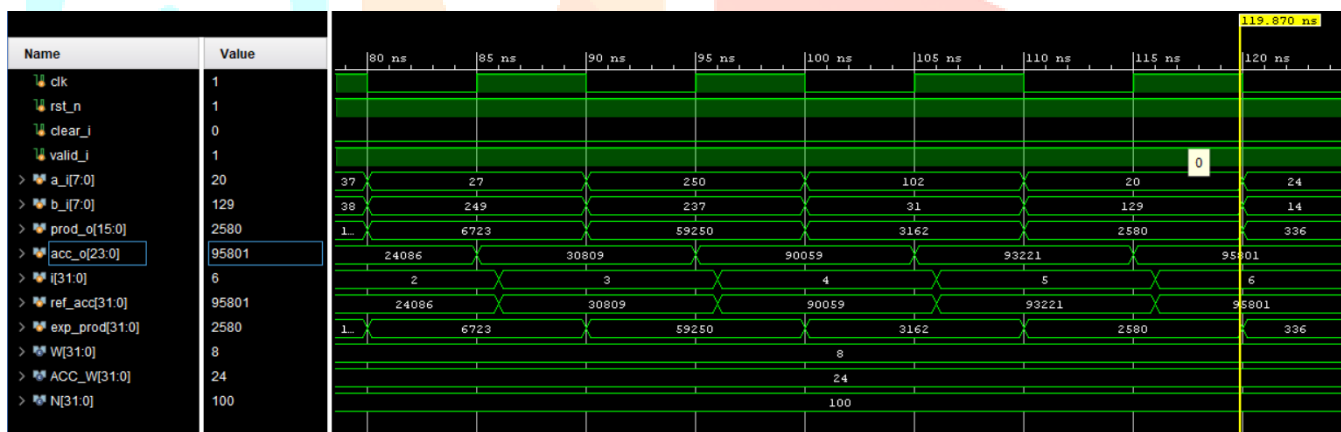


Fig. 4. RTL Schematic of Approximate MAC

Fig. 4 shows the RTL schematic of the proposed approximate MAC design. The schematic illustrates the main functional blocks used in the proposed architecture and their interconnections during MAC operation.



B. Simulation Results

Fig. 5. Exact MAC Waveform

Fig. 5 shows the simulation waveform of the exact MAC design. Different input values were applied to verify the multiplication and accumulation operations. The obtained output matches the expected results.

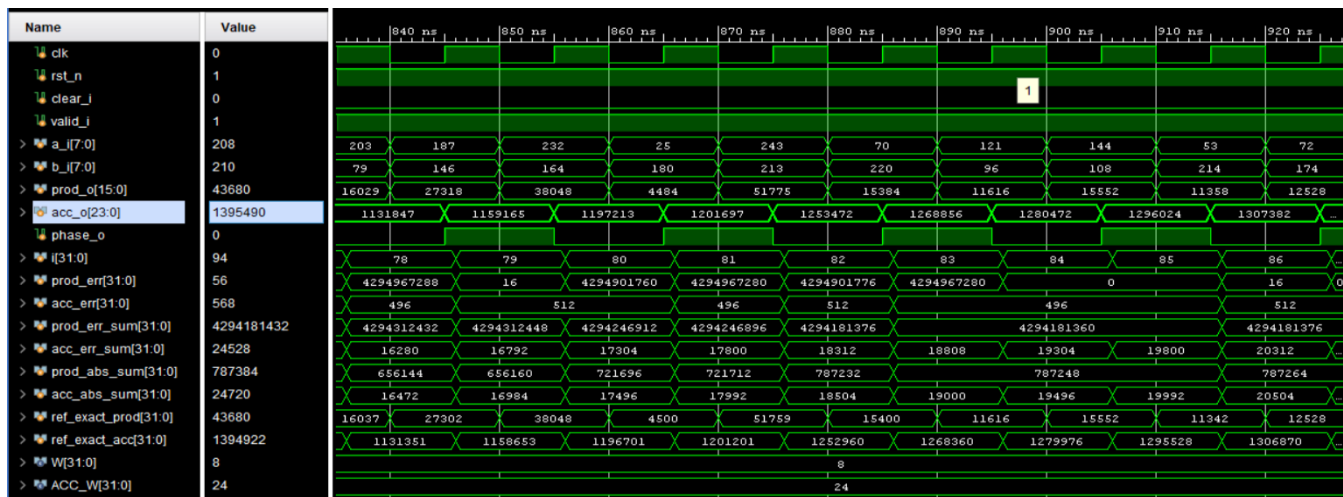


Fig. 6. Approximate MAC Waveform

Fig. 6 shows the simulation waveform of the proposed approximate MAC design. The waveform demonstrates the operation of the proposed architecture for different input conditions. The results indicate proper functioning of the multiplication and accumulation process.

C. Synthesis Results

The synthesis results obtained from Vivado are used to evaluate the performance of both the exact and approximate MAC designs. A comparison of resource utilization and timing parameters is presented in Table 1.

Table 1. Comparison of Exact and Approximate MAC Designs:

Parameter	Conventional Exact MAC	Proposed Approximate MAC
LUTs	111	107
Registers	24	25
Slices	34	36
Power(W)	0.106	0.079
WNS(ns)	6.237	7.409

From Table 1, it can be observed that the proposed approximate MAC design requires fewer LUTs and consumes less power compared to the exact MAC design. The approximate MAC utilizes 107 LUTs compared to 111 LUTs in the exact MAC and reduces power consumption from 0.106 W to 0.079 W. Furthermore, the WNS value improves from 6.237 ns to 7.409 ns, indicating better timing performance. These results demonstrate that the proposed design achieves improved hardware efficiency while maintaining acceptable computational accuracy.

D. Resource Utilization Comparison

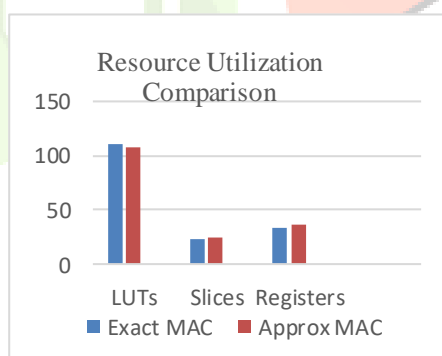


Fig. 7. Resource Utilization Comparison

Fig. 7 shows the comparison of resource utilization between the exact and approximate MAC designs. From the chart, it can be observed that the proposed approximate MAC uses fewer LUTs while maintaining similar resource usage in terms of slices and registers. This indicates that the proposed design is more hardware efficient than the exact MAC design.

VI. CONCLUSION

In this paper, a low-power approximate MAC design for balanced error accumulation has been presented. The proposed architecture utilizes approximate multipliers, a multiplexer, and an interleaved accumulator to perform multiplication and accumulation operations efficiently. The design was implemented using SystemVerilog and verified using Vivado. The synthesis results show that the proposed approximate MAC reduces LUT utilization and power consumption compared to the exact MAC design. The design also achieves improved timing performance while maintaining acceptable accuracy. Hence, the proposed architecture can be considered a suitable solution for low-power VLSI and digital signal processing applications.

VII. REFERENCES

- [1] Y. Wu, C. Chen, W. Xiao, X. Wang and C. Zhuo, "A Survey on Approximate Multiplier Designs for Energy Efficiency," 2023.
- [2] S. Vakili, M. Vaziri, A. Zarei and J. M. Pierre Langlois, "Fast and Low-Cost Approximate Multiplier for FPGAs," 2023.
- [3] V. Leon, M. A. Hanif, G. Armeniakos and M. Shafique, "Approximate Computing Survey: Hardware Approximation Techniques," 2023.
- [4] G. Di Meo, "Approximate MAC Unit Using Static Segmentation," IEEE Transactions on Computers, 2024.
- [5] B. Rashidi, "Efficient and Low-Cost Approximate Multipliers for Image Processing Applications," 2024.
- [6] A. Balasubramanian and P. Maskell, "Approximate Arithmetic Circuits for Low-Power VLSI Applications," 2023.
- [7] R. Kumar and S. Gupta, "Design of Energy Efficient Approximate MAC Architectures," 2024.
- [8] P. Thejaswini and J. Jose, "Energy Efficient Approximate MACs," IEEE INDICON, 2021.
- [9] AMD Xilinx, Vivado Design Suite User Guide, 2023.
- [10] IEEE Standard for SystemVerilog-Unified Hardware Design, Specification and Verification Language, IEEE Std 1800-2017.
- [11] J. Han and M. Orshansky, "Approximate Computing: An Emerging Paradigm for Energy-Efficient Design," 2013.
- [12] V. Gupta, D. Mohapatra, A. Raghunathan and K. Roy, "Low-Power Digital Signal Processing Using Approximate Adders," 2013.