

# Fault Tolerant Low Power Multiplier for Smart Edge Devices

Dondapati Leela Bhuvaneshwari<sup>1</sup>, Muppidi Chaitanya Sai<sup>2</sup>, Panchakarla Viswaja<sup>3</sup>,

Pallikanti Swami Raj<sup>4</sup>

<sup>1</sup>Assistant Professor, Department of Electronics and Communication Engineering, Seshadri Rao

Gudlavalleru Engineering College, Gudlavalleru, India

<sup>2 3 4</sup> Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru

Engineering College, Gudlavalleru, India

## ABSTRACT:

Edge computing platforms require energy-efficient arithmetic units to handle real-time, data-intensive workloads under strict power and area constraints. Conventional multipliers consume considerable power and hardware resources, making them less suitable for resource-limited edge devices. This work proposes a low-power approximate multiplier architecture that employs an optimized 5:2 approximate compressor to improve partial product reduction efficiency. By reducing the number of reduction stages and switching activity, the design achieves lower dynamic power consumption and improved power-delay product (PDP) while maintaining acceptable accuracy for error-tolerant applications. System-level validation using MATLAB-based image processing demonstrates that the proposed multiplier is well suited for signal processing, image processing, and deep neural network workloads in resource-constrained edge environments.

## I. INTRODUCTION:

Edge computing platforms are increasingly deployed in computer vision applications such as health monitoring, surveillance, and smart agriculture, where deep neural networks (DNNs) and signal processing algorithms must execute under stringent power, area, and latency constraints. Since multipliers dominate the computational workload in these systems, their energy and hardware efficiency significantly impact overall performance. Conventional exact multiplier architectures ensure high numerical accuracy but incur substantial energy-area overhead, limiting their suitability for resource-constrained edge devices. Approximate computing offers an effective alternative by exploiting error resilience in many real-time applications to achieve reductions in power and

complexity. In this work, a power-efficient approximate multiplier based on an optimized 5:2 compressor is

proposed to improve partial product reduction and decrease switching activity. The resulting

architecture achieves improved power-delay product (PDP) while maintaining acceptable accuracy for error-tolerant edge workloads.

## A. MOTIVATION:

In power- and area-constrained edge systems, reducing energy consumption is often more critical than achieving exact computational precision. Approximate computing enables controlled accuracy trade-offs to significantly lower hardware complexity and power dissipation. This approach is well suited for AI, DSP, IoT, and multimedia applications, where minor numerical errors do not noticeably affect system performance due to inherent error tolerance. By employing an optimized 5:2 compressor, the proposed approximate multiplier reduces switching activity and circuit complexity, resulting in improved power-delay efficiency and lower area utilization for real-time edge applications.

## B. RELATED RESEARCH

Numerous approximate multiplier architectures have been introduced in recent years, primarily leveraging compressor-based partial product reduction techniques to improve energy efficiency. A significant portion of existing research has concentrated on minimizing error probability by optimizing compressor logic and modifying internal gate structures. While these approaches effectively reduce erroneous output combinations, they often do not achieve proportional

improvements in hardware complexity or power efficiency.

Several approximate 4:2 compressor designs focus on rearranging logic paths and applying error compensation strategies to enhance output accuracy. Other works explore input reordering techniques to decrease the number of critical input patterns contributing to error. Although these strategies reduce error occurrences, the underlying hardware structure and bit representation typically remain unchanged, limiting gains in area and switching reduction.

More aggressive approximate multiplier architectures have been proposed that sacrifice higher accuracy to achieve improvements in speed and power consumption. However, balancing the trade-offs among power, area, delay, and accuracy remains a key challenge in approximate compressor design.

To address these limitations, the present work extends the concept of approximate compression by employing an optimized 5:2 compressor architecture. By increasing compression capability within each reduction stage, the proposed design reduces the number of reduction levels and associated switching activity. This structural enhancement leads to improved power–delay performance while maintaining acceptable error characteristics for error-resilient edge computing applications.

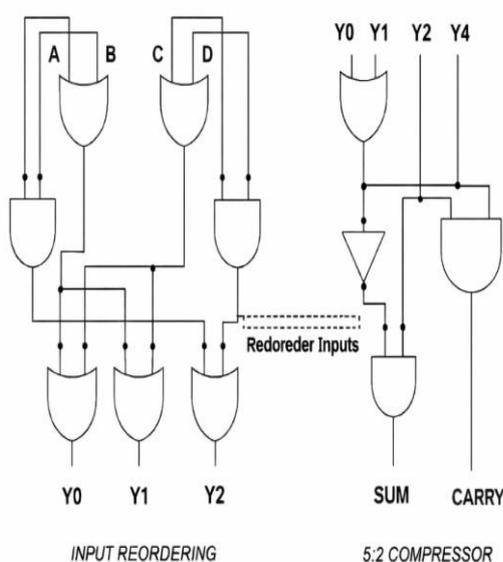


Figure 1. Implementation diagram for 5:2 Approximate Compressor with input recording

## II. PROPOSED ARCHITECTURE

Multiplication is a fundamental and computationally intensive operation in many computer vision and edge computing applications. Since multipliers contribute significantly to power consumption and hardware complexity, designing an energy-efficient multiplier is essential for real-time processing in resource-constrained environments. To address this challenge, this work proposes a low-power approximate multiplier architecture based on a 5:2 approximate compressor with input reordering. The architecture reduces switching activity and compression stages in the partial product reduction process, leading to improved power efficiency and reduced hardware overhead. By combining approximate computation in less significant regions with accurate processing in critical regions, the proposed design achieves a

balance between energy efficiency and acceptable computational accuracy, making it suitable for edge and error-tolerant applications.

### A. Approximate 5:2 Compressor with Input Reordering:

This section presents the proposed approximate 5:2 compressor integrated with an input reordering mechanism for use in an 8-bit approximate multiplier. In multiplier architectures, compressors play a critical role in partial product reduction, directly influencing hardware complexity, power dissipation, and delay. Increasing the compression capability from 4:2 to 5:2 enables greater reduction per stage, thereby decreasing the number of reduction levels and overall switching activity.

The input reordering circuit first rearranges the five primary inputs A,B,C,D,EA, B, C, D, EA,B,C,D,E into reordered signals Y0,Y1,Y2,Y3,Y\_0, Y\_1, Y\_2, Y\_3,Y0 ,Y1,Y2,Y3, and Y4Y\_4Y4. The reordering process groups input combinations based on the number of logic ‘1’s rather than their positional arrangement. Since multiple input patterns may contain the same number of ‘1’s, they produce identical compressor outputs. By exploiting this property, redundant switching transitions are minimized, leading to reduced dynamic power consumption with negligible area overhead.

The reordered outputs are directly applied to the 5:2 approximate compressor block, which generates the corresponding Sum and Carry signal.

Instead of treating all 32 possible input combinations independently, the reordering technique maps them into a smaller set of effective combinations based on their Hamming weight. This reduction decreases internal logic complexity and switching probability.

In approximate designs, small controlled deviations in certain low-probability input cases are acceptable to achieve improvements in hardware efficiency. The Boolean expressions for the reordered signals are derived from the input grouping strategy, and the Sum and Carry outputs are formulated accordingly to balance power reduction and acceptable error characteristics. By combining higher compression efficiency with input-aware approximation, the proposed 5:2 compressor achieves improved power–delay performance while maintaining suitability for error-resilient edge computing applications

$$\text{Sum} = Y_0 \cdot Y_1$$

$$\text{Carry} = Y_0(Y_1 + Y_2)$$

### B. Approximate Multiplier

The proposed multiplier is divided into three regions: the Least Significant Region (LSR), Approximate Region, and Accurate Region to balance power efficiency and accuracy. The LSR contains lower-order columns that contribute less to the final result and are simplified to

reduce hardware complexity, with a constant correction term added to compensate for truncation error.

The intermediate columns form the Approximate Region, where partial products are reduced using half adders and the proposed 5:2 approximate compressors with input reordering, thereby decreasing reduction stages and switching activity. The most significant columns constitute the Accurate Region and are processed using precise adders to maintain reliability. Finally, the compressed partial products are summed using ripple carry adders to produce the final output.

## IV. RESULTS AND DISCUSSION

### A. ERROR AND PERFORMANCE ANALYSIS

To evaluate the effectiveness of the proposed approximate multiplier, simulations were performed using Verilog in Xilinx Vivado and MATLAB for image processing analysis. The objective of this evaluation is to analyze the performance of the multiplier in terms of error metrics, power consumption, delay, hardware utilization, and image quality preservation.

#### 1. Functional Simulation

The functional correctness of the multiplier was verified using simulation waveforms generated in Vivado. The simulation results show the input operands  $a[7:0]$  and  $b[7:0]$  along with the corresponding output  $out[16:1]$ , confirming the correct operation of the multiplier for different

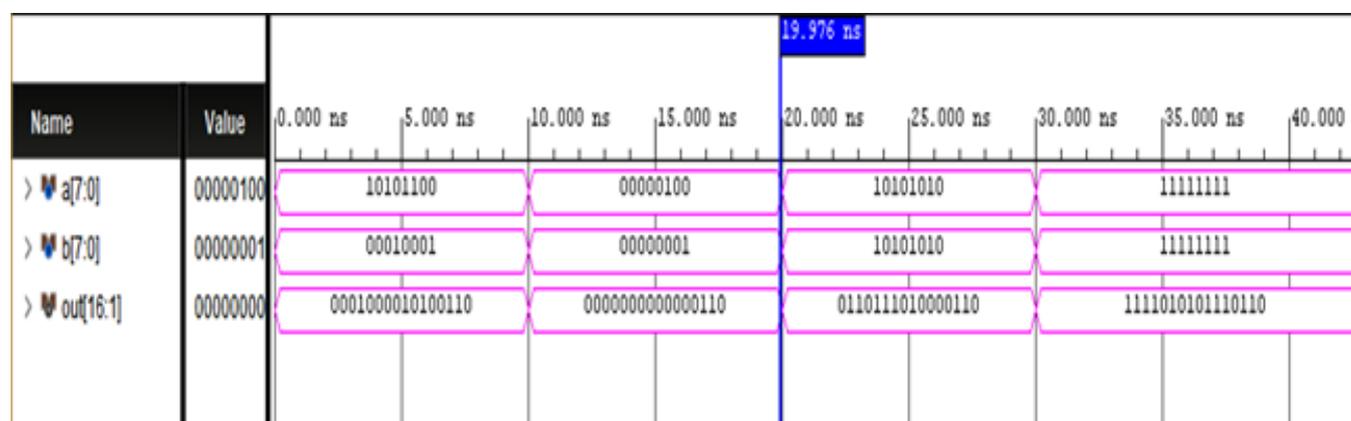


Figure 2. Simulation waveform

input combinations. The waveform results demonstrate that the designed architecture correctly produces multiplication outputs while utilizing approximate

compression logic. This verifies the functional behavior of the proposed multiplier before performing further performance analysis.

## 2. Hardware Resource Utilization (Area Analysis)

Although the design is implemented in simulation, the synthesis results from Vivado provide an estimation of hardware resources required for implementation. The synthesis report indicates that the proposed multiplier utilizes 67 Slice LUTs and 32 bonded I/O blocks. The reduced LUT utilization demonstrates that the architecture requires relatively fewer logic resources, which directly contributes to lower hardware complexity and improved efficiency when deployed on FPGA-based systems.

Name	Slice LUTs (134600)	Bonded IOB (400)
project_1	67	32

Figure 3. Area

## 3. Delay and Power Performance

The timing and power estimation results obtained from the synthesis analysis show that the proposed multiplier achieves a propagation delay of 9.139 ns and a power consumption of 10.038 W. These values indicate that the multiplier provides efficient computational performance while maintaining reduced hardware overhead. The lower delay ensures faster arithmetic operations, which is beneficial for real-time signal and image processing applications.

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay
Path 1	∞	10	11	16	a[7]	out[14]	9.139
Path 2	∞	10	11	16	a[7]	out[16]	9.139
Path 3	∞	10	11	16	a[7]	out[15]	9.120
Path 4	∞	9	10	16	a[7]	out[13]	8.661
Path 5	∞	9	10	16	a[7]	out[12]	8.313
Path 6	∞	8	9	16	a[7]	out[11]	7.843
Path 7	∞	7	8	16	a[7]	out[10]	7.326
Path 8	∞	7	8	8	b[2]	out[9]	6.744
Path 9	∞	5	6	17	a[2]	out[8]	6.176
Path 10	∞	4	5	11	a[1]	out[7]	5.362

Figure 4. Delay

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

**Total On-Chip Power:** 10.038 W  
**Design Power Budget:** Not Specified  
**Power Budget Margin:** N/A  
**Junction Temperature:** 43.8°C  
**Thermal Margin:** 41.2°C (21.8 W)  
**Effective  $\theta_{JA}$ :** 1.9°C/W  
**Power supplied to off-chip devices:** 0 W  
**Confidence level:** Low  
[Launch Power Constraint Advisor](#) to find and fix invalid switching activity

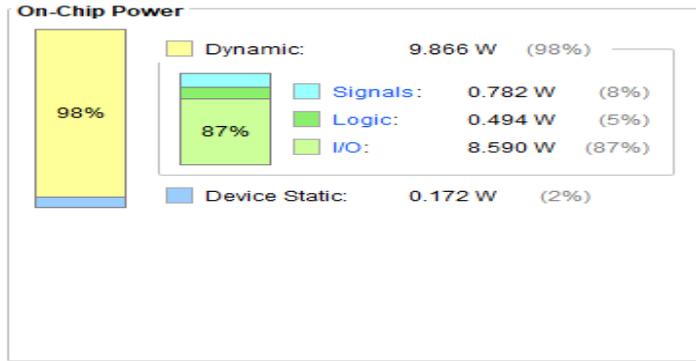


Figure 5. Power

## B. ERROR METRICS EVALUATION

### 1. Error Performance Comparison of Multipliers

This table compares the error metrics Mean Relative Error Distance (MRED) and Normalized Mean Error Distance (NMED) of the proposed multiplier with existing architectures such as MUL-1 and MUL-2. The proposed design achieves significantly lower error values (MRED = 0.0038 and NMED = 0.002388), indicating that the approximation technique introduces only minimal

deviation from exact results while maintaining computational efficiency.

Multiplier	PSNR	SSIM	SNR
Proposed	13.9916dB	0.9241	12.7153dB
Exact	12.4535	0.0388	-

Table 2. Error Performance Comparison of Multipliers

### 2. Performance Metrics of the Proposed Multiplier

This table presents the hardware performance parameters of the proposed multiplier, including power consumption, area utilization, and delay obtained from synthesis results. The architecture requires 67 LUTs, consumes 10.038 W power, and achieves a delay of

9.139 ns, demonstrating reduced hardware complexity and efficient computational performance suitable for edge applications.

### 3. Image Quality Evaluation Metrics

This table evaluates the impact of approximation on image processing applications using PSNR, SSIM, and SNR metrics. The proposed multiplier achieves PSNR = 13.9916 dB, SSIM = 0.9241, and SNR = 12.7153 dB, indicating that the output image maintains high structural similarity and acceptable visual quality compared to the exact multiplier.

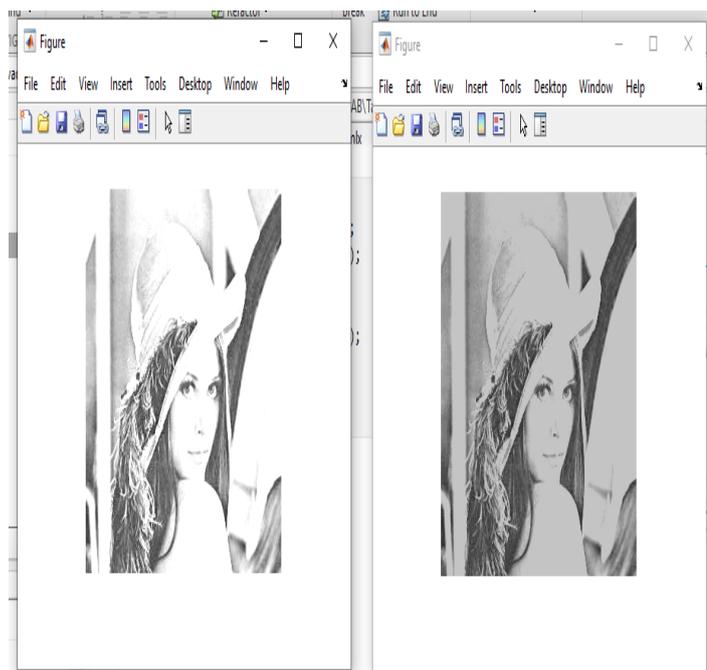
Multiplier	MRED	NMED
PROPOSED	0.0038	0.002388
MUL-1[8]	1.3	7.1
MUL-2[8]	1.1	3.2

Table 3. Image Quality Evaluation Metrics

### 4. Image Quality Evaluation

The impact of approximation on image processing applications was evaluated using MATLAB-based image multiplication experiments. The input

grayscale image and the processed output image are shown in the results. Visual comparison indicates that the output image preserves most of the structural details and visual quality of the original image, with only negligible distortion caused by approximation.



**Figure 6. Difference between image quality**

To quantitatively evaluate image quality, metrics such as Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Signal-to-Noise Ratio (SNR) were computed. The proposed multiplier achieves a PSNR of 13.9916 dB, SSIM of 0.9241, and SNR of 12.7153 dB. The SSIM value close to 1 indicates strong structural similarity between the original and processed images, demonstrating that the approximation has minimal perceptual impact.

## CONCLUSION

The experimental results confirm that the proposed approximate multiplier achieves an effective balance between accuracy, computational efficiency, and hardware resource utilization. The architecture demonstrates low error metrics, reduced LUT utilization, acceptable delay performance, and high image similarity in practical image processing tasks. These results indicate that the proposed design is suitable for error-resilient edge computing applications, where power efficiency and reduced hardware complexity are more critical than exact arithmetic precision.

## REFERENCES

- [1] A. Hazarika, S. Poddar, M. M. Nasralla, and H. Rahaman, "Area and energy efficient shift and accumulator unit for object detection in iot applications," *Alexandria Engineering Journal*, vol. 61, no. 1, pp. 795-809, 2022.
- [2] S. Amanollahi and G. Jaberipur, "Fast energy efficient radix-16 sequential multiplier," *IEEE Embedded Systems Letters*, vol. 9, no. 3, pp. 73-76, 2017.
- [3] M. T. Khan and J. Hazarika, "An area and energy efficient serial multiplier," *IEEE Embedded Systems Letters*, vol. 16, no. 4, pp. 425-428, 2024.
- [4] X. Yi, H. Pei, Z. Zhang, H. Zhou, and Y. He, "Design of an energy efficient approximate compressor for error-resilient multiplications," in *Int. Symp. Circuits Syst. (ISCAS)*. IEEE, 2019, pp. 1-5.
- [5] U. A. Kumar, S. K. Chatterjee, and S. E. Ahmed, "Low-power compressor-based approximate multipliers with error correcting module," *Embedded Systems Letters*, vol. 14, no. 2, pp. 59-62, 2021.
- [6] M. Ha and S. Lee, "Multipliers with approximate 4-2 compressors and error recovery modules," *Embedded Systems Letters*, vol. 10, no. 1, pp. 6-9, 2017.
- [7] Y. Xu, Y. Guo, and S. Kimura, "Approximate multiplier using reordered 4-2 compressor with or-based error compensation," in *Int. Conf. ASIC (ASICON)*. IEEE, 2019, pp. 1-4.
- [8] L. H. Krishna, A. Sk, J. B. Rao, S. Veeramachaneni, and N. M. Sk, "Energy efficient approximate multiplier design with lesser error rate using the probability-based approximate 4: 2 compressor," *Embedded Systems Letters*, 2023.