



## DESIGN AND EVALUATION OF FINFET-BASED APPROXIMATE MODIFIED BOOTH MULTIPLIER FOR ENERGY-EFFICIENT VLSI SYSTEMS

Veera Boopathy E.<sup>1\*</sup>, Kartik Subramaniam<sup>2</sup>, Jeniton S.<sup>3</sup>, Charmila D.<sup>4</sup>,  
Peer Mohamed Appa M AY<sup>5</sup>, Manojkumar B.<sup>6</sup>

<sup>1</sup>Department of Electronics and Communication Engineering, Karpagam  
Institute of Technology, Coimbatore, India-641021,

<sup>2</sup>Independent Researcher, Massachusetts, United States of America-01851,

<sup>3</sup>Department of Electronics and Communication Engineering, Rathinam  
Technical Campus, Coimbatore, India-641021,

<sup>4</sup>Department of Electrical and Electronics Engineering, Nandha College of  
Technology, Erode, India-638052,

<sup>5</sup>Department of Computer Science and Engineering, School of Computing,  
Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and  
Technology, Chennai, India-600062,

<sup>6</sup>Department of Electronics and Communication Engineering, KGiSL Institute  
of Technology, Coimbatore, India-641035

Email: <sup>1</sup>boopathy.veera@gmail.com, <sup>2</sup>skartik.author@gmail.com,  
<sup>3</sup>jenitonrajesh@gmail.com, <sup>4</sup>mahescharmi4667@gmail.com, <sup>5</sup>peer.appa@gmail.com,  
<sup>6</sup>manoj.dreamtwin@gmail.com

Corresponding Author: **Veera Boopathy E.**

<https://doi.org/10.26782/jmcmms.2026.10.00002>

(Received: July 24, 2026; Revised: September 28, 2026; Accepted: October 10, 2026)

### Abstract

*In the progress of energy-efficient digital multiplication systems, approximate multipliers have been proven to be a good solution to attain high energy efficiency with acceptable computational precision. This paper proposes a 22 nm FinFET-based approximate Modified Radix-8 Booth multiplier which is based on Modified Radix-8 Booth recoding, selective partial-product approximation, controlled truncation, and an optimized reduction and accumulation network to minimize arithmetic complexity and switching activity. This architecture was modeled in VHDL and designed using an ASIC design flow using a 22 nm standard-cell library. Post-layout evaluation showed about 63.4% silicon area reduction, 55.1% propagation delay reduction, and 65.6% power consumption reduction when compared with an exact Booth multiplier with lower Power-Delay Product (PDP) and Area-Power Product (APP). The application-level validation with representative image processing operations resulted in a PSNR*

*Veera Boopathy E. et al.*

*of 41.2 dB and SSIM of 0.995, which validated the proposed approximation strategy to maintain high computational quality. The proposed multiplier is well suited for low-power digital signal processing, multimedia processing, and embedded VLSI systems as it gives an effective trade-off between implementation efficiency and arithmetic accuracy.*

**Keywords:** Radix-8, Booth Recoding, truncation, approximate, image processing.

---

## **I. Introduction**

With the development of digital signal processing (DSP), multimedia computing, computer vision, wireless communication, artificial intelligence inference, and embedded systems, the requirement for high-performance arithmetic hardware has been increasing rapidly. Multiplication is a commonly performed task and constitutes the basic operation in convolution, filtering, matrix multiplication, transform computation, and image enhancement algorithms [XIII]. Therefore, the performance of the multiplier has a direct impact on system throughput, power consumption, silicon utilization, and computational latency. Recently, as semiconductor technology has been scaled to the nanometer range, the challenge of designing multiplier architectures that can maintain high speed, low power, and compact hardware implementation has become a key design constraint in very-large-scale integration (VLSI) systems [XI].

In the conventional binary and array multipliers, there is a large number of partial products and extensive reduction circuits, which leads to high switching activity, silicon overhead, and high propagation delay [XII]. To overcome these restrictions, Booth multiplication has been extensively used, in which the redundant arithmetic operations are eliminated by an efficient recoding of the bits of the multiplier. In particular, Modified Radix-4 Booth recoding decreases the quantity of partial products produced during the multiplication process as contrasted to the conventional multiplication process that improves computation efficiency [IX]. As operand precision increases, however, Booth-based architectures need complex reduction and accumulation networks, which increase routing complexity, logic utilization, and power dissipation. The above constraints have inspired the exploration of alternative Booth encoding methods with higher radix and alternative architectural optimizations with the goal of further reducing the arithmetic complexity.

The concept of approximate computing has proven to be an effective design approach for boosting the implementation efficiency of arithmetic circuits for error-resilient applications. Approximate arithmetic selectively simplifies arithmetic operations that are least likely to affect the quality of application output, rather than keeping numbers exactly the same for all computations. This design method is especially well suited to image processing, multimedia processing, and DSP applications because the small difference in numbers does not tend to have a noticeable impact on the perceptual results [V]. Approximate multipliers can achieve high energy efficiency with good accuracy by minimizing switching activity, combinational logic, and intermediate arithmetic operations. However, many existing architectures are not easily applicable because of high numerical errors introduced by excessive approximation. Hence, the need for an effective trade-off among hardware efficiency and computational correctness is a major requirement for designing approximate multipliers. With the

*Veera Boopathy E. et al.*

rapid progress of FinFET technology, new opportunities have emerged to improve the implementation efficiency of arithmetic circuits. Compared to planar CMOS transistors, FinFET transistors provide improved electrostatic control of the gate, lower leakage current, and better switching characteristics, which result in lower static power consumption and better timing performance. However, the arithmetic complexity of traditional multiplier architectures remains despite device-level enhancements [VIII].

In response to these challenges, this paper presents a 22 nm approximate Modified Radix-8 Booth multiplier based on Modified Radix-8 Booth recoding, selective partial-product approximation, controlled truncation, and an optimized reduction and accumulation network, which is integrated as an arithmetic framework. The proposed architecture is first modeled in VHDL and then implemented using a complete ASIC design flow provided by Cadence Genus, Cadence Innovus, and a 22 nm FinFET standard-cell library. Silicon area, propagation delay, power consumption, Power-Delay Product (PDP), and Area-Power Product (APP) are utilized to compute hardware performance, whereas Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) are used for application-level validation using representative image-processing operations [XI]. The experimental outcomes verify that the proposed architecture recommends an effective trade-off among computational accuracy, area utilization, and energy efficiency, building a potential multiplication result for future DSP, multimedia, and low-power embedded VLSI systems.

## **II. Literature Survey**

In the context of digital signal processing (DSP), multimedia processing, wireless communication, computer vision, and embedded applications, which involve the majority of computational tasks, efficient multiplication is crucial for modern digital systems. The multiplier is a major element of the system latency, silicon utilization, and energy consumption in convolution, filtering, transform computation, matrix multiplication, and image enhancement algorithms. Conventional array multipliers and exact binary multipliers require a large number of partial products and extensive reduction circuits, which leads to high switching activity, large power dissipation, and long propagation delay. With the continuous scaling towards nanotechnology, these constraints have made energy-efficient, hardware-effective, and faster multiplier architectures more in demand [XV].

Booth multiplication is a widely used algorithm due to its ability to minimize redundant arithmetic operations by performing efficient recoding of multiplicand bits. The original Radix-2 Booth algorithm decreases the quantity of partial products produced by using signed digits to represent each multiplier bit, and Modified Radix-4 Booth recoding takes it a step further by processing multiple bits at once. Typically, the critical paths and hardware complexity of Booth-based architectures are shorter and simpler than in conventional array multipliers [X]. As the precision of the operands grows, however, even Modified Radix-4 Booth multipliers create a large number of intermediate partial products, which need complex reduction and accumulation networks, leading to higher implementation cost, longer carry propagation delay, and higher routing complexity. Thus, further optimization is needed to achieve significant enhancements in high-performance VLSI systems [VI].

*Veera Boopathy E. et al.*

To overcome these limitations, Modified Radix-8 Booth recoding has been given significant attention as it further minimizes the quantity of partial products generated by encoding overlapping sets of multiplier bits into signed Booth coefficients. Combinational logic depth, switching activity, and critical-path delay are reduced due to the reduced arithmetic workload entering the accumulation stage [II]. However, the increased complexity of the encoder and wider intermediate arithmetic structures by higher-radix Booth encoding can partially counteract the hardware savings expected in the encoder [III]. Therefore, it is important to consider using other complementary optimization methods along with Booth recoding in the construction of low-power and high-speed arithmetic circuits, which leads to the concept of hybrid multiplier architecture [IV].

Approximate computing has been a successful technique for enhancing the implementation efficiency of arithmetic circuits with error-resilient applications. Approximate arithmetic, instead of maintaining high precision for all calculations, simplifies calculations with only a small impact on application output precision. Because small numerical differences are tolerated in multimedia processing, image enhancement, and some DSP algorithms, approximate multipliers can significantly reduce switching activity, logic utilization, propagation delay, and energy consumption [XII]. Prior approaches to approximation include logic simplification, approximate compressors, operand truncation, speculative arithmetic, and approximation of partial-product generation. These techniques simplify the hardware design, but often incur a high degree of approximation, which can lead to large computational errors for applications that demand a high level of numerical accuracy [X]. Several studies focus on selective approximation, that is, restricting the approximation to low-significant arithmetic operations while maintaining high-order arithmetic operations. This approach helps reduce the switching activity and the hardware complexity while preserving a similar accuracy level at the application level [VI]. However, many architectures reported still use relatively complex reduction networks to make improvements in propagation delay and silicon utilization. Moreover, there are several designs for the approximate multiplier that focus on a single hardware measure, like silicon area or power consumption, but have only a minor impact on the overall efficiency of the implementation. Thus, designing a multiplier architecture with minimum hardware cost, propagation delay, power consumption, and approximation errors is still an important research challenge [VII].

The recent continual scaling of semiconductor technology has reinforced the critical need to combine high-performance arithmetic architectures with innovative device technology. The leakage current, short-channel effects, and increasing power density of conventional planar CMOS devices are becoming limiting factors for circuit performance and reliability with the scaling of the physical device sizes. To overcome these challenges, FinFET technology employs a three-dimensional multi-gate transistor structure to improve electrostatic gate control, leakage current, and switching characteristics [III]. As a result, the arithmetic circuits implemented using FinFETs show lower static power consumption and better timing performance than the corresponding planar CMOS circuits. Many reported multipliers based on FinFET transistors still use traditional exact arithmetic architectures, which makes it difficult

to realize the full benefits of advanced transistor technology, and their arithmetic complexity is not significantly reduced [II].

The practical research on approximate multipliers has also gained more significance. Arithmetic circuits with high computational speed and acceptable output quality are needed for multiplication-heavy operations like image filtering, edge detection, blending, and extraction of features [XIV]. Thus, Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) are broadly used to assess the impact of arithmetic approximation on processed images [I]. Previous research has shown that perceptual image quality can be maintained at a low implementation cost through careful control of approximation. Thus, the literature shows three main challenges. First, the conventional Booth multipliers still produce a relatively high number of partial products, which makes the arithmetic more complicated and restricts the improvement of the propagation delay. Second, many approximation-based architectures gain hardware benefits by introducing too much computational error, leading to loss of accuracy in the application. Third, the benefits of the FinFET technology are not realized completely since many implementations still use conventional arithmetic structures that have complex reduction networks and large switching activity. These restrictions suggest that a common multiplier architecture is required to optimize the multiplication algorithm, the architecture for arithmetic operations, and the implementation technology.

With these observations, the proposed work proposes an Approximate Modified Radix-8 Booth multiplier in 22 nm FinFET technology that adopts the Modified Radix-8 Booth recoding, partial-product approximation, controlled truncation, and an optimized reduction and accumulation network in a single arithmetic framework. The architecture is designed with Cadence Genus, Cadence Innovus, and a 22 nm FinFET standard-cell library for realistic post-layout evaluation of silicon area, propagation delay, power consumption, Power-Delay Product (PDP), Area-Power Product (APP), and application-level image quality based on PSNR and SSIM. The proposed architecture offers a balanced multiplication solution for high-performance, low-power DSP, multimedia, and embedded VLSI systems by integrating algorithm-level optimization together with advanced FinFET technology.

### **III. Proposed Methodology**

The proposed approximate Modified Radix-8 Booth multiplier uses 22 nm FinFET technology to increase the computational speed, energy efficiency, and hardware utilization for multiplication-intensive digital systems. Multiplication is a basic arithmetic task used repeatedly in digital signal processing (DSP), multimedia processing, image enhancement, and embedded computing, so it consumes most of the power and execution time. While conventional Booth multipliers use recoding to simplify multiplication, more precision in the operands leads to a larger quantity of partial products and a complicated reduction network, thus higher propagation delay, switching activity, and hardware overhead. To alleviate these drawbacks, the proposed architecture incorporates Modified Radix-8 Booth recoding, Selective Partial-Product Approximation (SPPA), Controlled Truncation, and an optimized Reduction and Accumulation network using a single arithmetic framework. The entire architecture is modeled in synthesizable VHDL and implemented via a standard ASIC design flow

*Veera Boopathy E. et al.*

with Cadence Genus, Cadence Innovus, and a Cadence standard-cell library in 22 nm FinFET technology.

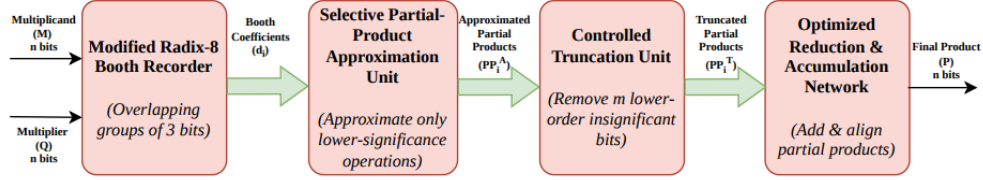


**Fig. 1.** Architecture of proposed methodology.

The overall architecture of the proposed multiplier is presented in Fig. 1. Overlapping groups of the multiplier bits are first converted to signed Booth coefficients using the Modified Radix-8 Booth recoder. This recoding substantially decreases the quantity of generated partial products before doing arithmetic reduction. The resulting partial products then pass through a selective approximation unit that streamlines lower-significance calculations but maintains higher-significance calculations. The controlled truncation removes insignificant lower-order bits to decrease the width and arithmetic of the intermediate operands. Finally, the partial products are reduced and accumulated into the final multiplication result by an optimized reduction and accumulation network.

In the first stage, Modified Radix-8 Booth recoding is used to minimize the number of arithmetic operations performed in partial product generation. The multiplier operand is broken into overlapping groups of bits, each of which is a signed Booth coefficient. As a result, the quantity of partial products pushed to the reduction phase is less, which decreases the depth of the combinational logic and increases the efficiency of the computation. The Booth-generated partial product is written as

$$PP_i = d_i(M \times 2^{3i}) \quad (1)$$

where  $PP_i$  is the partial product,  $d_i$  is the signed Booth coefficient,  $M$  is the multiplicand, and  $2^{3i}$  is the positional weight of the encoded multiplier group. The partial products produced are then selectively approximated, in which only the lower-significance arithmetic operations are approximated, and these approximations have little effect on the final product of the multiplication. The proposed approach retains higher-order computations to retain computational accuracy while minimizing switching activity and hardware complexity, as compared to fully approximate multipliers. The approximate partial product is shown as

$$PP_i^A = f(PP_i) \quad (2)$$

where  $f(\cdot)$  is the approximation function applied to the Booth-generated partial product. The approximate partial products are further simplified using controlled truncation, where the less significant bits are dropped before they are added together, which is shown to be useful to the arithmetic datapath. This operation reduces the bit width of the operands and the carry propagation with only small effects on the computation accuracy. The partial product is truncated as

$$PP_i^T = \left\lfloor \frac{PP_i^A}{2^m} \right\rfloor \quad (3)$$

where  $m$  is the number of lower-significance bits that are being truncated and  $\lfloor \cdot \rfloor$  is the truncation operation. The other partial products are arranged in a way that reflects their positional values and added together to find the final multiplication result

$$P = \sum_{i=0}^{k-1} PP_i^T \times 2^{3i} \quad (4)$$

where  $k$  is the total number of Booth-encoded groups. The intermediate arithmetic results are combined efficiently by an optimized reduction and accumulation network to get the final multiplication output; the truncated partial products are processed. Both the Modified Radix-8 Booth recoding and the selective approximation require fewer partial products, so that the reduction network can be simpler and more efficient.

The exact modified radix-8 Booth product is characterized as

$$P_{exact} = \sum_{i=0}^{k-1} PP_i \quad (5)$$

The proposed product is

$$P_{prop} = \sum_{i=0}^{k-1} PP'_i \quad (6)$$

Thus, the product error is

$$E = P_{exact} - P_{prop} = \sum_{i=0}^{k-1} (PP_i - PP'_i) \quad (7)$$

where  $E$  denotes the collective approximation and truncation error. The multiplier is described with synthesizable VHDL and simulated with a typical set of input vectors for normal operating and boundary conditions. Using the 22 nm FinFET library of standard cells, the validated design is synthesized with Cadence Genus and then implemented in Cadence Innovus by doing floorplanning, placement, clock-tree synthesis, routing, parasitic extraction, and timing optimization to create a physically realized design.

The multiplier is constructed in 22 nm FinFET technology, which provides better electrostatic control of the gate and lower leakage current than planar CMOS technology. The proposed arithmetic optimizations are complemented by these device characteristics that enhance switching performance and energy efficiency. The dynamic power consumption is measured by the switching activity during the post-layout netlist, and the result is represented as

$$P_{dynamic} = \alpha CV^2 f \quad (8)$$

Here,  $\alpha$  is the switching activity factor,  $C$  is the operative switching capacitance,  $V$  is the supply voltage, and  $f$  is the switching frequency. The proposed architecture reduces dynamic power by reducing switching activity and effective capacitance using Modified Radix-8 Booth recoding, selective approximation, and controlled truncation.

The proposed multiplier is validated by circuit-level and application-level evaluation. Circuit-level metrics are silicon area, propagation delay, power consumption, Power-Delay Product (PDP), and Area-Power Product (APP) after the post-layout implementation with identical synthesis conditions. For purposes of practical application, the created multiplier is incorporated into representative image processing

*Veera Boopathy E. et al.*

operations such as convolutional filtering, blending images, sharpening images, and enhancing the edges of images. Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) metrics are consumed for application-level performance comparison to evaluate processed images with the images generated using an exact Booth multiplier. The proposed methodology includes Modified Radix-8 Booth recoding, selective partial-product approximation, controlled truncation, an optimized reduction and accumulation network, and 22 nm FinFET technology to provide a low-energy multiplier architecture.

#### **IV. Results and Discussion**

The comprehensive circuit-level implementation and application-level validation of the proposed 22 nm approximate Modified Radix-8 Booth multiplier were performed to check its performance in terms of hardware efficiency and computational accuracy. The architecture was described in VHDL and functionally verified by simulation. Post-layout analysis was conducted to calculate the realistic silicon area, propagation delay, power dissipation, Power–Delay Product (PDP), and Area–Power Product (APP). For further assessment of the practical significance of arithmetic approximation, the synthesized multiplier was incorporated into typical image-processing applications: convolutional filtering, blending, sharpening, and edge enhancement. Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) were used to determine image quality.

All designs were modeled in VHDL with 8-bit unsigned operands and synthesized using an identical commercial 22 nm FinFET standard-Vt library and Cadence Genus/Innovus scripts. The exact Radix-8 Booth multiplier and existing Approximate Multipliers 1–3 were re-implemented with their published descriptions. All designs were characterized at 0.8 V, 25 °C with 50 ps clock uncertainty, 100 ps transition, 0 ps latency, standard low-drive input driving cell, and with 10 fF output load. The period of the clock was nominally swept from 1.0 ns to 0.4 ns, and the tightest period achieved for each design is reported in Table 1. The critical path delay is the delay from the primary input to the primary output. Genus utilized high-effort hierarchical synthesis, and Innovus used 70% utilization, 1:1 aspect ratio, all available 22 nm metal layers, and clock tree synthesis (CTS), with Quantus parasitic extraction at the typical RC corner. The area refers to the post-layout standard-cell area. Innovus computed total power (dynamic plus leakage) using 10,000 random functional vectors to simulate the post-layout netlist in Cadence Xcelium. All outcomes are post-layout.

The post-layout implementation results show substantial gains in all the major hardware metrics as listed in Table 1. The proposed architecture occupies a silicon area of 0.00199 mm<sup>2</sup> compared to 0.00543 mm<sup>2</sup> of the exact Booth multiplier, which is a reduction of 63.4%. The propagation delay is reduced from 0.78 ns to 0.35 ns, which is a 55.1% improvement, and the power consumption is decreased from 1.8746 mW to 0.6456 mW, a 65.6% reduction.

**Table 1: Hardware performance comparison of multiplier architectures**

| Multiplier Type                   | Area (mm <sup>2</sup> ) | Delay (ns) | Power (mW) | PDP (fJ) | APP (mm <sup>2</sup> ·mW) |
|-----------------------------------|-------------------------|------------|------------|----------|---------------------------|
| Exact Radix-8 Booth Multiplier    | 0.00543                 | 0.78       | 1.8746     | 1462.19  | 0.01018                   |
| Existing Approximate Multiplier 1 | 0.00265                 | 0.52       | 0.6705     | 348.66   | 0.00178                   |
| Existing Approximate Multiplier 2 | 0.00368                 | 0.63       | 1.2874     | 811.06   | 0.00474                   |
| Existing Approximate Multiplier 3 | 0.00401                 | 0.59       | 1.3507     | 796.91   | 0.00542                   |
| Proposed Approximate Multiplier   | 0.00199                 | 0.35       | 0.6456     | 225.96   | 0.00128                   |

The proposed multiplier also provides the minimum Power–Delay Product (225.96 fJ) and Area–Power Product (0.00128 mm<sup>2</sup>·mW) values, which indicate simultaneous speed, energy, and hardware efficiency. The improvements are mainly due to a synergistic combination of Modified Radix-8 Booth recoding, selective partial-product approximation, controlled truncation, and an optimized reduction and accumulation network. The number of partial products generated by Modified Radix-8 Booth recoding is considerably reduced, which reduces the arithmetic complexity before accumulation. Selective approximation makes it easier to compute the lower significance while maintaining higher significance arithmetic accuracy. The controlled truncation also reduces intermediate operand width and carry propagation, thus decreasing the logic usage and switching activity. This allows the optimized reduction network to have a smaller number of arithmetic operations, and hence a shorter critical path and lower dynamic power consumption. Table 2 separates the contribution of each of the architectural elements, with respect to realizing the same exact Radix-8 multiplier at similar synthesis constraints.

**Table 2: Ablation analysis of architectural components**

| Architecture                   | Area (mm <sup>2</sup> ) | Delay (ns) | Power (mW) | PDP (fJ) | APP (mm <sup>2</sup> ·mW) | Error (MED) |
|--------------------------------|-------------------------|------------|------------|----------|---------------------------|-------------|
| Exact Radix-8                  | 0.00543                 | 0.78       | 1.8746     | 1462.19  | 0.01018                   | 0           |
| Radix-8 + SPPA                 | 0.00412                 | 0.61       | 1.2148     | 741.03   | 0.00500                   | 0.82        |
| Radix-8 + Truncation           | 0.00336                 | 0.52       | 0.9824     | 510.85   | 0.00330                   | 1.125       |
| Radix-8 + Optimized Reduction  | 0.00271                 | 0.43       | 0.7815     | 336.05   | 0.00212                   | 1.125       |
| Complete Proposed Architecture | 0.00199                 | 0.35       | 0.6456     | 225.96   | 0.00128                   | 1.125       |

**Table 3: Exhaustive Error Analysis**

| Metric                          | Proposed Multiplier |
|---------------------------------|---------------------|
| Maximum Error Distance (MED)    | 3                   |
| Mean Error Distance (MEd)       | 1.125               |
| Normalized Mean Error (NME)     | 0.00173%            |
| Worst-Case Relative Error (WRE) | 100%                |

Table 3 summarizes the arithmetic error associated with the proposed multiplier obtained through a comprehensive simulation of the entire input space. Product error is directly related to  $f(\cdot)$  and  $m$ , where  $f(\cdot)$  is the approximation function, and  $m$  is the truncation parameter. The accuracy decreases with increasing approximation and  $m$ , but setting  $m$  to lower-significance bits will keep the accuracy acceptable.

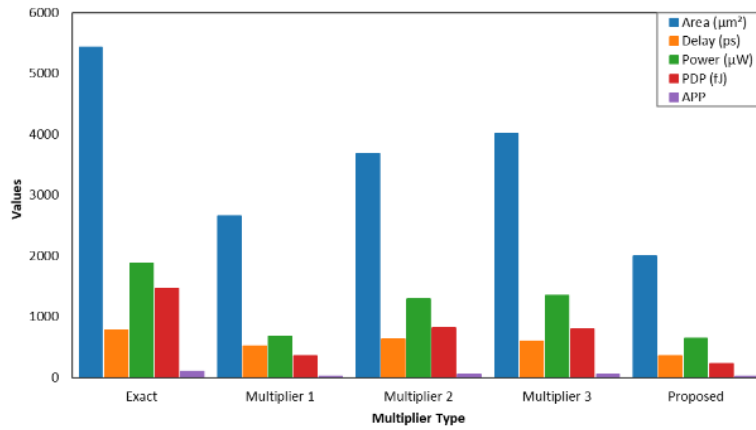


**Fig. 2.** Comparative hardware performance of the exact Booth multiplier and the proposed multiplier.

The comparative performance of evaluated multipliers in terms of hardware is shown in Fig. 2. The proposed architecture achieves superior performance in all the metrics evaluated compared to the exact Booth multiplier and the current approximate multipliers. This simultaneous reduction of silicon area, propagation delay, and power consumption shows that the proposed optimization strategy works well to achieve hardware efficiency while satisfying the computational requirement, with minimal additional approximation error. The synthesized design was simulated with various timing constraints to assess the scalability of its implementation. The silicon area was almost unchanged for the various timing target cases, which shows that the proposed architecture has stable hardware utilization, as shown in Fig. 3. Similarly, the power consumption of the higher-frequency runs was higher due to more switching activity, but was still much lower than that of the exact Booth implementation, as illustrated in Fig. 4.

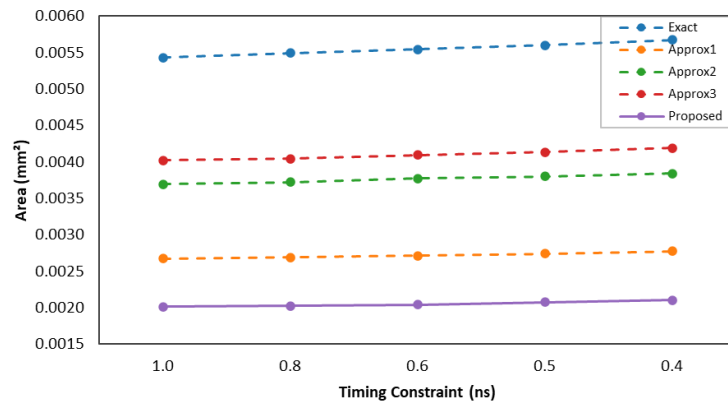


**Fig. 3.** Silicon-area variation under different timing constraints.

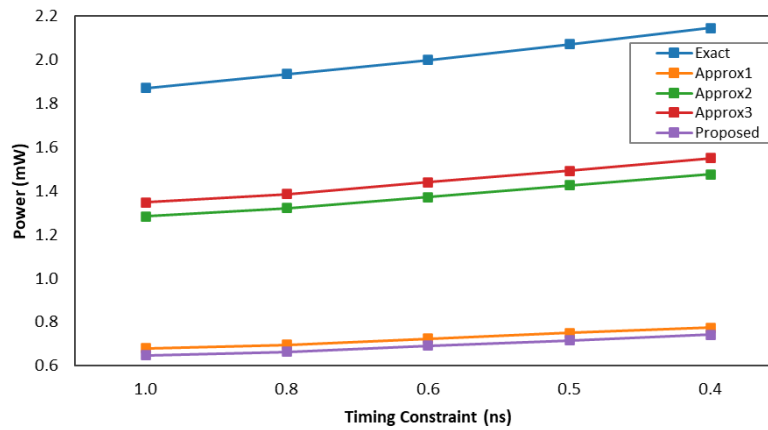


**Fig. 4.** Power consumption under different timing constraints.

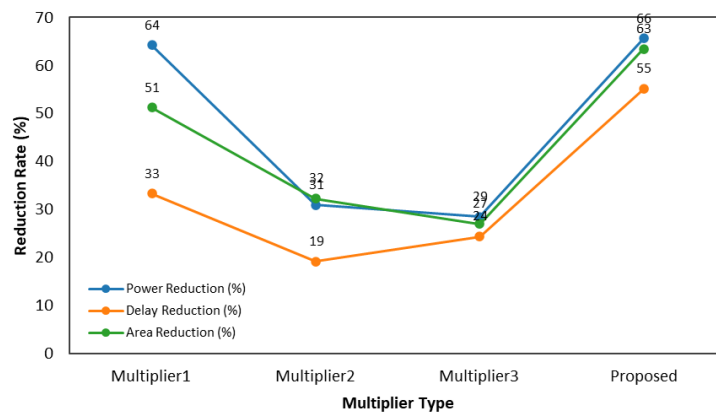


**Fig. 5.** Normalized comparison of hardware performance metrics.

The balanced optimization obtained by the proposed multiplier is further confirmed from the following normalized comparison, as shown in Fig. 5. The hardware metrics continue to be significantly lower for all normalized metrics, which shows that there are significant gains in the area efficiency, timing, and energy metrics, even with

*Veera Boopathy E. et al.*

concurrent improvements. The proposed multiplier was applied to representative image processing tasks such as image blending, convolution-based filtering, image sharpening, and edge enhancement to investigate the application-level performance. These applications carry out a high number of multiplication operations, making them suitable places to test the impact of the use of arithmetic approximation on the quality of the resulting image.

**Table 4: Application-level image-quality evaluation**

| Metric    | Proposed Multiplier | Exact Booth Multiplier |
|-----------|---------------------|------------------------|
| PSNR (dB) | 41.2                | 36.0                   |
| SSIM      | 0.995               | 0.970                  |

The proposed multiplier has a Peak Signal-to-Noise Ratio (PSNR) of 41.2 dB and a Structural Similarity Index (SSIM) of 0.995, as tabulated in Table 4, which shows that the output of the proposed multiplier is in excellent agreement with the output of the exact Booth multiplier. The PSNR value is higher than the standard of 40 dB for high-quality image reconstruction, and the SSIM value is close to 1, which means that the reconstructed image contains little structural information loss. The results show that restricting approximation to lower-significance computations has a minimal perceptual impact and significantly boosts hardware efficiency.

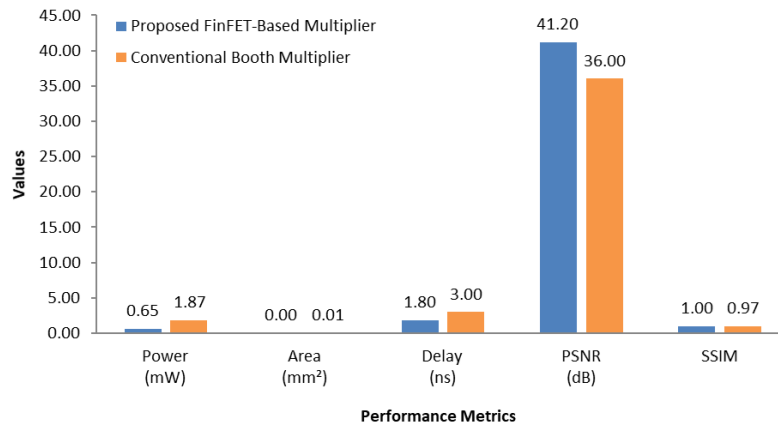


**Fig. 6.** Representative image-processing results using the proposed multiplier.

The visual results shown in Fig. 6 also illustrate that the proposed approximation strategy captures the information of the edges, texture features, and overall image fidelity. The images generated by the approximation do not introduce any noticeable visual artifacts, thus proving that the approximation error is well controlled for lower-significance arithmetic operations. The results shown in Fig. 7 illustrate the smoother intensity transitions and fewer approximation-induced artifacts with a PSNR of 41.2 dB and an SSIM of 0.995, representing very good perceptual image quality. The results shown in Fig. 8 validate the effectiveness of the proposed multiplier in preserving high-frequency image information and reducing computational complexity, while also showing the improved edge definition and the preservation of image structural details.

*Veera Boopathy E. et al.*

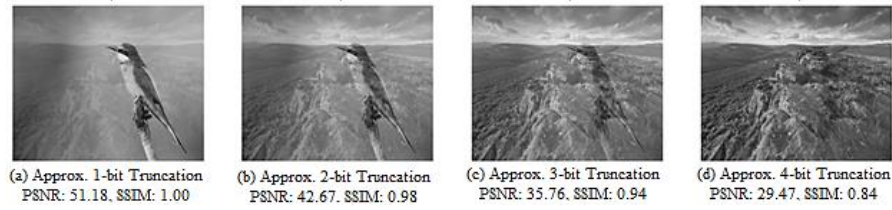


**Fig. 7.** Image with smoother transitions and quality images.

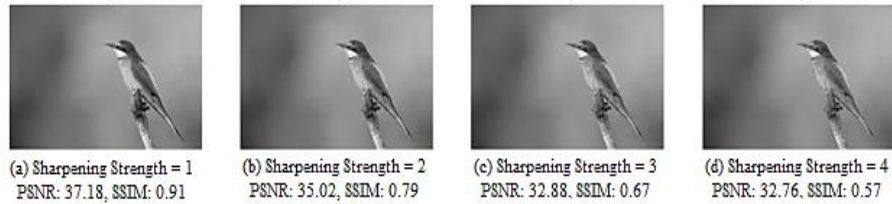


**Fig. 8.** Image with enhanced version, representing better clarity and edges.

Overall, the experimental results reveal that the proposed 22 nm FinFET-based approximate Modified Radix-8 Booth multiplier successfully combines algorithm-level optimization with advanced semiconductor technology to achieve substantial reductions in silicon area, propagation delay, power consumption, Power-Delay Product, and Area-Power Product while maintaining excellent application-level image quality. The architecture thus offers an efficient and practical solution for multiplication in high-performance digital signal processing, multimedia processing, artificial intelligence accelerators, and low-power embedded VLSI systems.

## V. Conclusion and Future Scope

This paper offered a 22 nm FinFET approximate Modified Radix-8 Booth multiplier which consists of Modified Radix-8 Booth recoding, selective partial-product approximation, controlled truncation, and an optimized partial-product reduction and accumulation network to enhance the multiplication efficiency in modern VLSI systems. The proposed architecture reduced silicon area by 63.4%, propagation delay by 55.1%, and power consumption by 65.6%, relative to an exact Booth multiplier, when implemented using a complete ASIC design flow. Moreover, the reduced Power-Delay Product (PDP) and Area-Power Product (APP) indicate a balance of speedup, energy efficiency, and hardware utilization. The application-level validation achieved a PSNR of 41.2 dB and an SSIM of 0.995, demonstrating high image quality even with the proposed approximation strategy while significantly reducing the hardware cost. Future research will include adaptive approximation techniques, extended bit-width multiplier architectures, inclusion in multiply-accumulate units and DSP hardware accelerators, and full FPGA and advanced-node ASIC implementations to improve scalability, energy efficiency, and practical application in future embedded computing architectures.

## Conflict of Interest:

There was no relevant conflict of interest regarding this paper.

*Veera Boopathy E. et al.*

## References

- I. Akbari, Omid, et al. "Dual-Quality 4:2 Compressors for Utilizing in Dynamic Accuracy Configurable Multipliers." *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 4, 2017, pp. 1352–1361. 10.1109/tvlsi.2016.2643003.
- II. Anchula, Sathish, et al. "Analysis of Approximate Multipliers Using FinFET." *2025 International Conference on Electronics, Computing, Communication and Control Technology (ICECCC)*, IEEE, 2025, pp. 1–7.
- III. Antelo, Elisardo, et al. "Improved 64-Bit Radix-16 Booth Multiplier Based on Partial Product Array Height Reduction." *IEEE Transactions on Circuits and Systems. I, Regular Papers*, vol. 64, no. 2, 2017, pp. 409–418. 10.1109/tcsi.2016.2561518.
- IV. Boopathy, E. Veera, and K. Kalirajan. "VLSI Implementation of Power-Aware Image Processing Using Hybrid Baugh-Booth Approximate Multiplier-Based MAC in Conviformer Architecture." *European Physical Journal Plus*, vol. 141, no. 9, 2026. 10.1140/epjp/s13360-026-08302-3.
- V. Chakraborty, Adrija, et al. "Energy Efficient Approximate Multiplier for Image Processing Applications." *Results in Engineering*, vol. 25, no. 103798, 2025, p. 103798. 10.1016/j.rineng.2024.103798.
- VI. Cui, Xiaoping, et al. "A Modified Partial Product Generator for Redundant Binary Multipliers." *IEEE Transactions on Computers. Institute of Electrical and Electronics Engineers*, vol. 65, no. 4, 2016, pp. 1165–1171. 10.1109/tc.2015.2441711.
- VII. George, Rachana, et al. "Fast Approximate Booth Multiplier for Error Resilient Applications." *International Journal of Signal and Imaging Systems Engineering*, vol. 13, no. 2, 2024, pp. 95–108. 10.1504/ijssise.2024.142338.
- VIII. Haq, Shams Ul, and Vijay Kumar Sharma. "Review of the Nanoscale FinFET Device for the Applications in Nano-Regime." *Current Nanoscience*, vol. 19, no. 5, 2023, pp. 651–662. 10.2174/1573413719666221206122301.
- IX. Kalaiselvi, C. M., and R. S. Sabeenian. "A Modular Technique of Booth Encoding and Vedic Multiplier for Low-Area and High-Speed Applications." *Scientific Reports*, vol. 13, no. 1, 2023, p. 22379. 10.1038/s41598-023-49913-5.
- X. Momeni, Amir, et al. "Design and Analysis of Approximate Compressors for Multiplication." *IEEE Transactions on Computers. Institute of Electrical and Electronics Engineers*, vol. 64, no. 4, 2015, pp. 984–994. 10.1109/tc.2014.2308214.
- XI. Pogue, Trevor E., and Nicola Nicolici. "Fast Inner-Product Algorithms and Architectures for Deep Neural Network Accelerators." *IEEE Transactions on Computers. Institute of Electrical and Electronics Engineers*, 2023, pp. 1–14. 10.1109/tc.2023.3334140.

*Veera Boopathy E. et al.*

- XII. Rao, E. Jagadeeswara, and P. Samundiswary. "High-Speed and Low-Power Recursive Rounding Based Approximate Multipliers for Error-Resilience Applications." *Wireless Personal Communications*, vol. 136, no. 2, 2024, pp. 773–791. 10.1007/s11277-024-11283-0.
- XIII. Sakthimohan, M., et al. "Enhanced FPGA Linear Phase FIR Filter with Amalgam Multiplier." *International Journal of Electronics*, vol. 111, no. 10, 2024, pp. 1653–1678. 10.1080/00207217.2024.2349972.
- XIV. Sayadi, Ladan, et al. "Two Efficient Approximate Unsigned Multipliers by Developing New Configuration for Approximate 4:2 Compressors." *IEEE Transactions on Circuits and Systems. I, Regular Papers*, vol. 70, no. 4, 2023, pp. 1649–1659. 10.1109/tcsi.2023.3242558.
- XV. Silvano, Cristina, et al. "A Survey on Deep Learning Hardware Accelerators for Heterogeneous HPC Platforms." *ACM Computing Surveys*, vol. 57, no. 11, 2025, pp. 1–39. 10.1145/3729215.