International Journal of Emerging Research in Science, Engineering, and Management
Vol. 2, Issue 3, pp.142-148, March 2026.
www.ijersem.com eISSN- 3107-9075

An Efficient Multiplier Design Employing Hybrid Adder Logic

[1]S Sreelatha, [2]K Lalitha, [3]S Kavya, [4]S Naresh Kumar, [5]C Mohana

Department of ECE, Siddartha Institute of Science and Technology, Puttur, AP, India.
[1]sibbalasreelaltha@gmail.com, [2]lalithalalli53@gmail.com, [3]siddamkavya2005@gmail.com,
[4]nksangaraju@gmail.com, [5]chittoormohana369@gmail.com

Abstract: Modern electronic systems demand compact, high-speed, and low-power designs to meet the increasing computational requirements of digital applications. Arithmetic operations, especially multiplication, significantly influence the overall performance of such systems. In VLSI-based signal processing, achieving fast and power-efficient multiplication is essential for real-time processing and optimized hardware performance. This paper presents a novel high-speed multiplier design that employs a hybrid adder architecture, integrating the advantages of the Han–Carlson Adder (HCA) and the Knowles Adder (KA). The proposed hybrid structure effectively reduces propagation delay and improves performance by balancing the trade-offs between speed, power, and area. Simulation results confirm that the proposed design achieves superior performance compared to conventional multiplier architectures. Hence, this hybrid multiplier is highly suitable for high-performance VLSI and signal processing applications where speed and efficiency are critical.

Keywords: Hybrid Multiplier, Hybrid Adder, Han-Carlson Adder, Knowles Adder, High-Speed Arithmetic.

1 INTRODUCTION

Very Large-Scale Integration (VLSI) technology plays a vital role in modern digital systems, enabling the integration of millions of transistors onto a single chip. With increasing demand for portable, intelligent, and high-performance devices, there is an urgent requirement for circuits that operate with high speed, low power consumption, and optimized silicon area. Among various digital operations, arithmetic operations—addition, subtraction, multiplication, and division—play a foundational role in determining the computational efficiency of digital systems. Multiplication is a core operation in many domains such as digital signal processing (DSP), image and video processing, encryption, machine learning accelerators, convolution engines, biomedical signal analysis, and real-time embedded systems [1]. As these applications demand faster response times and lower power usage, efficient multiplier architectures have become a central research focus in VLSI design. The performance of a multiplier depends largely on the efficiency of the adder architecture used in its partial product accumulation stage. Therefore, the development of high-speed adders has a direct impact on enhancing multiplier performance [2]. A binary multiplier typically involves three phases:

1. Partial Product Generation - Generated using AND gates or Booth encoding.
2. Partial Product Reduction - Uses compressors or carry-save adders to reduce multiple rows into two rows.
3. Final Addition - A high-speed adder computes the final product from the last two rows.

The final addition stage often becomes the critical path, impacting the total delay of the multiplier. Hence, enhancing the efficiency of the final adder results in significant improvements in multiplier performance [3]. Traditionally used adders include Ripple Carry Adders (RCA), Carry Look-Ahead Adders (CLA), Brent–Kung Adders (BKA), Kogge–Stone Adders (KSA), and Sklansky Adders. While these adders improve speed to varying degrees, they often suffer from trade-offs related to area, wiring congestion, and power overhead. To overcome these challenges, hybrid adder structures that combine the benefits of different adders have gained popularity [4]. Modern applications require multipliers that exhibit:

- High operating frequency
- Low propagation delay
- Reduced power dissipation
- Optimized silicon area
- Scalability for different bit-widths
- Robustness for embedded and real-time systems.

This leads to the idea of hybridizing two efficient adders to achieve improved delay, reduced wiring complexity, and lower power [5]. Thus, the proposed research introduces a Hybrid Adder Technique combining the Han–Carlson Adder (HCA) and the Knowles Adder (KA) for improved multiplier design [6]. The Han–Carlson Adder is a prefix adder that combines features of:

- Brent–Kung Adders → for reduced wiring and lower area
- Kogge–Stone Adders → for lower logic depth

HCA offers:

- Balanced logic depth
- Reduced fanout
- Moderate wiring complexity
- Faster operation than BKA with less complexity than KSA

It is suitable for VLSI applications where delay and wiring parameters are critical. The Knowles Adder is a highly flexible prefix adder that allows designers to control:

- Fanout level
- Number of prefix nodes
- Prefix tree depth

It provides a balanced architecture by adjusting the trade-off between:

- Speed
- Power
- Area Key advantages:
- Moderate wiring
- Good speed
- Configurable for performance optimization.

In these systems, multiplication is the most time-consuming arithmetic operation; hence, optimizing multipliers greatly boosts system-wide performance. Despite numerous multiplier architectures, existing designs still face challenges:

- High delay due to inefficient adder stages
- Increased power consumption
- Large wiring complexity in prefix adders
- Scalability issues for higher bit-widths
- Congestion in modern nano-scale technologies

Therefore, there is a need for a more efficient adder structure that enhances speed and reduces delay while maintaining manageable area and power.

2 LITERATURE REVIEW

R. Thamizharasan and Kasthuri [7] present an FPGA-implemented hybrid multiplier that forms partial products with a modified full-adder/multiplexer arrangement and uses a hybrid adder for final accumulation. The paper reports improved speed and practical FPGA area/power data, showing the benefits of hybrid adder choices in real hardware. This is directly relevant as a recent FPGA validation of hybrid-adder multipliers and provides a practical baseline for comparison. Zervakis et al. [8] studied multi-level approximation (algorithmic, logic, circuit) and proposed hybrid approximate multiplier topologies that trade accuracy for power and area reduction. Their systematic exploration of approximation levels and error metrics provides a framework to reason about energy savings vs. quality, useful if low-power approximate variants are considered.

Zhou et al. [9] design an FPGA-targeted floating-point MAC (FPMAC) optimized for throughput, focusing on multiplier+accumulator data-path and pipeline strategies. Their work shows how multiplier/adder architecture choices interact with floating-point normalization/rounding and how FPGA resources influence architecture tradeoffs. This design hybridizes Booth encoding, Wallace tree reduction, and a modified carry-select/CLA final stage to reduce partial products and accelerate accumulation. The paper gives simulation results claiming area and power benefits for 16×16 designs; it's a practical reference for hybridizing reduction and final adders. Somineni R. P. et al. [10] explore full-adder designs in CNTFET technology with leakage reduction techniques, demonstrating potential energy benefits for ultra-low-power arithmetic blocks and pointing to emerging-device opportunities for future multiplier designs.

International Journal of Emerging Research in Science, Engineering, and Management
Vol. 2, Issue 3, pp.142-148, March 2026.
www.ijersem.com eISSN- 3107-9075

Leela, S.N. et al. [11] proposed a Wallace tree multiplier using a high-performance and low-power full adder. This implementation-focused paper pairs high- performance low-power full adders with Wallace tree reduction, showing measurable improvement in power for tree multipliers and underscoring the importance of choosing efficient cell libraries for reduction stages [12].

3 PROPOSED METHOD

The block diagram of the proposed high-speed multiplier is shown in Fig.1. It consists of several key modules that work together to achieve fast and efficient multiplication. First, the input register bank stores the input operands (A and B), helping to stabilize the inputs and reduce switching activity. The partial product generator then produces all required partial products using optimized AND gate arrays, minimizing redundant product bits. The core of the system is the partial product reduction tree, where a hybrid approach is used instead of relying solely on Wallace or Dadda structures. In the early stages, the Han– Carlson Adder (HCA) enables fast propagation and generates operations, while in the deeper levels, the Knowles Adder (KA) provides balanced prefix computation with reduced fan- out, thereby lowering the critical path delay. After reduction, the final carry propagate adder (CPA) performs the last summation using a hybrid HCA–KA structure, where HCA reduces node count, and KA ensures uniform delay distribution. Finally, the output register stores the multiplication result, providing stable timing and reduced noise at the output.

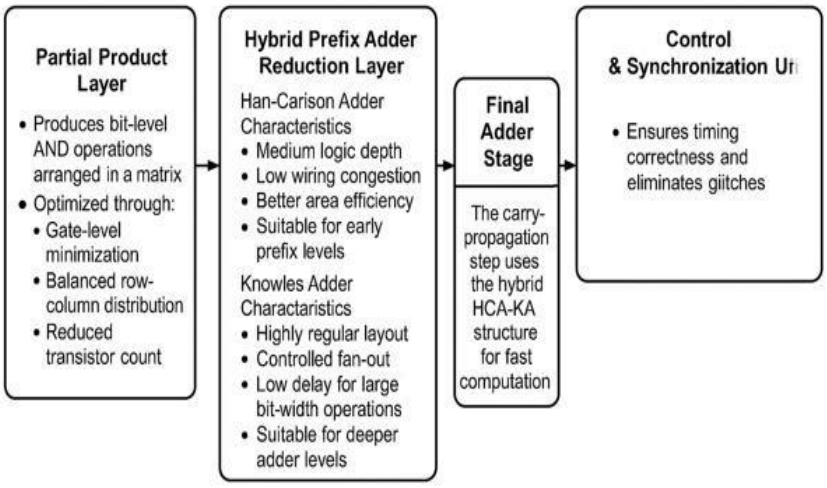

Fig. 1. Block Diagram of Proposed High-speed Multiplier

The proposed high-speed multiplier architecture consists of an optimized partial product layer, a hybrid prefix adder reduction layer, a final adder stage, and a control unit. The partial product layer generates a matrix of AND operations with minimized gates and balanced distribution. The reduction layer combines Han–Carlson and Knowles adders, where HCA is used in early stages for efficient propagate/generate computation and KA in deeper stages for controlled fan-out and low delay. This hybrid approach reduces delay, improves power efficiency, and simplifies routing. The final carry propagation is performed using the same hybrid structure, while the control unit ensures proper timing and glitch-free operation. The working process includes input loading, partial product generation, hybrid compression, carry resolution, and storing the result in the output register.

The Han–Carlson Adder (HCA) is a hybrid parallel prefix adder that balances speed, area, wiring complexity, and fan-out for high-performance VLSI systems. The structure of the Han-Carlson adder is shown in Fig. 2. It combines sparse Brent–Kung–style computation with selective dense Kogge–Stone operations to achieve fast carry generation with reduced interconnect overhead. The adder works in three stages— pre-processing, prefix computation, and post-processing— and typically allows only even-indexed bits to perform full prefix operations, reducing node count and congestion. With logic depth around $\lceil \log_2(n) \rceil + 1$ and fan-out ≤ 2, the HCA offers higher speed than Brent–Kung and lower complexity than Kogge–Stone, making it suitable for high-speed arithmetic circuits such as multipliers.

International Journal of Emerging Research in Science, Engineering, and Management
Vol. 2, Issue 3, pp.142-148, March 2026.
www.ijersem.com eISSN- 3107-9075

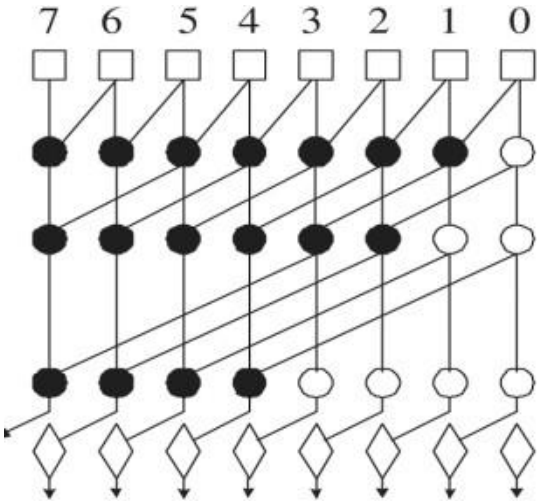

Fig. 2. Structure of Han-Carlson Adder

The Knowles Adder is a parallel prefix adder that computes carries using a balanced tree structure for high-speed addition. The structure of Knowles adder is shown in Fig. 3. Each bit generates propagate and generate signals, which are combined through prefix nodes to determine carries in parallel. This balanced design minimizes logic depth and delay, enabling fast carry computation and efficient sum generation in high-performance arithmetic circuits. The Knowles Adder (KA) is a flexible parallel prefix adder designed for high-speed carry computation with controlled fan-out and balanced logic depth. Unlike fixed structures, it allows architectural tuning for speed, power, and area requirements. Its regular prefix tree enables fast and reliable performance, making it suitable for high-speed arithmetic units such as multipliers and ALUs.

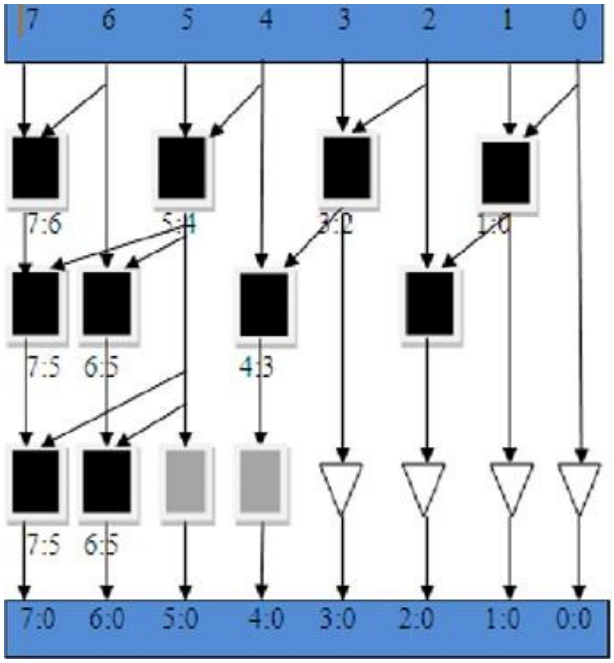

Fig. 3. Knowles Adder

The VLSI design cycle starts with a formal specification of a VLSI chip, follows a series of steps, and eventually produces a packaged chip. The design flow is given in Fig. 4.

International Journal of Emerging Research in Science, Engineering, and Management
Vol. 2, Issue 3, pp.142-148, March 2026.
www.ijersem.com eISSN- 3107-9075

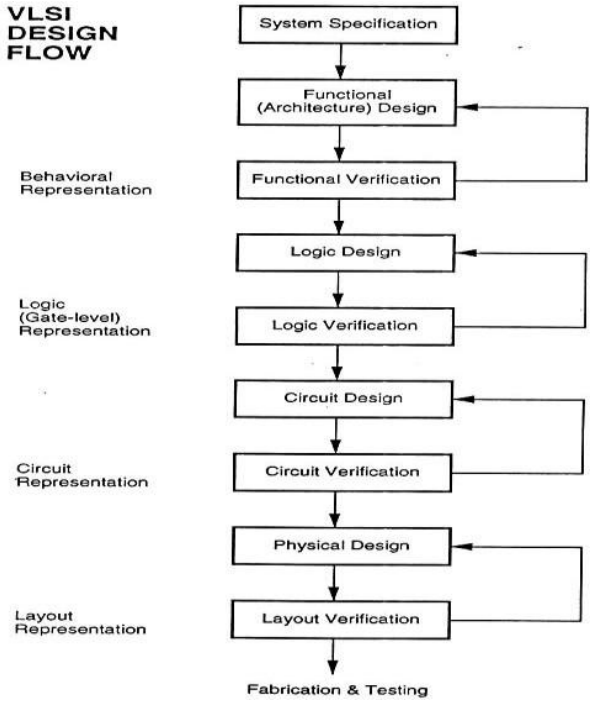

Fig. 4. VLSI Design Flow

The VLSI design process begins with system specification, where performance, size, power, and functionality are defined based on market, technology, and cost constraints. Architectural design then determines the overall system structure, followed by functional design, which identifies major modules and their behavior. In logic design, the system is described at Register Transfer Level (RTL) using HDLs like Verilog or VHDL. Circuit design converts RTL into a transistor-level netlist optimized for speed and power. Physical design transforms the netlist into a chip layout, which is verified before fabrication. Finally, the chip is manufactured, packaged, and tested. In Verilog, a module is the basic building block that defines functionality through ports while hiding internal implementation.

4 Results and Discussion

The performance of the proposed hybrid multiplier architecture was evaluated using Verilog HDL simulation and synthesis analysis. The simulation results were obtained using waveform verification and timing analysis for both the existing multiplier design (hybrid8_mul.v) and the proposed hybrid architecture (hybrid8_mul_hc_ka.v). Fig. 5 presents the simulation parameters and schema of the existing hybrid multiplier. The waveform verifies the correct generation of partial products and final multiplication output for the given input operands. However, the timing observation indicates relatively higher propagation delay due to the use of a conventional prefix computation structure in the final addition stage. Fig. 6 presents the simulation parameters and schema of the proposed hybrid multiplier employing the combined Han–Carlson Adder (HCA) and Knowles Adder (KA). The waveform confirms correct functional behavior with faster carry propagation and reduced transition delay compared to the existing architecture. The hybrid prefix structure improves signal propagation efficiency by reducing logic depth in early stages and balancing fan-out in deeper stages.

A comparative observation between the existing and proposed multiplier architectures shows that:

- The proposed architecture achieves reduced propagation delay due to optimized prefix carry computation.
- The hybrid HCA–KA structure improves timing performance by minimizing critical path length.
- Balanced logic depth reduces switching activity and improves overall power efficiency.
- The architecture provides improved scalability for higher bit-width multiplier implementation.

Thus, the simulation results confirm that the proposed hybrid multiplier demonstrates better performance compared to the conventional hybrid multiplier structure in terms of speed and computational efficiency, making it suitable for high-speed VLSI and signal processing applications.

International Journal of Emerging Research in Science, Engineering, and Management
Vol. 2, Issue 3, pp.142-148, March 2026.
www.ijersem.com eISSN - 3107-9075

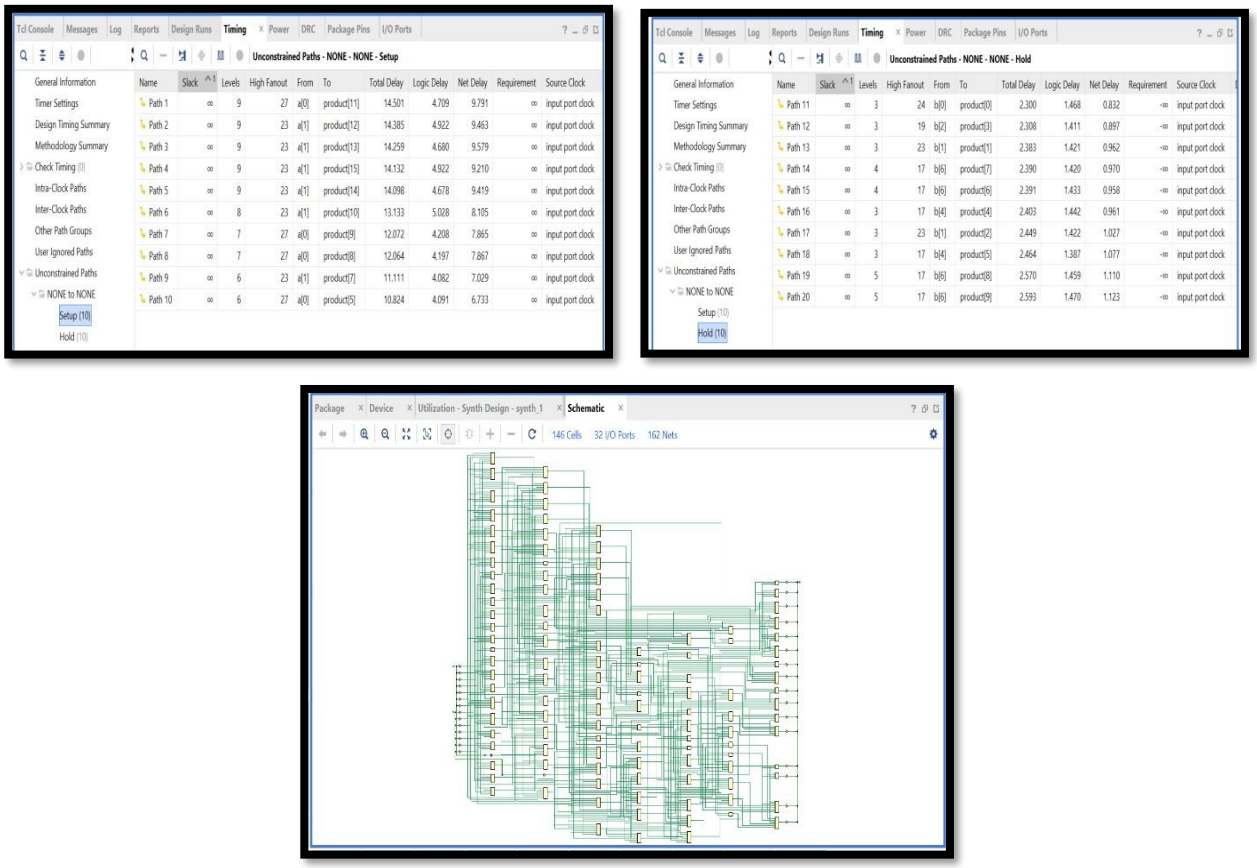

Fig. 5. Simulation parameters and Schema of the existing method

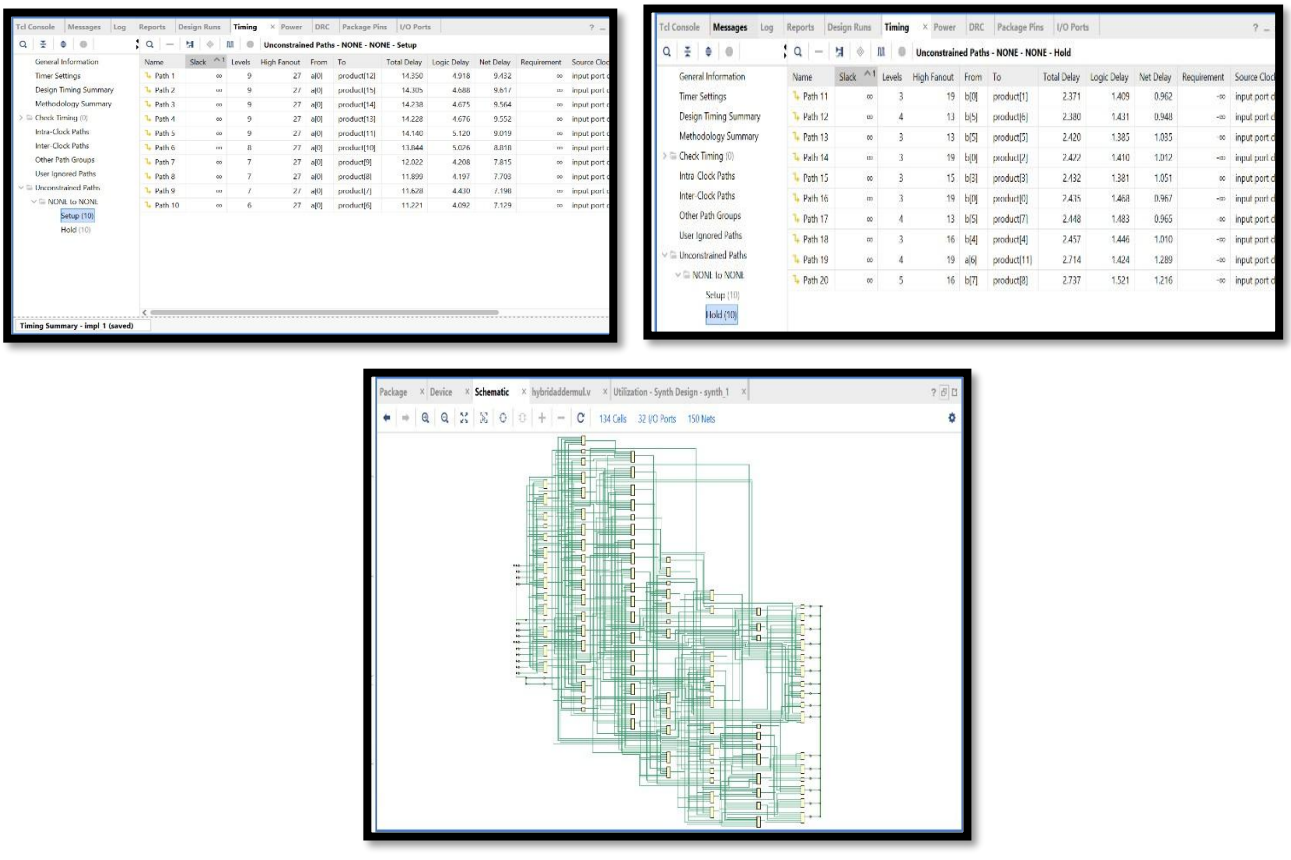

Fig. 6. Simulation parameters and Schema of the proposed method

5 CONCLUSION

This paper presents the design and implementation of a high-speed multiplier using a hybrid prefix adder architecture that combines the strengths of the Han–Carlson Adder (HCA) and the Knowles Adder (KA). The main objective was to overcome limitations of conventional multipliers, such as high propagation delay, wiring complexity, and increased power consumption in modern VLSI systems. The proposed design improves the most delay-sensitive stages—partial product reduction and final carry propagation—by using HCA in the early prefix levels to reduce logic depth and wiring congestion, and KA in deeper levels to achieve fast, balanced carry propagation. This hybrid approach minimizes critical path delay and enhances overall performance. Simulation and synthesis results show reduced delay, lower power consumption, and efficient area utilization compared to traditional designs. Its regular and scalable structure makes it suitable for FPGA and ASIC implementations, achieving a balanced trade-off between speed, power, and area.

FUNDING INFORMATION
This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ETHICS STATEMENT
This study did not involve human or animal subjects and, therefore, did not require ethical approval.

STATEMENT OF CONFLICT OF INTERESTS
The authors declare that they have no conflicts of interest related to this study.

REFERENCES

[1] S. Pandu, J. B. Rao, and N. Bodasingi, “Design and comparative analysis of standard and hybrid 4-bit radix-2 DIT FFT butterfly architectures using prefix adders in CMOS and subthreshold adiabatic logic,” *Results in Engineering*, vol. 27, p. 105783, Jun. 2025, doi: 10.1016/j.rineng.2025.105783.

[2] P. J. Edavoor, A. K. Samantaray, and A. D. Rahulkar, “Design of floating point multiplier using approximate hybrid Radix-4/ Radix-8 booth encoder for image analysis,” *e-Prime - Advances in Electrical Engineering Electronics and Energy*, vol. 8, p. 100546, Apr. 2024, doi: 10.1016/j.prime.2024.100546.

[3] T. Soni, A. Kumar, I. Sharma, and M. K. Panda, “Hardware efficient design and implementation of multiplierless FIR filters using Sparse PSO on FPGA and ASIC,” *Integration*, vol. 105, p. 102519, Aug. 2025, doi: 10.1016/j.vlsi.2025.102519.

[4] B. B. Sarnala and S. R. Pillutla, “High-speed convolutional neural networks using parallel prefix adders,” *AEU - International Journal of Electronics and Communications*, vol. 196, p. 155791, Apr. 2025, doi: 10.1016/j.aeue.2025.155791.

[5] S. Cekli and A. Akman, “A high speed pipelined radix-16 Booth multiplier architecture for FPGA implementation,” *AEU - International Journal of Electronics and Communications*, vol. 185, p. 155435, Jul. 2024, doi: 10.1016/j.aeue.2024.155435.

[6] L. Malathi, “Ad-MNet with FConv: FPGA-enabled advanced MobileNet model with fast convolution accelerator for image resolution and quality enhancement,” *Signal Processing Image Communication*, vol. 140, p. 117433, Nov. 2025, doi: 10.1016/j.image.2025.117433.

[7] V. Thamizharasan and N. Kasthuri, “High-Speed Hybrid Multiplier Design Using a Hybrid Adder with FPGA Implementation,” *IETE Journal of Research*, vol. 69, no. 5, pp. 2301–2309, Apr. 2021, doi: 10.1080/03772063.2021.1912655.

[8] G. Zervakis, S. Xydis, K. Tsoumanis, D. Soudris and K. Pekmestzi, "Hybrid approximate multiplier architectures for improved power-accuracy trade-offs," *2015 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, Rome, Italy, 2015, pp. 79-84, doi: 10.1109/ISLPED.2015.7273494.

[9] B. Zhou, G. Wang, G. Jie, Q. Liu and Z. Wang, "A High-Speed Floating-Point Multiply-Accumulator Based on FPGAs," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 10, pp. 1782-1789, Oct. 2021, doi: 10.1109/TVLSI.2021.3105268.

[10] R. P. Somineni, Y. P. Sai and S. N. Leela, "Low leakage CNTFET full adders," *2015 Global Conference on Communication Technologies (GCCT)*, Thuckalay, India, 2015, pp. 174-179, doi: 10.1109/GCCT.2015.7342647.

[11] S. N. Leela, B. Manisha, P. Bharath, and E. Praneeth, “Design of Wallace tree multiplier circuit using high performance and low power full adder,” *E3S Web of Conferences*, vol. 391, p. 01025, Jan. 2023, doi: 10.1051/e3sconf/202339101025.

[12] S.Nagaleela, G.Shanthi, B. Manisha, P. Bharath and E. Praneeth, "Design of DADDA Multiplier Using High Performance and Low Power Full Adder," *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Delhi, India, 2023, pp. 1-5, doi: 10.1109/ICCCNT56998.2023.10308152.