# IMPLEMENTATION AND TESTING OF A 32-BIT MULTIPLIER FOR CUTTING-EDGE VLSI CIRCUITS

**[1]Boppudi Adithya, [2]Angalakurthi Stanli Johns, [3]Borugadda Koushik, [4]Biruduraju Venkata Sai Chaitanya Raju, [5]Edulamudi Sirisha**

[1-4]UG Students, [5]Associate Professor, Department of Electronics and Communication Engineering,
Vasireddy Venkatadri Institute of Technology, Nambur, Andhra Pradesh, India

***Abstract:*** As digital systems continue to scale in complexity, conventional multipliers face significant challenges related to power dissipation, propagation delay and area. Reversible logic offers a promising solution by preserving information and minimizing energy loss during computation , thereby improving the overall efficiency of arithmetic circuits. This paper presents the design and implementation of a 32-bit multiplier using reversible logic, employing HNG and Peres reversible gates. The proposed architecture follows a modular and hierarchical design approach, starting from a 4-bit multiplier block and extending to a complete 32-bit structure. The design is implemented and verified using Xilinx Vivado, with performance evaluated in terms of propagation delay, power consumption and resource utilization. Furthermore, the reliability of the design is further assessed through stuck-at fault analysis, considering both stuck-at-0 and stuck-at-1 conditions at critical nodes. Experimental results demonstrate that the proposed multiplier design achieves improved speed and reduced power consumption compared to conventional multiplier architectures. These findings highlight the potential of reversible logic in developing high-performance and energy-efficient hardware suitable for next-generation VLSI systems.

***Keywords:*** **Reversible Logic, 32-bit Multiplier, Power, Delay, Resource Utilization, Stuck-at-Fault Model, Verilog HDL.**

## I.INTRODUCTION

Multiplication is a fundamental operation in digital systems and has a direct impact on the performance of modern computing devices. It is widely used in applications such as digital signal processing, communication systems, and microprocessors. Since multiplication influences processing speed, power consumption, and hardware utilization, designing an efficient multiplier is an important challenge in VLSI system design. Conventional multiplier architectures improve performance by increasing parallelism, but they often face limitations such as higher power consumption, increased propagation delay, and complex interconnections, especially in higher bit-width designs like 32-bit multipliers. These challenges highlight the need for alternative approaches that can provide better efficiency without significantly increasing design complexity. Reversible logic has emerged as a promising approach for designing low-power digital circuits. Unlike conventional logic, reversible circuits aim to reduce energy dissipation by preserving information through a one-to-one mapping between inputs and outputs. This characteristic makes them suitable for future energy-efficient computing systems. However, designing optimized reversible circuits while maintaining performance remains a challenge. In this work, a 32-bit multiplier based on reversible logic is proposed to improve power efficiency and computational performance. The design is implemented using optimized reversible gates and a scalable architecture. In addition, fault analysis is performed using the stuck-at fault model to evaluate the reliability of the design. The proposed approach aims to achieve a balanced trade-off between speed, power consumption, and design complexity.

## II. LITERATURE REVIEW

Several multiplier architectures have been proposed to improve the performance of digital systems. C. S. Wallace introduced the Wallace Tree multiplier to minimize propagation delay through parallel reduction. While this significantly boosts speed, the resulting structure is irregular, leading to complex routing and hardware overhead. In contrast, Array multipliers are favored for their simple, regular layouts, though they quickly become inefficient in terms of delay and power as bit-widths increase. To bridge this gap, the Booth algorithm was developed by A. D. Booth to reduce the total number of partial products. While modified Booth architectures improve computational efficiency, they often introduce complex control logic that offsets some of the power gains. These conventional multiplier architectures, although capable of achieving high computational speed, often suffer from increased power consumption and design complexity. This creates a need for an optimized design approach that can effectively balance performance and efficiency. In recent years, reversible logic has gained significant attention as a promising approach for low-power VLSI design. Rolf Landauer demonstrated that information loss in irreversible circuits leads to energy dissipation, which led to the concept of reversible computing. Later, Charles H. Bennett showed that reversible computation can theoretically eliminate energy loss caused by information erasure. Based on these principles, reversible logic has been widely explored for designing energy-efficient arithmetic circuits. In this work, a 32-bit multiplier based on reversible logic is proposed to address these challenges. The proposed approach aims to achieve a balanced trade-off between speed, power consumption, and design complexity, making it suitable for modern high-performance VLSI applications.

Table 1: Comparison of Conventional Multipliers

| **Parameter** | **Array Multiplier** | **Booth Multiplier** | **Wallace Tree Multiplier** |
|---|---|---|---|
| Power Consumption | High | Medium | Medium |
| Time Delay | High | Medium | Low |
| Area | High | Medium | High |
| Complexity | Simple | Moderate | Complex |

The comparison highlights that conventional multipliers either suffer from higher power consumption, increased delay, or greater design complexity. While each architecture improves specific parameters, achieving an optimal balance among these factors remains a challenge. This motivates the need for an alternative design approach, which is addressed in this work using reversible logic.

## III. PROPOSED WORK

### 1. REVERSIBLE LOGIC

Reversible logic is an emerging design approach in VLSI systems that aims to reduce power dissipation by avoiding information loss during computation. In conventional logic circuits, loss of information results in energy dissipation in the form of heat. Reversible logic overcomes this limitation by ensuring a one-to-one mapping between inputs and outputs, allowing the original inputs to be uniquely reconstructed from the outputs. A reversible circuit is characterized by having an equal number of inputs and outputs, with no fan-out and no feedback paths. Important parameters in reversible logic design include garbage outputs, which are unwanted outputs required to maintain reversibility, and constant inputs, which are fixed input values added to realize specific logic functions. Minimizing these parameters is essential for improving the efficiency of reversible circuits. Due to these properties, reversible logic is considered a promising solution for low-power and energy-efficient digital system design.

### 2. REVERSIBLE GATES

In this work, reversible gates such as the Peres gate and HNG gate are utilized to construct the multiplier architecture.

**a) Peres Gate:**

The Peres Gate (PG) is a widely used 3×3 reversible logic gate known for its low quantum cost of 4 and efficient implementation of arithmetic functions. Due to this functionality, the Peres gate can perform both XOR and AND operations simultaneously, making it suitable for arithmetic circuit design. Additionally, by fixing input C = 0, it can directly generate XOR and AND outputs.

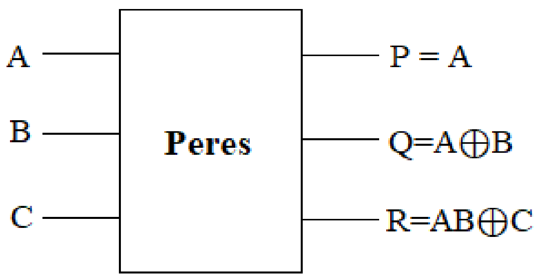


Figure 1: Peres Gate

**b) HNG Gate:**

The HNG gate is a 4×4 reversible logic gate widely used in the design of efficient arithmetic circuits, with a quantum cost of 6. It maps the input vector (A, B, C, D) to the output vector (P, Q, R, S) and is capable of implementing full adder functionality within a single gate. The sum output is given by $R = A \oplus B \oplus C$, the carry output is generated as a function of inputs A, B, and C by taking the input D set to 0 to maintain reversibility.

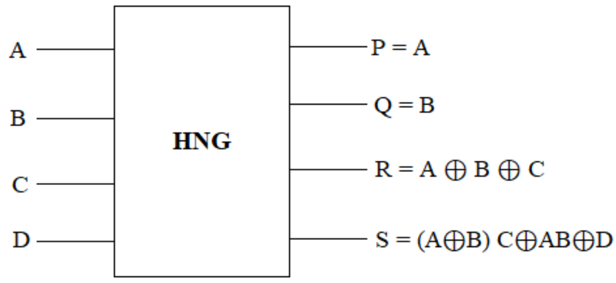


Figure 2: HNG Gate

## 3. STRUCTURAL DESIGN APPROACH

The proposed multiplier is developed using a hierarchical and modular design approach to achieve scalability and efficient implementation. Initially, a 4-bit multiplier is designed using Peres and HNG reversible gates, which serves as the fundamental building block for higher-bit multipliers. To support efficient addition of partial products, Carry Look Ahead (CLA) adders are implemented using HNG gates. A 4-bit CLA adder is first designed and then extended to 8-bit and 16-bit adders using a structured approach. The 8-bit multiplier is constructed by combining multiple 4-bit multiplier modules, where partial products are generated and summed using CLA adders. This methodology is further extended to design 16-bit and 32-bit multipliers by hierarchically integrating lower-bit modules.

## IV. PROPOSED 32-BIT MULTIPLIER ARCHITECTURE

The proposed 32-bit multiplier is designed using 16-bit multipliers. Proper bit positioning ensures accurate alignment and summation of intermediate results, enabling correct final output generation.

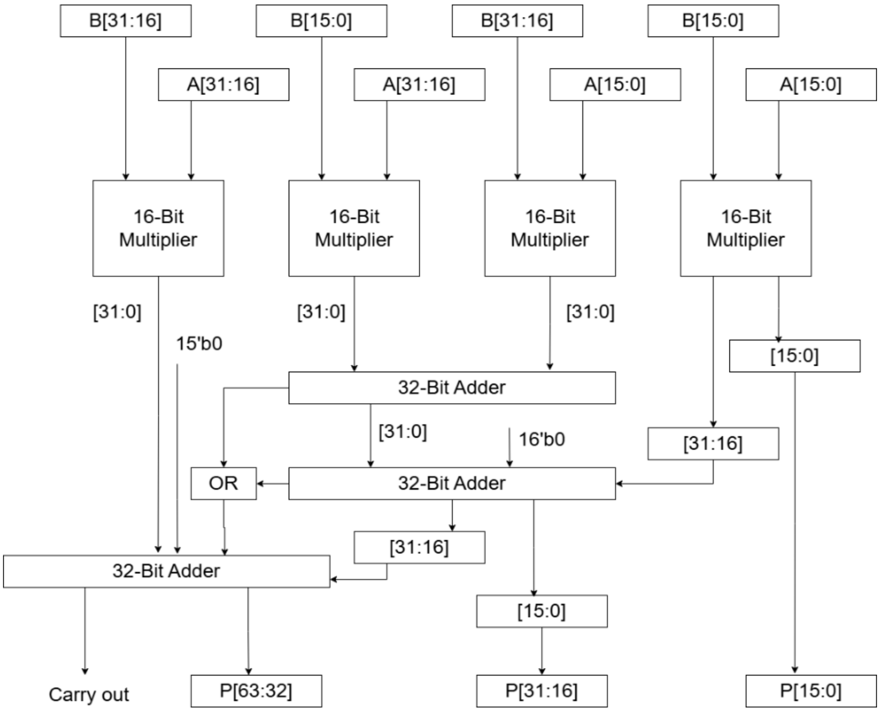


Figure 3: 32-Bit Multiplier Architecture

## V. SIMULATION AND RESULTS

The proposed 32-bit multiplier is designed using a hierarchical approach by combining lower-bit multiplier modules. Multiple 16-bit multiplier blocks are interconnected to construct the complete 32-bit architecture. Partial products are generated in parallel and properly aligned based on their bit positions to ensure accurate computation. These partial products are then efficiently summed using Carry Look Ahead (CLA), which help reduce carry propagation delay. Proper bit positioning ensures accurate alignment and summation of intermediate results, leading to correct final output generation.

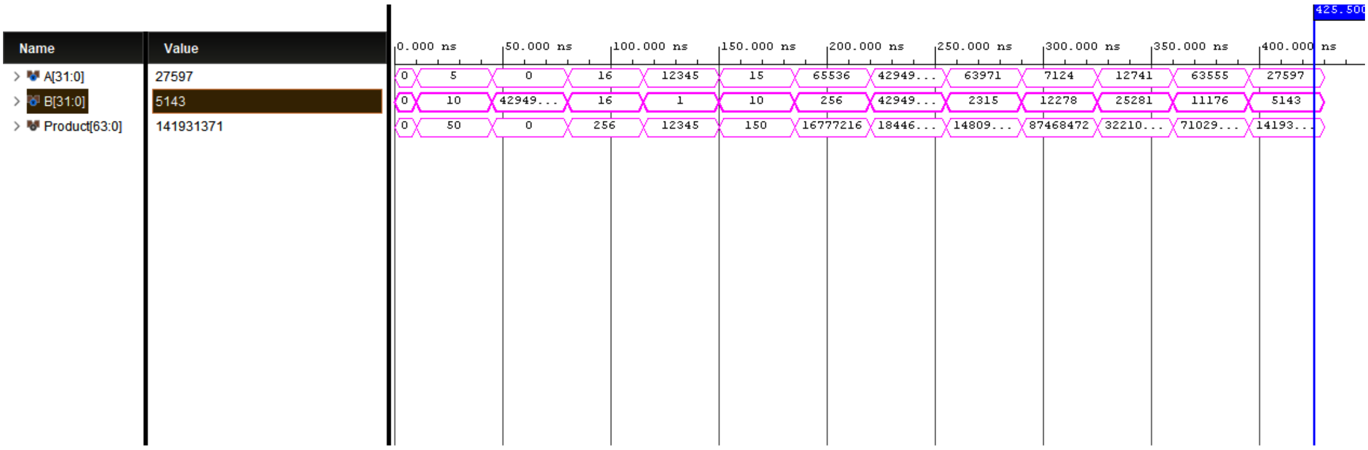


Figure 4: Simulation waveform of 32-bit multiplier

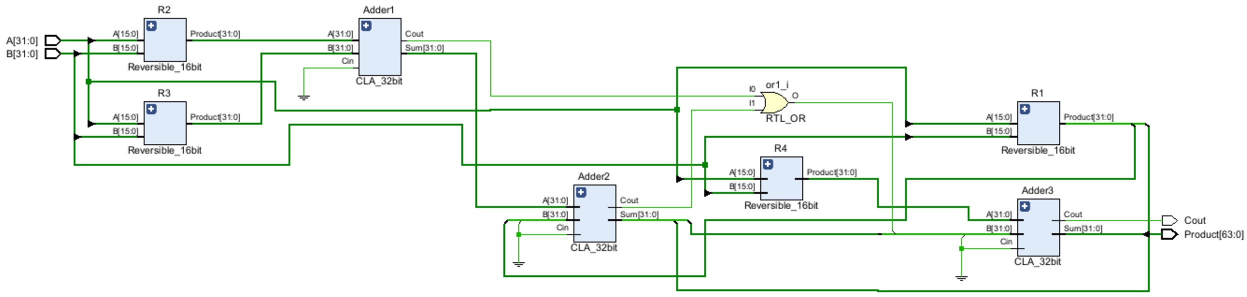


Figure 5: Schematic of 32-bit multiplier

The performance of the proposed design is evaluated using reports generated by the Vivado design tool. The key parameters considered for analysis include power consumption, propagation delay, and area utilization. These parameters are essential in determining the efficiency and practicality of the design for VLSI applications.

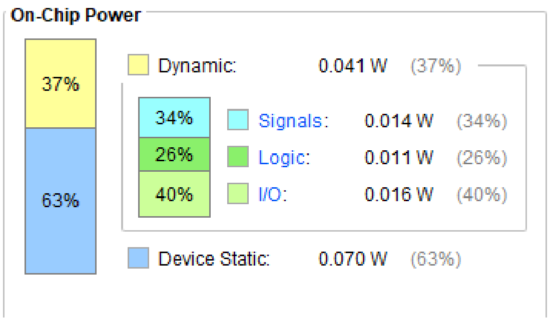


Figure 6: Power

The power analysis of the proposed design shows a total on-chip power consumption of 0.111 W, with dynamic power of 0.041 W

and static(leakage) power of 0.070 W.

| Name | Total Delay | Logic Delay | Net Delay |
|---|---|---|---|
| Path 1 | 33.784 | 8.570 | 25.214 |
| Path 2 | 33.724 | 8.485 | 25.239 |
| Path 3 | 33.650 | 8.464 | 25.186 |
| Path 4 | 33.460 | 8.477 | 24.983 |
| Path 5 | 33.317 | 8.342 | 24.975 |
| Path 6 | 33.295 | 8.473 | 24.822 |
| Path 7 | 33.291 | 8.478 | 24.813 |
| Path 8 | 33.268 | 8.332 | 24.936 |
| Path 9 | 33.256 | 8.471 | 24.785 |
| Path 10 | 33.074 | 8.285 | 24.789 |

Figure 7: Total delay

The critical path delay of the proposed design is observed to be approximately 33.784 ns, which represents the maximum delay among all the paths in the circuit. A virtual clock of 35 ns is applied 37 during timing analysis.

| Name | Slice LUTs (20800) | Slice (8150) | LUT as Logic (20800) | Bonded IOB (170) |
|---|---|---|---|---|
| Reversible_32bit | 2036 | 572 | 2036 | 129 |
| Adder3 (CLA_32bit_0) | 993 | 369 | 993 | 0 |
| R1 (Reversible_16bit) | 227 | 84 | 227 | 0 |
| R2 (Reversible_16bit_1) | 343 | 121 | 343 | 0 |
| R3 (Reversible_16bit_2) | 235 | 89 | 235 | 0 |
| R4 (Reversible_16bit_3) | 238 | 93 | 238 | 0 |

Figure 8: Resource Utilization

The design utilizes a total of 2036 Look-Up Tables (LUTs) and 572 slices. The resource distribution shows that the major portion of the area is consumed by the 32-bit Adders and intermediate reversible multiplier modules.

## VI. FAULT TESTING AND ANALYSIS

To evaluate the reliability of the proposed 32-bit multiplier, fault testing is performed using the stuck-at fault model. In this model, a signal line is assumed to be permanently fixed at logic ‘0’ (stuck-at-0) or logic ‘1’ (stuck-at-1), which may occur due to manufacturing defects or hardware failures. Faults are intentionally introduced at selected internal nodes of the design, including key inputs and intermediate signals. The circuit is then simulated under these fault conditions, and the obtained outputs are compared with the expected results. Any mismatch between the outputs indicates the presence of a fault, allowing effective detection of errors in the design.

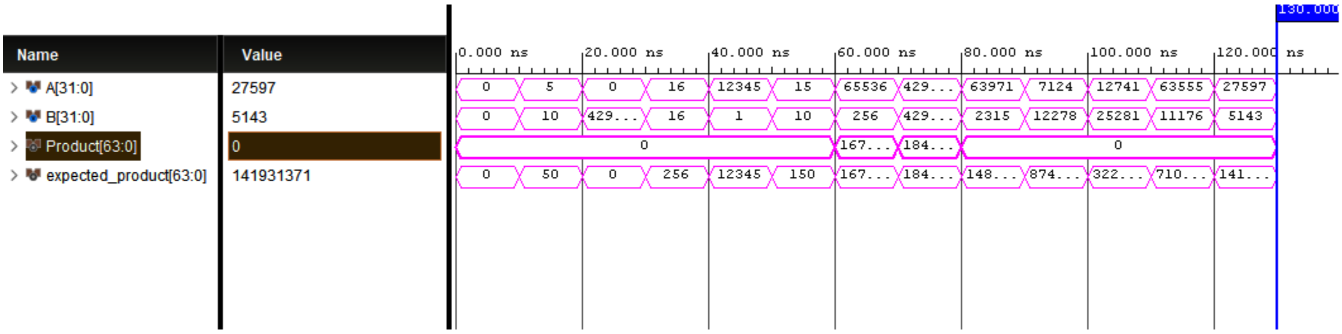


Figure 9: Stuck-at-0 Fault Simulation Waveform

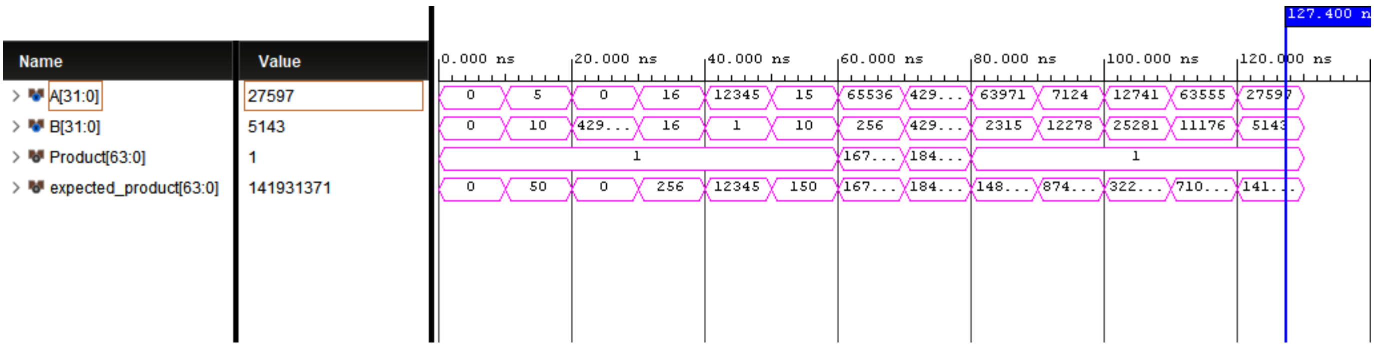


Figure 10: Stuck-at-1 Fault Simulation Waveform

## VII. CONCLUSION

The simulation results of the proposed 32-bit reversible multiplier demonstrate efficient performance in terms of power, delay, and area. The total on-chip power consumption is observed to be 0.111 W, indicating reduced energy dissipation. The propagation delay is approximately 33 ns, showing improved computational speed. From the utilization summary, the design occupies 2036 LUTs and 572 slices, reflecting a moderate hardware requirement. Fault testing using the stuck-at fault model confirms reliable operation under fault conditions. Overall, the proposed design achieves a balanced trade-off between speed, power efficiency, and complexity, making it suitable for high-performance VLSI applications.

## VIII. REFERENCES

[1] C. S. Wallace, “A Suggestion for a Fast Multiplier,” *IEEE Transactions on Electronic Computers*, vol. EC-13, no. 1, pp. 14–17, Feb. 1964.

[2] R. Landauer, “Irreversibility and Heat Generation in the Computing Process,” *IBM Journal of Research and Development*, vol. 5, no. 3, pp. 183–191, 1961.

[3] H. Thapliyal and M. B. Srinivas, “A Novel Reversible TSG Gate and Its Application for Designing Reversible Carry Look-Ahead Adder and Other Adder Architectures,” in *Proc. IEEE Computer Society Annual Symposium on VLSI*, 2005, pp. 805–810.

[4] H. Thapliyal and M. B. Srinivas, “Novel Reversible Multiplier Architecture Using Reversible Logic Gates,” *International Conference on VLSI Design*, pp. 569–574, 2006.

[5] M. Venkatesha and K. Dayanand, “An Improved Design of a Multiplier Using Reversible Logic Gates,” *International Journal of Computer Applications*, vol. 25, no. 10, pp. 1–5, 2011.

[6] S. Kumar and R. K. Sharma, “Design and Implementation of 32-bit Wallace Tree Multiplier,” *International Journal of Engineering Research and Applications*, vol. 4, no. 5, pp. 12–16, 2014.

[7] M. Haghparast and K. Navi, “A Novel Reversible Full Adder Circuit for Nanotechnology Based Systems,” *Journal of Applied Sciences*, vol. 7, no. 24, pp. 3995–4000, 2007.

[8] S. Kotiyal, H. Thapliyal, and N. Ranganathan, “Design of Efficient Reversible Logic Based Multiplier Circuits,” *IEEE Computer Society Annual Symposium on VLSI*, pp. 206–211, 2012.

[9] M. P. Frank, “Introduction to Reversible Computing: Motivation, Progress, and Challenges,” *Proceedings of the IEEE*, vol. 99, no. 4, pp. 1–17, Apr. 2011.

[10] H. T. Vergos, D. Nikolos, and C. Efstathiou, “Area-Time Efficient Carry Look-Ahead Adders,” *IEEE Transactions on Computers*, vol. 43, no. 12, pp. 1393–1397, Dec. 1994.