Grenze International Journal of Engineering and Technology, June Issue

Optimized Design and Analysis of Low Power 32-Bit Reversible Adder

Hamsa S[1], Asha Bharathi S[2], Shankara C[3], P Rupesh Kumar[4], Tarun MR[5] and Sparsha LM[6]
[1-2,4-6]Jyothy Institute of Technology, Department of ECE, Bangalore, India
Email: hamsagudibande@gmail.com, asha.bharati@jyothyit.ac.in, rupeshp0704@gmail.com, tmr4831@gmail.com, sparshalm248@gmail.com
[3]Government Polytechnic Malavalli Mandya, India
Email: shankara151987@gmail.com

***Abstract*— There is a great demand for low power design in the emerging computing era. Reversible logic emerges as perspective design that is useful in power-constrained environments, minimizes the information loss and energy dissipation unlike conventional irreversible logic. This work explores the low-energy design and performance optimization of a 32-bit reversible adder using a 45 nm full-custom approach. The design process starts with implementing fundamental reversible logic elements namely the Peres, Toffoli, Fredkin, and Feynman gates and later enhances them using transmission-gate-based optimization. The design adopts a modular approach by creating a one-bit reversible addition module and cascading multiple units to construct a 32-bit reversible carry-based adder, while minimizing the generated garbage outputs. The optimized reversible addition module enhances the standard design by lowering both area and power consumption. A 1-bit reversible adder design shows 20.2% power savings and 8.8% speed improvement over standard CMOS. While for the 32-bit adder, power drops 30.3% and delay improvement of 4.8% is achieved over conventional design, demonstrating efficiency for low power applications. The 32-bit reversible adder is designed to form as a fundamental element for reversible ALU used in reversible processor. The results demonstrate low power, reduced transistor count, scalable high speed reversible adder making it a promising approach for emerging computing paradigms.**

***Index Terms*— Reversible Gates, Garbage outputs, Transmission Gate, Power Optimization, Reversible Adder.**

I. INTRODUCTION

The demand for energy-efficient computing has intensified due to the rapid growth of portable and high-performance digital systems. Traditional irreversible logic circuits inherently lose information during computation, resulting in heat dissipation as described by Landauer's principle [1]. As CMOS technology approaches its physical limits, power consumption has become a major bottleneck in VLSI design. This has motivated the exploration of alternative design methodologies such as reversible computing, which aims to reduce or eliminate information loss and thereby minimize energy dissipation [2]. By offering the bijective mapping amongst inputs and outputs, reversible logic circuits guarantee that no data is lost during the computation process. This makes the logic circuits theoretically capable of operating with near-zero power dissipation [3].

Grenze ID: 01.GIJET.12.2.341_21

Due to these characteristics, reversible logic has gained significant scope in low-power VLSI systems and future computing architectures where information preservation is critical [4]. In order to create effective and scalable designs for reversible logic, it is important to minimize garbage outputs, constant inputs, gate count.
In the present work, a reversible 32-bit adder is designed to serve as a fundamental building block of an Arithmetic Logic Unit (ALU) for a low-power reversible processor. The adder is implemented using reversible logic gates to preserve information and reduce energy dissipation. This reversible 32-bit adder forms the core component of the reversible ALU, contributing to the realization of an energy-efficient processor architecture suitable for future low-power and information-preserving computing systems.

II. Literature Survey

Landauer's and Bennett's pioneering studies established the theoretical basis for reversible computation by demonstrating the link between information loss and energy dissipation, and by proving that logically reversible computation can circumvent this energy loss [1,2]. These insights provided the foundation for modern reversible logic design and triggered extensive research into energy-preserving digital circuits. Early progress in reversible logic focused on constructing primitive reversible gates capable of forming complex systems. Toffoli introduced a gate which can synthesize reversible function [4], while Fredkin proposed a conservative reversible gate that preserves the logic high of design making it suitable for energy-preserving switching networks [5]. Peres later introduced an efficient reversible gate that reduced quantum cost by combining Toffoli and CNOT functionality [6]. These gates became the fundamental components for building reversible arithmetic circuits, memory units, and processor elements.
Recent research has emphasized optimizing reversible arithmetic circuits for reduced hardware complexity. Thapliyal and Ranganathan designed reversible adders and sequential circuits with optimized speed, and unused/non-computed outputs [7]. Haghparast and Navi introduced an improved full adder that is reversible using a optimized TG design that further reduced unused (garbage) outputs and minimized hardware cost [8]. These advancements enabled the development of reversible ALUs and processor architectures, which form the basis for future low-power and quantum-ready computational platforms.

III. Methodology

The design methodology for the proposed 32-bit low-power reversible adder leverages contemporary reversible logic design approaches [9, 10].

A. Reversible Gate Design

Basic reversible logic gates (Peres, Toffoli, Fredkin, Feynman) were designed and simulated in Cadence Virtuoso following reversible design constraints reported in latest reversible gate design works [2].
B. Custom Conventional Logic (XOR, AND, OR, NOT)
XOR and AND gates were implemented using TG-based structures, OR and NOT gates were realized at Full-Custom level. The design approach aligns with reversible-irreversible hybrid architectures [5].

C. Adder Design (1-bit) Construction

The adder was implemented with two Peres gates and one OR gate, consistent with Peres-based adder structures used in reversible arithmetic circuits [1].

D. Extension to 32-bit Adder

The 1-bit reversible addition unit was extended through cascading to realize a 32-bit ripple-carry adder. The design preserves key reversible logic principles: balanced inputs and outputs, elimination of fan-out, and reduction of garbage and constant inputs, consistent with current reversible logic methodologies [11].

E. Simulation & Performance Evaluation

Cadence Virtuoso was used to perform functional, power, and delay analyses, with evaluation metrics including gate count, garbage outputs, dynamic power, and propagation delay, in accordance with recent reversible adder benchmarks [12]

IV. Reversible Gates

In Reversible gates the computations are not only in forward direction but also in reverse direction. Reversible gates has constant number of input and outputs, it also gives some unused fanouts in computations but helpful in

obtaining reversible logic. During computations of reversible logic the data will not be lost as they are reversible also it gives an advantage of low power design but gives garbage outputs. Reversible logic gates produce a unique output for every input combination. Key gates used in this work include:

- Feynman Gate – used for fan-out and copying signals.
- Peres Gate – utilized for optimized XOR–AND logic and full adder design.
- Toffoli Gate – universal reversible gate supporting AND-controlled operations.
- Fredkin Gate – swap gate for multiplexing operations.

These gates ensure the design yields minimal garbage outputs and constant inputs, enhancing efficiency.

A. Feynman Reversible Gate

The gate is designed to have A, B as inputs & outputs as P, Q where the first output (P) gives A which is considered as garbage or unused output while the second output (Q) gives A xor B function given as in (1)

$$P = A \quad \text{and} \quad Q = A \text{ xor } B \qquad (1)$$

The Fig 1 shows that Feynman gate is designed using 45nm Technology using Cadence Virtuoso.

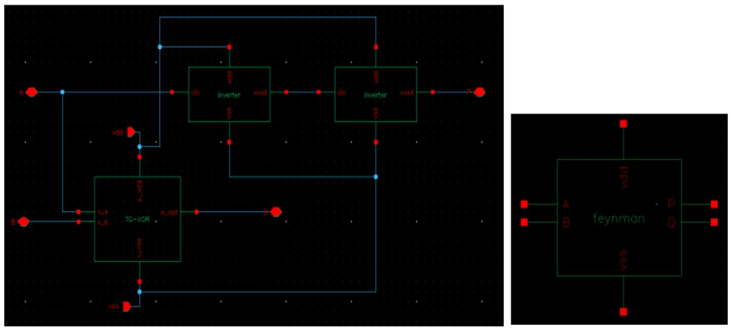

Fig 1: Implementation of Feynman Gate

B. Peres Reversible Gate

The gate is implemented to have A, B, C and P, Q, R as inputs and outputs respectively. The first output (P) gives A which is considered as garbage output while the second output (Q) gives A xor B and the third output (R) gives A.B xor C function given as in (2)

$$P = A \qquad Q = A \text{ xor } B \qquad R = A.B \text{ xor } C \qquad (2)$$

The Fig 2 shows that Peres gate is designed using 45nm Technology using Cadence Virtuoso.

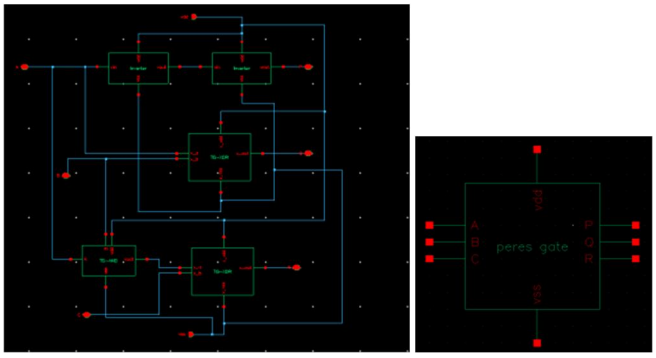

Fig 2: Peres gate Implementation

C. Toffoli Reversible Gate

The gate is designed to have A, B, C and P, Q, R as inputs and outputs respectively. The first output (P) gives A and second output (Q) gives B which is considered as garbage output while the third output (R) gives A.B xor C function given as in (3)

$$P = A \qquad Q = B \qquad R = A.B \text{ xor } C \qquad (3)$$

The Fig 3 shows the Toffoli gate is designed using 45nm Technology using Cadence Virtuoso.

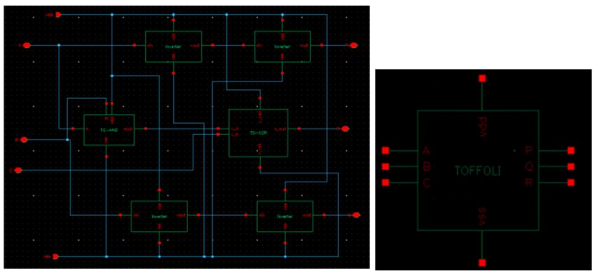

Fig 3: Toffoli gate Implementation

D. Fredkin Reversible Gate

The gate is implemented to A, B, C and P, Q, R as inputs and outputs respectively. The first output (P) gives A considered as garbage output, while second output (Q) gives A'.B xor A.C and third output (R) gives A'.C xor A.B function given as in (4)

$$P = A \qquad Q = A'.B \text{ xor } A.C \qquad R = A'.C \text{ xor } A.B \qquad (4)$$

The Fig 4 shows the Fredkin gate is designed using 45nm Technology using Cadence Virtuoso.

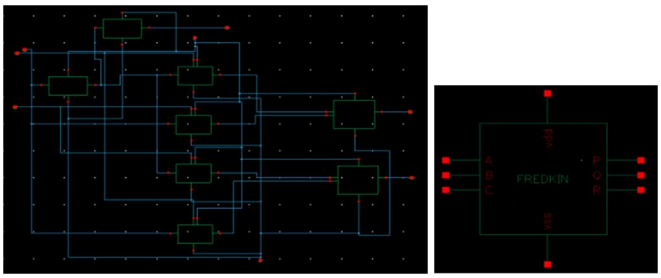

Fig 4: Fredkin gate Implementation

V. IMPLEMENTATION

A. Reversible 1-Bit Full Adder

The adder is implemented with two Peres gates and one OR gate as depicted in the schematic and its behavioural characteristics are represented in Fig 5. The first Peres gate generates the partial sum ($A \oplus B$) and intermediate carry ($A \cdot B$). The second Peres gate computes the final sum output by combining ($A \oplus B$) with the input carry. The OR gate produces the final carry-out from the two intermediate carries.

This configuration ensures a bijective input–output mapping, minimal garbage outputs, and correct reversible operation. The 1-bit adder forms the fundamental building in designing higher-bit adders.

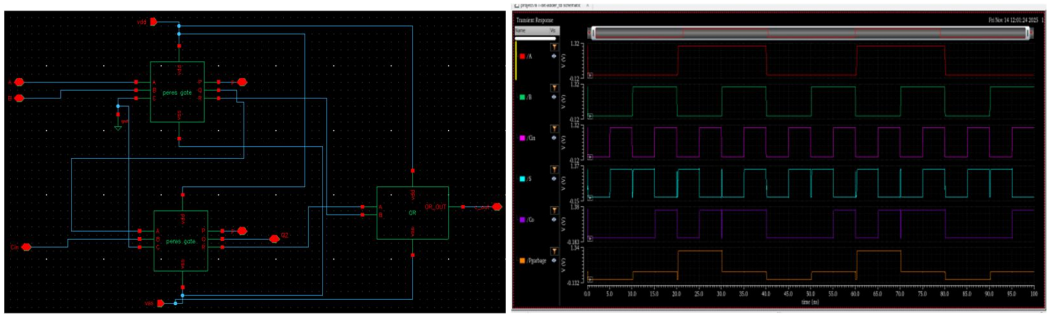

Fig 5: schematic and behavioural characteristics of Reversible 1-bit Adder

B. Reversible 4-Bit Ripple Carry Adder

The adder is realized by cascading four 1-bit full adder modules but managing and connecting the outputs in a way that unused outputs are minimized to attain the behaviour characteristics as shown in Fig 6. Each stage accepts its own A[j], B[j], and the $C_{out}[j-1]$. A reversible constraint of no fan-out is maintained by using direct cascading without duplicating signals.

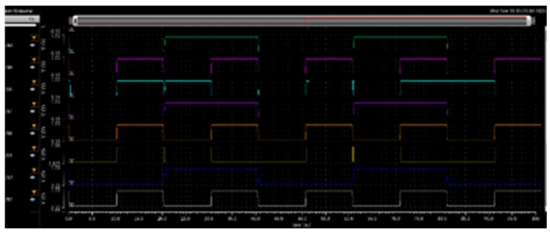

Fig 6: Behavioural characteristics of Reversible 4-bit Adder

C. Reversible 8-Bit Full Adder

The adder is implemented by cascading 8 adders in series but managing and connecting the outputs in a way that garbage outputs are minimized. The architecture follows a standard ripple-carry structure, where carries propagate from the LSB to the MSB. The reversible design maintains the number of inputs and outputs same across all stages as designed in Fig 7. The 8-bit adder allows intermediate verification of functional correctness and power behaviour before scaling to full 32-bit width.

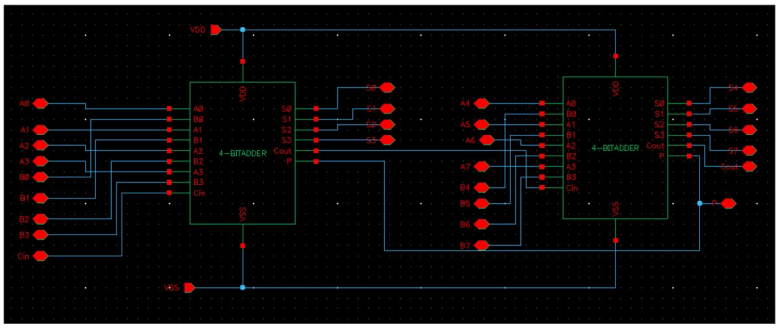

Fig 7: Schematic design of 8-bit Reversible Adder

D. Reversible 32-Bit Ripple Carry Adder

The adder is formed by cascading 32 identical 1-bit reversible full adder units but managing and connecting the outputs in a way that garbage outputs are minimized as designed in Fig 8. The ripple-carry approach ensures structural reproducibility while maintaining reversible logic constraints. Each stage contributes its own sum output and propagates the carry to the next stage. The design preserves bijective mapping, no fan-out, minimized garbage outputs. Simulation results confirm correct operation across input patterns, with power and delay metrics aligning with reversible logic expectations.

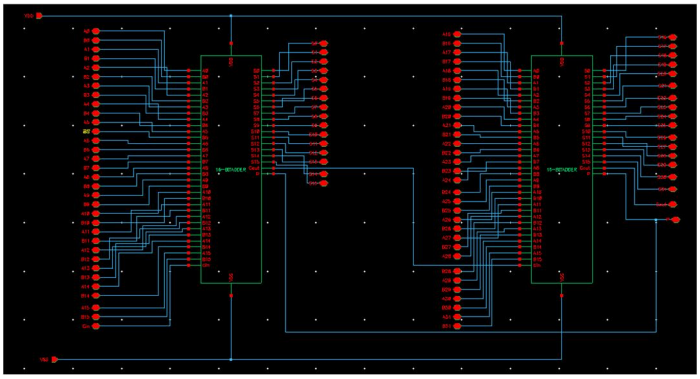

Fig 8: schematic of 32-bit adder

VI. LIMITATIONS

The proposed reversible adder, while efficient in power and delay, has certain limitations. In particular, the design suffers from garbage outputs with limited utilization and increased transistor count leading to an area bottleneck. Additionally, reversible adders inherently involve higher design complexity and scalability challenges for larger bit-widths, as well as fan-out and interconnect constraints.

VII. Conclusion

The current work, has successfully implemented a low-power 32-bit reversible adder based on reversible logic principles, validated through Full Custom simulation in Cadence Virtuoso. Starting from fundamental reversible gates (Peres, Toffoli, Fredkin, Feynman) and custom XOR/AND/OR/NOT modules. A reversible 1-bit full adder was designed using Peres gates and an OR gate, verified its correct functionality, and then scaled the design to a 32-bit ripple-carry architecture. Table 1 shows the evaluation results of the reversible and conventional adders. The optimized reversible adder outperforms conventional designs in power and delay. The 1-bit reversible adder consumes 31.79 µW that is 20.2% less than the conventional design and achieves a delay of 20.25 ns, 8.8% faster. For the 32-bit adder, power drops to 2.02mW (30.3% lower) with a delay of 20 ns (4.8% faster) compared to the conventional design, demonstrating its efficiency for low-power applications. However, the larger transistor count required for reversible logic presents a potential area bottleneck. For the 32-bit reversible adder, the maximum operating frequency is 50 MHz, obtained from the 20 ns delay, and the energy per operation is 40.4pJ based on the power–delay product. These metrics further validate the reversible adder's suitability for low-power applications.

Table I: Evaluation Results of the Design

Circuit	Parameters					
		Power	Delay	Transistor Count	Garbage Output	Garbage Utilization
Feynman Gate		243ns	20.26ns	16	1	1
Peres Gate		785.2Nw	20.16ns	36	1	0.6667
Toffoli Gate		418.6nw	40.46ns	28	1	0.6667
Fredkin Gate		29.54nw	40.47ns	60	1	0.6667
1-bit Adder	Reversible	31.79µW	20.25ns	78	1	Used as Buffer/Inverter for ALU
	Conventional	39.83µW	22.16ns	42	Nil	Nil
4-bit Adder	Reversible	225.3µW	19.99ns	318	4	Used as Buffer/Inverter for ALU
	Conventional	270 µW	20 ns	168	Nil	Nil
8-bit Adder	Reversible	1.867µW	9.84ns	636	8	Used as Buffer/Inverter for ALU
	Conventional	520 µW	10.5 ns	336	Nil	Nil
32-bit Adder	Reversible	2.02mW	20ns	2544	32	Used as Buffer/Inverter for ALU
	Conventional	2.9 mW	21 ns	1344	Nil	Nil

Future Work

The proposed 32-bit reversible adder can serve as a fundamental building block for designing a low-power, high-performance ALU, which in turn can be integrated into a reversible processor architecture. Future work can explore further optimization of the reversible logic using advanced technologies such as FinFETs, which offer reduced leakage, improved drive current, and better scalability at nanometre nodes. Additionally, exploring pipelining, parallelism, and garbage output minimization techniques in reversible ALU design can further enhance energy efficiency and computational speed, making reversible architectures more viable for next-generation low-power processors and quantum-inspired computing systems.

References

[1] R. Landauer, "Irreversibility and heat generation in the computing process," IBM Journal of Research and Development, vol. 5, no. 3, pp. 183–191, 1961.

[2] C. H. Bennett, "Logical reversibility of computation," IBM Journal of Research and Development, vol. 17, no. 6, pp. 525–532, 1973.

[3] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge, U.K.: Cambridge University Press, 2010.

[4] T. Toffoli, "Reversible computing," in Automata, Languages and Programming, J. W. de Bakker and J. van Leeuwen, Eds., Springer, 1980, pp. 632–644.

[5] E. Fredkin and T. Toffoli, "Conservative logic," International Journal of Theoretical Physics, vol. 21, no. 3–4, pp. 219–253, 1982.

[6] A. Peres, "Reversible logic and quantum computers," Physical Review A, vol. 32, no. 6, pp. 3266–3276, 1985.

[7] H. Thapliyal and N. Ranganathan, "Design of reversible sequential circuits optimized for quantum cost, delay and garbage outputs," ACM JETC, vol. 6, no. 4, pp. 1–29, 2010.

[8] 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.

[9] P. K. Kadbe and S. D. Markande, “Efficient design of reversible adder and multiplier using Peres gates,” Applied Sciences, vol. 14, no. 20, p. 9385, 2024.
[10] S. S. Samrin, et.al., “Design of logic gates using reversible gates with reduced quantum cost,” Global Transitions Proceedings, vol. 3, no. 1, pp. 136–141, Jun. 2022.
[11] H. A. Bhoskar and H. S. Thakar, “Review on Reversible Logic Gates,” International Journal for Research in Applied Science and Engineering Technology (IJRASET), vol. 11, no. 11, pp. 2058–2063, Nov. 2023.
[12] K. S. Tiwari, et.al., “Reversible logic gates and applications – A low power solution to VLSI chips,” Mathematical Modelling of Engineering Problems, vol. 11, no. 3, pp. 705–720, 2024.
[13] Hamsa S., et.al.,. Low power device design application by magnetic tunnel junctions in Magnetoresistive Random Access Memory (MRAM). SN Appl. Sci. 1, 828 (2019).
[14] Hamsa S, et.al., “Composition of magnetic tunnel junction-based Magnetoresistive random access memory for Field Programmable Gate Array”, Current Science – Research Communications, Vol 119, Issue 1, pp 119-123, July 2020.