International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Efficient Design of Configurable Logic Block with Customized LUT using Reversible Fault Tolerant Gates

Ravi L. S.[*1], Dr. Naveen K. B.[2]

Submitted: 10/12/2023 **Revised**: 14/01/2024 **Accepted**: 31/01/2024

Abstract: The parity of the input vector must match the parity of the output vector in a fault-tolerant reversible logic gate circuit. It enables the circuit's problem to be discovered. As a result, parity-preserving reversible logic will assist in the development of fault-tolerant systems for nanotechnology. It is commonly recognized that fault-tolerant (FT) reversible logic gates (RLG) are compatible with revolutionary computing paradigms such as optical and quantum computing. We presented reversible fault-tolerant lookup tables (LUTs) in this paper, which are employed in the development of Configurable logic blocks (CLB). CLB architecture employs fault-tolerant reversible logic components such as D-Latch, MS-Flip-flop, and Multiplexer. The suggested architecture is simulated, tested logically, and implemented using an FPGA Spartan 3. The simulation findings and implementation demonstrate the reversible fault tolerant gate design's functioning. The power dissipation and delay in reversible fault tolerant gates are found to be less, with a power reduction of around 95.5 % at 90 nm CLB technology.

Keywords: *Look-Up Table, FPGA, Fault Tolerant, Reversible Logic, Low Power Dissipation, Garbage Output*

1. Introduction

Energy is dissipated as a result of irreversible hardware computing, regardless of how it is realized due to the loss of information, is shown by R. Landaurer's research in early 1960. Each k: T: ln2 joule of information is dissipated, where k is the Boltzman constant is denoted by k and absolute temperature is denoted by T [1, 2]. The huge energy dissipation is avoided by using reversible logic gates for making a circuit, as shown by Benett demonstrated in 1973. A system to continue to function successfully by using the Fault tolerance even if any of its components fail (due to one or more faults). FPGAs are typically made up of a set of customizable I/O blocks, interconnects, and logic blocks [3,4]. The FPGA can be customized to meet the needs of any application. FPGAs have evolved from simple logic to extremely sophisticated programmable fabrics as a result of the same semiconductor technological advancements which have brought processors to their performance limits. Plessy logic blocks and Lookup-Table (LUT) are the most common logic blocks. A LUT can use fewer logic blocks to implement more logic with more inputs. As a result, it aids in a smaller routing area [5,6]. The area and delay performance are increased by using the 3 to 4 input LUT size shown by the authors in the paper. As a result, we'll examine a generic 4-input LUT-based logic block.

The main objective of the project work are as follows:

- To create a Configurable Logic Block (CLB) based on Reversible Fault Tolerant Gates (RFTG)
- Design of Reversible Fault Tolerant (RFT) D-latch, Master Slave Flip Flop and Multiplexer using our proposed gate, which aims for the shortest unit delay, the lowest quantum cost, and the smallest number of gates.
- A novel RFT multiplexer with a number of gates, unit latency, and lower quantum cost has been presented.
- The unit delay, quantum cost, garbage, and the number of gates is minimized by using the enhanced RFT LUT-based FPGA CLB.
- The CLB Models' performance evaluation is determined by using the FPGA technology with 90nm (Backend) logic.

The following is the structure of this paper, the previous understanding of FPGA and reversible logic design as well as a review of relevant research are shown in section 2. Section 3 shows an enhanced architecture of a RFT 4 to 1 multiplexer, master slave flip flop and D-latch. The simulation result and performance analysis are discussed in Section 4. Finally, the paper is concluded in the last Section.

2. Basic Definitions and Literature Survey

The terms quantum cost, LUT, unit delay, garbage output, reversible fault tolerant gate is defined in this section. In

[1]*Research Scholar, Dept. of Electronics and Communication Engg., B.G.S Institute of Technology, Adichunchanagiri University, B.G. Nagara,*
[1]*Assistant Professor, Rajeev Institute of Technology, Hassan.*
ORCID ID : https://orcid.org/0009-0005-8405-5977
[2]*Professor, Dept. of Electronics and Communication Engg., B.G.S Institute of Technology, Adichunchanagiri University, B.G. Nagara, Email: naveenkb@bgsit.ac.in*
** Corresponding Author Email: ravigowda1234@gmail.com*

addition, the most recent related studies are discussed briefly.

2.1. Reversible Gate

A digital circuit is represented by a reversible gate with the same no. of outputs and inputs. There will be k outputs if there are k inputs. Each output pattern is distinct from the input pattern. One-on-one correspondence is the term for this, which is represented by k*k.

2.2. Fault Tolerant Gate

A fault-tolerant gate ensures input-output parity. The analytical expression for a k*k RFTG is,

$$I1\_I2.\ldots.\_IN = O1\_O2.\ldots.\_ON$$

The ability to identify faulty signals is a key attribute of a fault tolerant gate. Fault tolerant reversible gates include the new fault tolerant gate (NFT), Modified Islam gate (MIG), Islam gate (IG), Double Feynman gate (F2G), and Fredkin gate (FRG).

2.3. Garbage Output

Employing garbage outputs maintains the reversible characteristic. These garbage outputs are either undesired or unusable outputs. Garbage outputs are kept expensive.

2.4. Quantum Cost

A cascade of quantum gates can be used to replace a RLG in a circuit. The RG is designed by using the 2×2, 1×1 reversible gates or the number of quantum gates, is called the quantum cost. The q-bits are used in quantum gates whereas pure logic values are used in RLG. There are numerous basic quantum gates. Controlled-V+ gate, controlled-V gate, controlled-NOT (CNOT) gate and NOT gate are some examples. A single qbit is inverted using the NOT gate. Only if the control q-bit is 1, The CNOT gate (Controlled-NOT gate) performs the inversion of the input q-bit. The square root of NOT (SRN) gate is another name for the controlled-V gate. Inversion operation is equal to two consecutive V operations. The square root of the NOT gate is the controlled-V+ gate. FRG has a quantum cost of 5, IG has a cost of 7, F2G has a cost of 2, MIG has a cost of 7 and NFT has a cost of 5.

2.5. Quantum Gate Complexity

Elementary quantum gates such as Controlled-V+ gates, Controlled-V, CNOT, and NOT are used to construct each reversible logic gate. The RLG is designed by using the number of elementary quantum gates, which is represented as quantum gate complexity. The controlled-V+ gate and Controlled-V gate (SRN) are similar gates. Four CNOT gates and three controlled-V & V+ gates make up a Fredkin gate.

2.6. Delay

A is expressed in terms of unit delay, which corresponds to a 1*1 or 2*2 elementary quantum gate's delay, like controlled-V+ gate, Controlled-V gate, CNOT, and NOT. As a result, measuring delay requires the circuit's logical depth. The gate delay for Fredkin is 5.

2.7. Power

The power of each reversible logic gate adds up to the total power of the circuit. There are n reversible logic gates with powers p1; p2; p3; ::::::::; pn in a reversible logic circuit then the reversible logic circuit's overall power is given by:

$$P = \sum P_i \tag{1}$$

Quantum-dot cellular automata circuits are designed by using the layout strategy, described by the author in [7]. Designing configurable memory and logic blocks at a Low Cost are developed by using the presented technique. For FPGA-based systems, a novel approximate adder design methodology is proposed by the author in [8] with enhanced performance of the system while keeping SWaP benefits. The overall optimization is achieved, and also the long carry chain in light of FPGA hardware architecture is fine-tuned by using this methodology with only a single Look-Up Table (LUT) delay, regardless of operand size. When compared to the existing best-known LUT-based BCD adders, a parallel BCD adder is described by the author in [9], it gains a radical achievement VHDL is used to code the proposed BCD adder and Virtex-6 platform is used for implementation with ISE 13.1. It has a -3 speed grade for the Xilinx FPGA XC6VLX75T. The reversible PLAs is designed by the author in [10] that maximize the utility of garbage outputs while lowering the amount of ancilla inputs. The range of these gates are analyzed by author in [11]. The reversible circuits are designed, and implemented by using these gates for quantum computation. Using the reversible logic principle, the low-power Configurable Fault-Tolerant Embryonic Hardware is designed by the author in [12] lower area overhead.

3. Proposed Methodology

This section describes the algorithms, design layout, theoretical characteristics, and operation of the proposed logic parts for use in RFT FPGAs based on LUTs.

- “Mubin_Sworna_Hasan (MSH) Gate”
- “Mubin_Sworna_Babu (MSB) Gate”

The following section contains these two Gates Block-Diagrams, Truth-Table and transistor-realization.

Mubin_Sworna_Hasan (MSH) Gate: Consists of (a, b, c, d} → inputs and {p.q.r.s} → output-vectors and with Quantum-Realization. Figure 2 and 1 displays the quantum

realization and block diagram of MSH gate. Truth table is displayed in table 1.

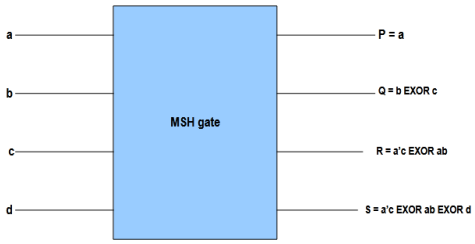


Fig. 1. Block diagram of MSH – Gate

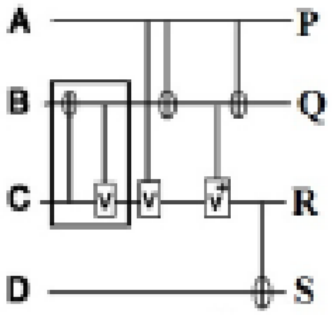


Fig. 2. MSH – Quantum Realization

Table 1. MSH Gate

INPUT				OUTPUT				Parity
A	B	C	D	P	Q	R	S	1=Odd
								0=Even
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	1
0	0	1	0	0	1	1	1	1
0	0	1	1	0	1	1	0	0
0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	0
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	0	1
1	0	0	0	1	0	0	0	1
1	0	0	1	1	0	0	1	0
1	0	1	0	1	1	0	0	0
1	0	1	1	1	1	0	1	1
1	1	0	0	1	1	1	1	0
1	1	0	1	1	1	1	0	1
1	1	1	0	1	0	1	1	1
1	1	1	1	1	0	1	0	0

Mubin_sworna_Babu (MSB) Gate: it's Block-Diagram represents the {a,b,c,d,e,f} → inputs and {p,q,r,s,t,u} → output-vectors. Figure 4 and 3 shows the quantum realization and block diagram of MSB gate.

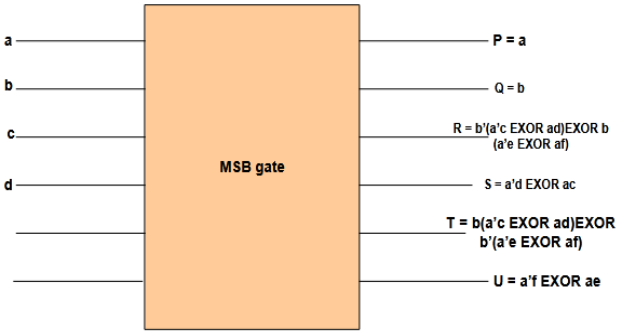


Fig. 3. Block diagram of MSB – Gate

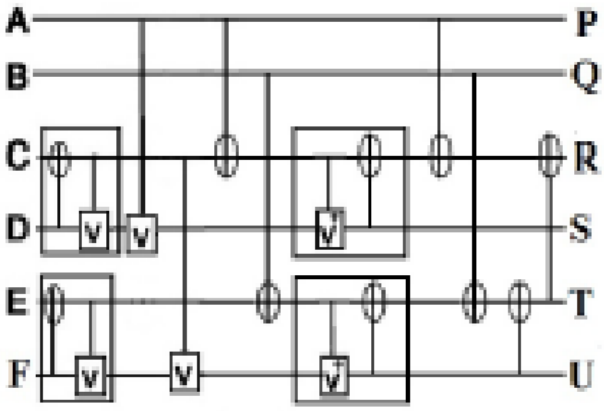


Fig. 4. MSB – Quantum Realization

3.1. RFT D_Latch

By using proposed MSH-gate, we design the "Reversible-Fault-Tolerant" D_latch", and generates the required output, s => clk'.Feed-back EXOR clk. D along with 2-g1 and g2 garbage output and 1-constant input. It is designed by 4 MSH gate. The total quantum cost of this D-latch is 10. Figure 5 shows the RFT D latch designed by MSH gate.

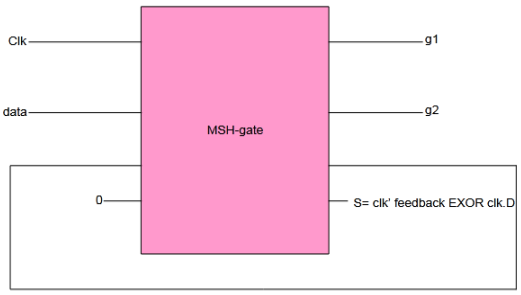


Fig. 5. RFT D_latch

3.2. RFT Master – Slave Flip – Flop (MS-FF)

The following fig: 6 represents the proposed "RFT-Master slave FF" applying with 12-FRG, 2-F2G gate and described 5-RFT D_latch, which generates the final output Q and Q+. and g1,g2,g3,g4 are the garbage outputs. The

total quantum cost of this RFT MS-FF is 19.

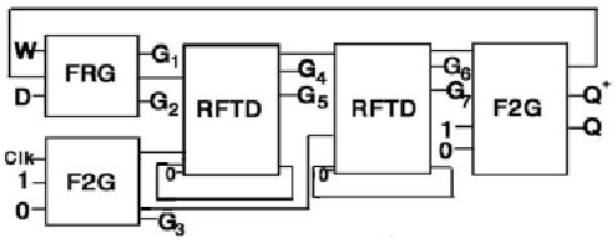


Fig. 6. Master Slave Flip Flop

3.3. RFT (4x1) – Multiplexer

For RFT (4x1)-mux design, we apply the proposed 8-MSB Gate. This multiplexer produces the least amount of garbage, which is proved in following theorem. The total quantum cost of this RFT Multiplexer is 2.

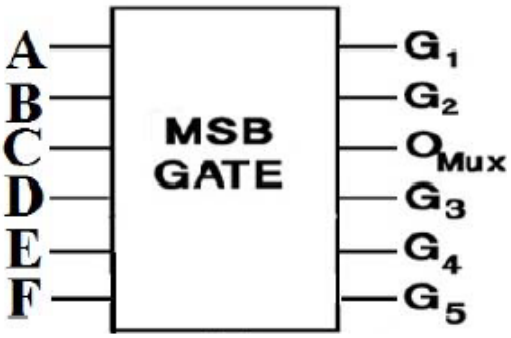


Fig. 7. 4x1 MUX

Figure 7 shows the design of RFT Multiplexer using MSB gates. Table 2 shows the truth table of Truth Table "4x1 MUX.

Omux =a'b'c + a'be + ab'd + abf

Table 2. Truth Table "4x1 MUX"

a	b	Output
0	0	c
0	1	d
1	0	e
1	1	f

Theorem-1 : $4x - to - x$ "Reversible-fault-tolerant mux" contains $4x + 1$ garbage out-puts with no constant input and $x => no.data\ bits$.

Lemma (i): The designed x-i/p "Reversible-fault-tolerant D_latch" contains x −Reversible-FT gates, $2x$ − Garbage-outputs and $6x$ −Quantum_cost.

Lemma (ii): The x-input "Reversible-fault-tolerant Write Enable Master_slave-Flip_Flop" contains at least $5x$-Reversible-FT gates, $7x$-Garbage-outputs and $21x$ Qc.

3.4. RFT X-Input "LOOK-UP-TABLE" (LUT)

3.4.1. Three-Input "Look-up-Table"

The designed RFT-3_input "Look-up-Table" contains 20-MSB gates and 9-FRG gate, and generates the garbage-output and Omux final-output. The total quantum cost of this RFT LUT is 164. Figure 8 shows the design of 3-input LUT gate using MSB gates. Table 3 displays the truth table for 3 – i/p LUT.

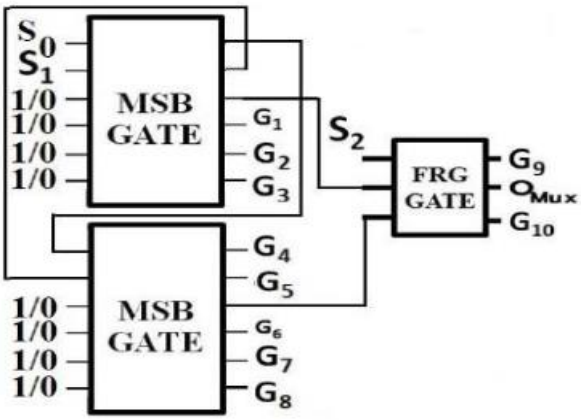


Fig. 8. Three – input "LUT"

Omux =(a'b'c).(a'be).(ab'd).(abf)

Table 3. Truth Table of 3 – i/p LUT

a	b	Output
0	0	c
0	1	d
1	0	e
1	1	f

3.5. Four – Input "Look Up table"

In this we used "MSB" gates with four inputs and generate the G1, G2.....G19 garbage values and Omux as output results. Figure 9 shows the design of 4-input LUT gate using MSB gates. Truth table is displayed in table 4.

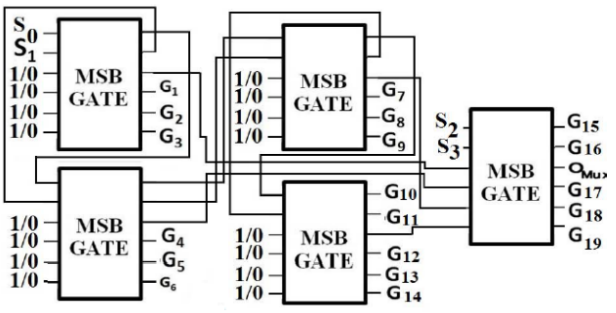


Fig. 9. Four – input "Lookup table"

Table 4. Truth table of 4 – i/p LUT

S3	S2	S1	S0	Output
0	0	0	0	I0
0	0	0	1	I1
0	0	1	0	I2
0	0	1	1	I3
0	1	0	0	I4
0	1	0	1	I5
0	1	1	0	I6
0	1	1	1	I7
1	0	0	0	I8
1	0	0	1	I9
1	0	1	0	I10
1	0	1	1	I11
1	1	0	0	I12
1	1	0	1	I13
1	1	1	0	I14
1	1	1	1	I15

3.6. RFT CLB for FPGA

The Reversible fault tolerant CLB of FPGA designed by using Verilog and simulated by Modelsim. In this system taken RFT- four- input - LUT from I/O block. And {S0, S1, S2, S3} are the selector for selection of bits from the I/O-block. Used four-RFT_LUT, RFT-(2x1)-mux, and RFT-Flip-flops which generates the (Q1),(0 mux) and Q or (Q+) out-put results. Figure 10 shows the proposed RFT CLB of FPGA.

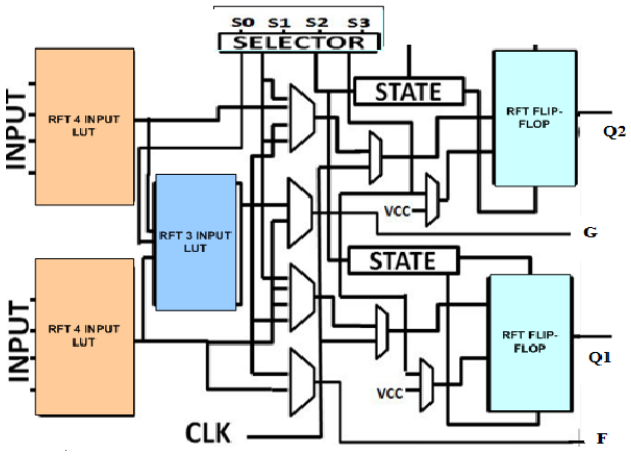


Fig. 10. "Reversible FT" CLB of FPGA

The proposed programmable-gate contains two-configurable components: (CLBs and IOBs).

"Configurable-Logic-Blocks": for designing programmable logics, provides functional elements, "I/O-Blocks": interface between internal signals and package pins.

The "Configurable-Logic-Blocks" has two, Four-input-function-generators (F1, F2, F3, F4 and G1, G2, G3, G4) and 3rd H-function-generator contains 3-inputs. Either (0, 1, or 2) are the inputs and it will be the output of F & G. Every "CLBs" has 2-independent storage components, which are used for storage of output function-generators. The 13-input-CLB and 4-output-CLB provide access to the storage-elements and function-generators. All these i/p and o/p are connected with programmable-interconnected-resources outside the block. There are 4-independent i/p are available for each function-generators F1...F4 and G1...G4 and F' & G' are the outputs. The 3rd function -generator H has H' output label.

Algorithm 1: Design of an n-input Reversible Fault Tolerant (RFT) LUT based Configurable Logic Block

Input : Input from I/O block
Output: $Omux$, which will carry to next logic block

begin
  while *I/O Block is not empty* **do**
    $[S_0, S_1, .., S_n]$:= selection bits from I/O Block;
    Apply 4-input RFT LUT where Input:={ $[S_0, S_1, .., S_n]$}; Output:={Q_0};
    Apply RFT Flip Flop where Input:={ Q_0, clock signal}; Output:={Q or Q^+};
    Apply RFT 2×1 Mux where Input:={ Q_0 , Q or $Q+$}; Output:={ O_{mux}};
    $SelectionBit_{RFTMUX}$:={$Control_{bit}$};
  end while
end

3.7. Application of the Proposed FPGA to Design an n-bit Adder Circuit

We implemented a full adder circuit using the DSCH simulation tool and our implemented RFT CLB design, as seen in Figure 10, where Block 1 evaluates the sum (A ⊕ B ⊕ C) of three input bits (A, B, and C) and Block 2 evaluates the carry (AB + AC + BC). While our design has complete adder logic, we have evaluated the delay and power usage. This method is scalable and hence generalizable to n-bit adders.

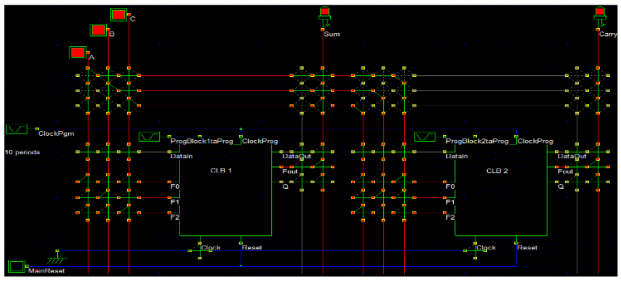


Fig. 11. Design of a Full Adder

4. Results and Discussions

This section simulates and analyses the performance of the

proposed RFT MSB gate, MSH gate, and reversible fault tolerant components of the LUT Logic Block.

The proposed components' simulations (RFT Master slave FF, LUT, Multiplexer and D-Latchbased FPGA) are shown in Figures 12 – 15.

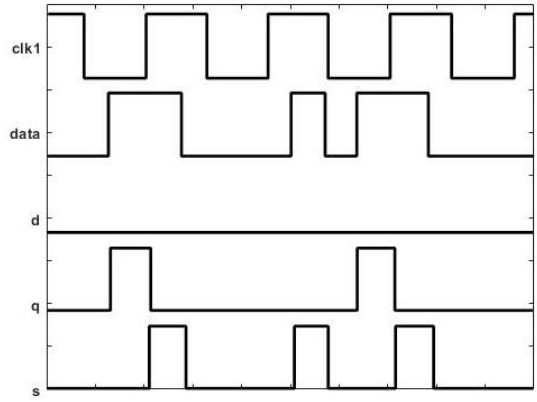


Fig. 12. Simulation of the proposed RFT D-Latch

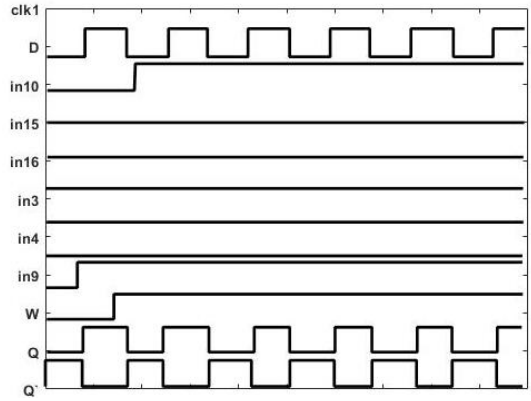


Fig. 13. Simulation of the proposed RFT 4x1 Multiplexer

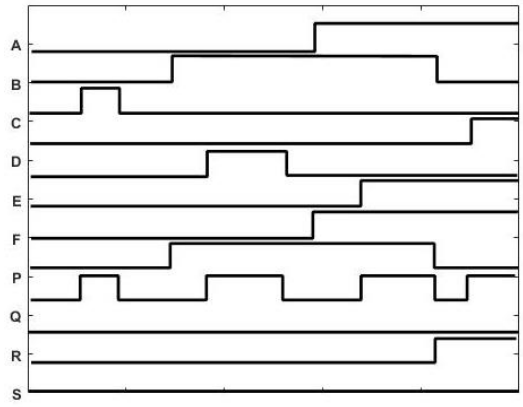


Fig. 14. Simulation of the proposed RFT Master Slave Flip flop

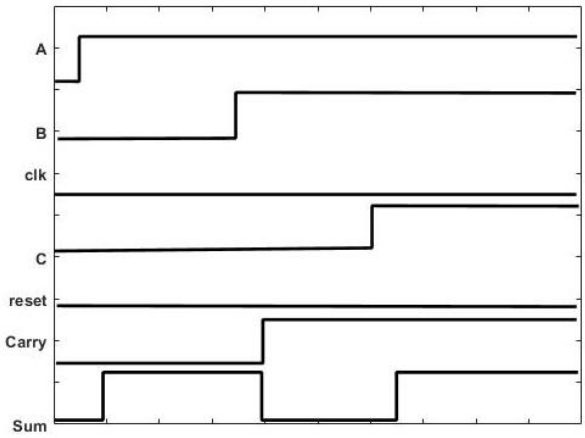


Fig. 15. Simulation of the proposed RFT FPGA

In Figure 12, CLK denotes clock input, D denotes data input, Q denotes the D-non-complementary Latch's output, and Q+ denotes the negation of Q. Figure 12 illustrates that when CLK=1, D becomes Q. Latch keeps previous state when CLK=0.

Following is the simulation result of the FT MS-Flip-Flop, where CLK represents the same as D-Latch notations. W often stands for write. When W=1, Q fluctuates based on CLK, much like D-Latch Q did previously.

According to the fusion of two selected bits (S1 and S0), the output designated by Omux may be given to any of the four inputs (C, D, E, and F), proving the accuracy of the architecture depicted in Figure 14.

4.1. Performance Metrics

We use four indicators to evaluate reversible FPGA performance. These measures are described briefly below:

- Gates: A circuit uses reversible gates. The total number of gates will be obtained by adding together all the gates. This is a reversible circuit metric. A circuit with fewer essential gates is more efficient.
- Garbage outputs: The unused outputs of a circuit are counted as garbage. We can get the total trash outputs by adding up all the garbage outputs. The low garbage output value shows the design's efficiency.
- Quantum Cost: It is a key performance parameter for reversible circuits. This is the number of NOT, Controlled-V +, and CNOT gates necessary to implement it.
- Delay: In reversible circuits, the critical path reversible gates are replaced with quantum gates, and the overall critical path delay defines the circuit latency. Each 1×1 or 2×2 RG has a unit-delay and is indicated by Δ.

4.2. Performance Analysis

Performance analysis of the different components of the

LUT Logic Block is given in this section.

Table 5. Fault tolerant (FT) master Slave Flip flop Output comparison

Techniques	No. of Gates	Unit Delay	Quantum Cost	Garbage Output	Transistors
Existing [3]	7	6	25	9	60
Existing [4]	5	5	21	7	38
Proposed	4	4	10	2	6

Table 6. FT 4x1 Multiplexer Output comparisons

Techniques	No. of Gates	Unit Delay	Quantum Cost	Garbage Output	Transistors
Existing [3]	9	7	57	11	70
Existing [4]	1	1	12	5	12
Proposed	1	1	5	3	5

Table 7. FT D-latch Output comparison

Techniques	No. of Gates	Unit Delay	Quantum Cost	Garbage Output	Transistors
Existing [3]	2	2	9	7	19
Existing [4]	1	1	6	4	10
Proposed	1	1	2	1	3

Table 8. 4 i/p LUT-FPGA Output comparison

Techniques	No. of Gates	Unit Delay	Quantum Cost	Garbage Output	Transistors
Existing [3]	39	39	243	50	342
Existing [4]	5	5	60	19	60
Proposed	5	5	56	5	37

Table 9. Comparative analysis of CLB

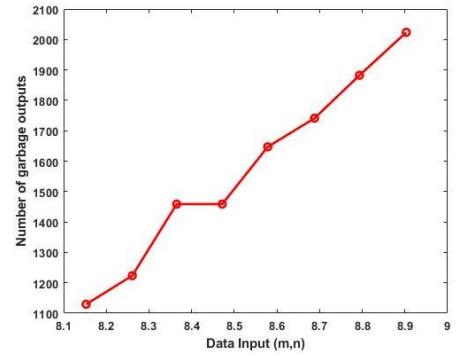
Techniques	No. of Gates	Unit Delay	Quantum Cost	Garbage Output	Transistors
Existing [3]	44	44	208	57	424
Existing [4]	18	18	170	57	192
Proposed	12	15	156	32	112

Tables 5–9 show that the suggested architecture performs better than existing designs in terms of gate count, transistor count, garbage output, latency, and quantum cost.

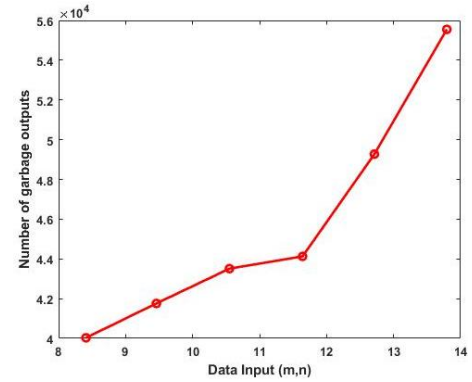
The FPGA structure is fixed by two variables n and m. For each of these ways, the overall results are displayed in comparison to the findings for the three approaches (i) Changing m without changing n (ii) Changing n while maintaining m. (iii) Changing m and n.

4.2.1. Garbage Outputs Comparison

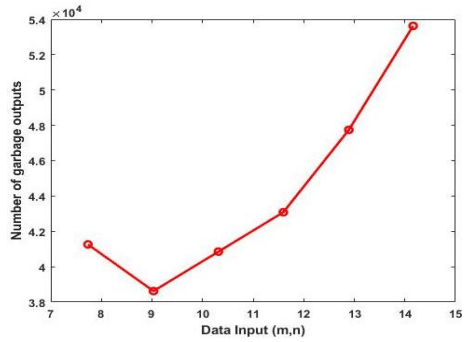
Figure 16 compares the proposed FPGA to existing FPGAs in terms of garbage outputs. The above-mentioned (i), (ii), and (iii) design techniques are presented in Figure 16(a)–(c). Figure 16 indicates that the suggested approach outperforms existing efforts in all scenarios. In the final two scenarios, existing designs rise exponentially, but the system design increases linearly with the number of inputs.



(a)



(b)

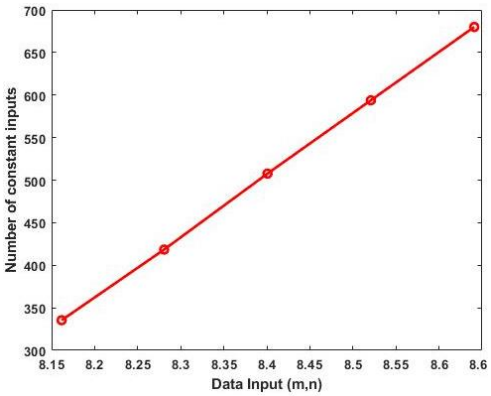


(c)

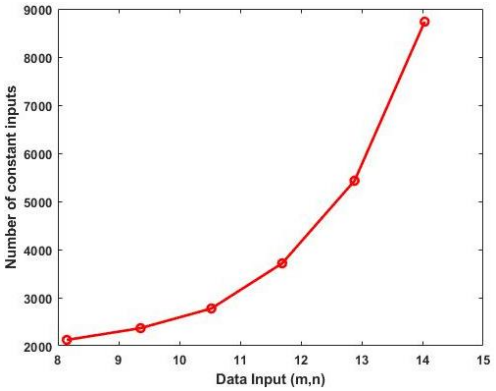
Fig. 16. Comparison of proposed with existing method on the garbage outputs (a) Changing m without changing n (b) Changing n while maintaining m. (c) Changing m and n

4.2.2. Constant Inputs Comparison

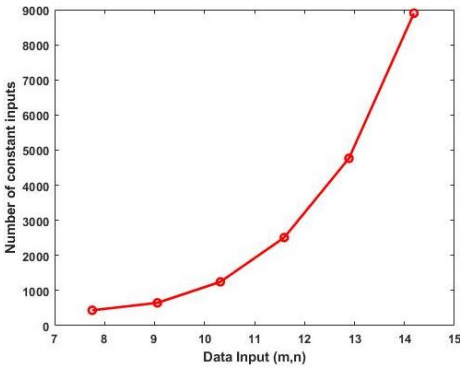
Quantum circuits with multiple constant bits and reversible gates are unmanageable. Constant input is therefore regarded a substantial overhead that must be reduced. Figure 17(a)–(c) compares the proposed FPGA to existing FPGAs in terms of necessary constant inputs for design methods I (ii), and (iii). The slope (growing rate) of all the edges in Figure 17 is about equal, but the recommended design has the greatest performance.



(a)



(b)

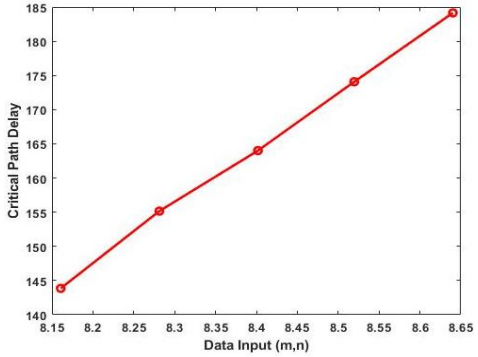


(c)

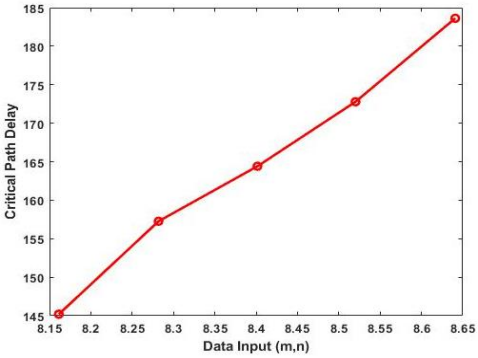
Fig. 17. Comparison of proposed with existing method on constant inputs (a) Changing m without changing n (b) Changing n while maintaining m. (c) Changing m and n

4.2.3. Delay Comparison

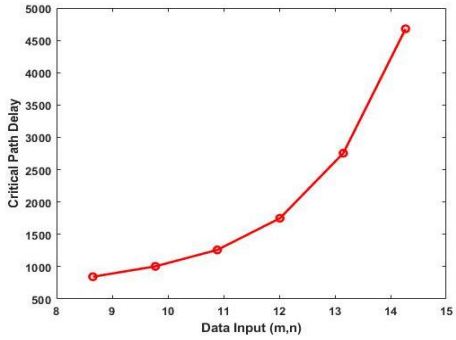
A logic circuit's delay is equal to the delay of the crucial circuit. However, finding all essential paths is an NP-complete task, especially for large circuits, or for large n or m in the case of FPGAs. Thus, researchers utilised to choose the route that was most likely to include critical paths. The proposed circuit's delay is analyzed in terms of the most likely critical pathways. Figure 18 compares the implemented work to the existing work in terms of circuit delay. According to Figure 18(a)-(c), the suggested design outperforms the existing designs under all feasible conditions.



(a)



(b)



(c)

Fig. 18. Comparison of proposed with existing method on the critical path delay (a) Changing m without changing n (b) Changing n while maintaining m. (c) Changing m and n

4.3. Additional Experiments

To examine power consumption and delay, a comparison is done between a RFT a LUT-FPGA, a RFT Multiplexer, a RFT Master-Slave Flip-Flop and RFT D-Latch. According to the following table, the most recent technique outperforms all other rapid multipliers in terms of the aforementioned criteria.

Table 9. Implementation in Spartan-3 FPGA for analysis of power and delay

Logic gates	Power consumption (nW)	Delay (ns)
D-Latch	1.35	15
MS-FF	0.91	12
Multiplexer	1.57	17
LUT-FPGA	1.23	14

All of the RFT mentioned above are implemented in a 90nm (Spartan -3 FPGA) device for the purpose of analyzing the parameters listed in Table 9.

5. Conclusion

The reversible fault tolerant Configurable logic block (CLB) based FPGA are developed. The reversible CLB are important building components of FPGA. Here we have validated the functioning of RFT CLB which comprises 3 – input, 4 – input LUT, 4:1 Multiplexer, D – Latch and Master slave Flip – flop. Our RFT CLB and its components are improved in terms of Garbage outputs, number of gates, Quantum cost and Number of transistors with comparison to conventional approaches. The described design employing the proposed reversible gate provides reduced complexity of hardware and consumption of power.

Acknowledgements

The authors would like to express their gratitude to B.G.S Institute of Technology, Adichunchanagiri University, B.G. Nagara, Mandya for all of their assistance and encouragement in carrying out this research and publishing this paper.

Author contributions

Ravi L.S is the principal author responsible for the study's conception and design, overseeing experimental procedures, conducting data analysis, and composing the manuscript. He adeptly executed data acquisition and analysis, generated graphical representations, and made substantial contributions to manuscript development. He was actively engaged in study design, offering invaluable insights during data interpretation, and precisely revising the manuscript. Dr. Naveen K.B served as the Research supervisor, providing critical assessment of the manuscript.

Conflicts of interest

The authors declare no conflict of interest.

References

[1] Mirizadeh, S. M. A., & Asghari, P. (2022). Fault-tolerant quantum reversible full adder/subtractor: Design and implementation. Optik, 253, 168543.

[2] Tomar, P., Bansod, P. P., Gurjar, R. C., & Jarwal, R. (2021, October). Realization of testable D Flip-Flop based on reversible logic. In 2021 5th International Conference on Information Systems and Computer Networks (ISCON) (pp. 1-6). IEEE.

[3] Md. Shamsujjoha, H.M.H. Babu, L.Jamal, "Design of a compact reversible fault tolerant field programmable gate array: A novel approach in reversible logic synthesis", Microelectron J., 2013

[4] Mubin Ul Haque, Zarrin Tasnim Sworna and Hafiz Md. Hasan BabuAn. Improved Design of a Reversible Fault Tolerant LUT-Based FPGA. 2016 29th International Conference on VLSI Design and 2016 15th International Conference on Embedded Systems (VLSID).IEEE

[5] Sasamal, T. N., Singh, A. K., & Ghanekar, U. (2018). Toward efficient design of reversible logic gates in quantum-dot cellular automata with power dissipation analysis. International Journal of Theoretical Physics, 57(4), 1167-1185.

[6] Nomani, T., & Mohsin, M. (2019, November). A novel approximate adder design methodology with single LUT delay for fault-tolerant FPGA-based systems. In 2019 Second International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT) (pp. 1-6). IEEE.

[7] Junsangsri, P., & Lombardi, F. (2018, October). Multiple Fault Detection in Nano Programmable Logic Arrays. In 2018 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT) (pp. 1-6). IEEE.

[8] Seyedi, S., Navimipour, N. J., & Otsuki, A. (2021). Design and Analysis of Fault-Tolerant 1: 2 Demultiplexer Using Quantum-Dot Cellular Automata Nano-Technology. Electronics, 10(21), 2565.

[9] Raj, M., Gopalakrishnan, L., Ko, S. B., Naganathan, N., & Ramasubramanian, N. (2020). Configurable Logic Blocks and Memory Blocks for Beyond-CMOS FPGA-Based Embedded Systems. IEEE Embedded Systems Letters, 12(4), 113-116.

[10] Nomani, T., & Mohsin, M. (2019, November). A novel approximate adder design methodology with single LUT delay for fault-tolerant FPGA-based systems. In 2019 Second International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT) (pp. 1-6). IEEE.

[11] Haque, M. U., Sworna, Z. T., Babu, H. M. H., & Biswas, A. K. (2018). A fast fpga-based bcd adder. Circuits, Systems, and Signal Processing, 37(10), 4384-4408.

[12] Tara, N. (2020). Design and Implementation of Reversible Programmable Devices (Doctoral dissertation, University of Dhaka).

[13] Gaur, H. M., Singh, A. K., Mohan, A., & Pradhan, D. K. (2019). Computational analysis and comparison of reversible gates for design and test of logic circuits. International Journal of Electronics, 106(11), 1679-1693.

[14] Tara, N., Babu, H. M. H., & Matin, N. (2016, January). "Logic Synthesis in Reversible PLA". In 2016 29th International Conference on VLSI Design and 2016 15th International Conference on Embedded Systems (VLSID) pp. 110-115. IEEE.

[15] De, D., Purkayastha, T., & Chattopadhyay, T. "Design of QCA based Programmable Logic Array using decoder". Microelectronics journal, 55 in 2016, pp.92-107.

[16] Sharma, P., Bhandari, A. S., & Singh, C. Design & Performance Analysis of 8-Bit Low Power Parity Preserving Carry-Look Ahead Adder. International Journal on Recent and Innovation Trends in Computing and Communication, 5(6), pp.655-664.

[17] A. Anjana, "Design of RS Flip-Flop using Reversible logic Gate", International Journal of Electronics & Communication, vol. 2, issue 4, (2014).

[18] Naveen K B and M N Sree Rangaraju. Energy Efficient QAM Modulation/Demodulation Architecture using Reversible Logic Gates International Journal of Scientific & Engineering Research, Volume 7, Issue 8, August-2016

[19] Naveen K B and Prashantha.N.C. Design and Synthesis of QPSK: A Subsystem Module of Digital Communication using Reversible Logic. International Journal of Computer Applications (0975 – 8887) Volume 74– No.10, July 2013.

[20] Karthik, P., & Kumar, A. P. (2017). "Realization of Programmable Logic Array using Compact Reversible Logic Gates".

[21] Khalil, K., Dey, B., Sherazi, Y., Kumar, A., & Bayoumi, M. (2020, October). A novel design reversible logic based configurable fault-tolerant embryonic hardware. In 2020 IEEE International Symposium on Circuits and Systems (ISCAS) (pp. 1-5). IEEE.