An Improved Design of a Reversible Fault Tolerant LUT-Based FPGA

Mubin Ul Haque*, Zarrin Tasnim Sworna† and Hafiz Md. Hasan Babu‡
*Department of Computer Science and Engineering, University of Dhaka, Dhaka-1000, Bangladesh
Email: *mubin10haque@gmail.com, †sworna07@yahoo.com, ‡hafizbabu@hotmail.com

***Abstract*—This paper presents the design of reversible fault tolerant architecture of logic elements of LUT (look-up table) based Field Programmable Gate Array (FPGA). The proposed logic elements are master slave Flip Flop, D-Latch and multiplexer. A new 4×4 and a new 6×6 fault tolerant reversible gates are proposed for designing efficient reversible fault tolerant D-latch, master slave Flip Flop and multiplexer, respectively. The design of the proposed logic elements achieve the improvement of 41.67% in terms of number of gates compared to the best known existing approach. Besides, the proposed logic elements outperform the best existing technology by 13.33% and 27.27% in terms of quantum cost and unit delay, respectively. Finally, the efficiency of the proposed elements are clarified by implementing an n-bit adder using the proposed Configurable Logic Block (CLB) of FPGA with 60.9% power savings and 23.56% delay minimization.**

Keywords: Reversible logic, Fault tolerant, Field Programmable Gate Array, Look-Up Table

I. Introduction

In the early 1960, R. Landaurer's research [1] showed that irreversible hardware computation regardless of its realization technique results in energy dissipation due to information loss. Each bit of information dissipates $k.T.ln2$ joules of energy where k is Boltzman constant, T is absolute temperature. In 1973, Benett showed that a circuit must be made using reversible logic gates to avoid this huge energy dissipation [2]. Fault tolerance is the property that enables a system to continue operating properly in the event of the failure of (one or more faults within) some of its components. Usually FPGA consists of an array of configurable logic block, interconnects and I/O blocks. FPGA can be configured as needed for each application. The same semiconductor technology advances that have brought processors to their performance limits have turned FPGAs from simple logic to highly capable programmable fabrics. Most popular logic blocks are Look-Up-Table (LUT) and Plessy logic block. With more inputs, a LUT can implement more logic using fewer logic blocks. Thus it helps in less routing area. Authors in paper [3] showed that, a 3 to 4 input LUT size results in better performance in area and delay. So we consider a generalized 4-input LUT based Logic block.

Three main contributions have been addressed in this paper:,

- Design of a Reversible Fault Tolerant (RFT) D-latch and Master Slave Flip Flop using our proposed gate targeting optimum number of gates, quantum cost and unit delay.
- A new RFT multiplexer has been proposed with lower quantum cost, unit delay and number of gates.
- An improved RFT LUT based CLB of FPGA is presented targeting lower number of gates, garbage, quantum cost and unit delay.

Finally, an n-bit adder implementation and its performance has substantiated the effectiveness of the components.

The organization of this paper is as follows, Section 2 provides prior knowledge of reversible logic design and FPGA with brief description of the related existing works. Improved architecture of reversible fault tolerant D-latch, master slave flip flop and 4 to 1 multiplexer are illustrated in Section 3. In Section 4, An n-bit adder implementation using the proposed components is elucidated to prove the efficiency of the components. Section 5 expounds the simulation result and performance analysis. Lastly, the paper is concluded in the last Section.

II. Basic Definitions and Existing Works

In this section, the basic definitions of reversible fault tolerant gate, garbage output, unit delay, LUT and quantum cost are presented. Besides, the recent related existing works are briefly described.

A **reversible gate** is an n input I_n, n output O_n (denoted by n x n) circuit that produces a unique output pattern for each possible input pattern as $I_n \leftrightarrow O_n$, where unused output is known as **garbage** output.

Fault tolerant gates are reversible gates which maintain parity between input and output vectors as $I_1 \oplus I_2 \oplus ... \oplus I_n$= $O_1 \oplus O_2 \oplus ... \oplus O_n$. In this paper, the fault tolerant gates are used to preserve the parity of the proposed circuits, where the fault tolerant gates are used to detect the faults of the circuits when the fault occurs.

The **quantum cost** can be derived by substituting the reversible gates of a circuit by a cascade of elementary quantum gates. Quantum Cost of a circuit is the minimum number of 2×2 unitary gates to represent the circuit keeping the output unchanged. The quantum cost of a 1×1 gate is zero and that of any 2×2 gate is one. In other words number of 2×2 Ex-OR, controlled-V (square root of NOT i.e. SRN) or controlled-V_+(hermitian of SRN) represent the quantum cost of a circuit.

For example, Fig. 1 shows the block diagram of two fault

978-1-4673-8700-2/16 $31.00 © 2016 IEEE
DOI 10.1109/VLSID.2016.39

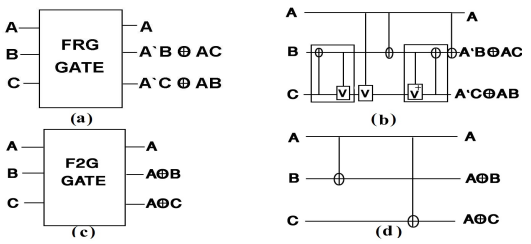

Fig. 1. Fredkin Gate (a) Block Diagram (b) Quantum Realization(Cost 5) Feynman Double Gate (c) Block Diagram (d) Quantum Realization(Cost 2)

tolerant reversible gates named Fredkin (FRG) and Feynman double gate (F2G) along with their quantum realization.

Delay represents the critical delay of the circuit which considers the following two assumptions. Firstly, each gate performs computation in unit time which means that every gate will take same amount of time for internal logic operations. Secondly, all the inputs are known to the circuit before the computation begins.

Power can be obtained from current using Microwind DSCH [7] and voltage across each transistor. We have considered the static power consumption of the circuit and power across each transistor is calculated using the following formula:

$$P(Power_i) = (V_i)(\text{voltage}) \times (I)(\text{current}).$$

where $Power_i$ represents power consumed by i number of transistors in a circuit and V_i symbolizes voltage required at each transistor.

For example, let us consider a half adder circuit which is composed of an AND gate and $Ex\text{-}OR$ gate. Using Microwind DSCH [7], critical path delay of each of the $Ex\text{-}OR$ and AND gate is $0.160\ ns$. So, delay of a half adder circuit is $0.160\ ns$ and unit delay of the circuit is one. On the other hand, an AND gate and $Ex\text{-}OR$ gate require 6 and 8 transistors, respectively. Assuming the voltage across each of the transistor is $1.5\ Volt$ and using Microwind DSCH [7], current passing through transistors is $0.1\ mA$. So, a half adder has a total of $(8 + 6) = 14$ transistors and power consumption is $(14 \times 0.1 \times 1.5) = 2.1\ mW$.

A **LUT (look-up table)** is a memory with output that essentially implements a truth table where each input combination generates a certain logic output [5]. An n-input LUT can be implemented with multiplexers and a memory cell where n selection inputs of the Mux are the input bits of look up table and input bits of the Mux are stored in a memory cell.

Authors in papers [3, 4, 6] proposed the improved reversible design of Plessy logic block of FPGA and on the other hand authors in paper [5] proposed a LUT based reversible FPGA. Among these works confined structure of LUT was not described in [5]. To design all the components of reversible Plessy logic block, authors in [4] used only reversible NH, BSP, FG and TG gates with Quantum cost of 4, 13, 1 and 5 respectively, whereas authors in [6] used reversible Mux and Peres gates and their quantum cost are

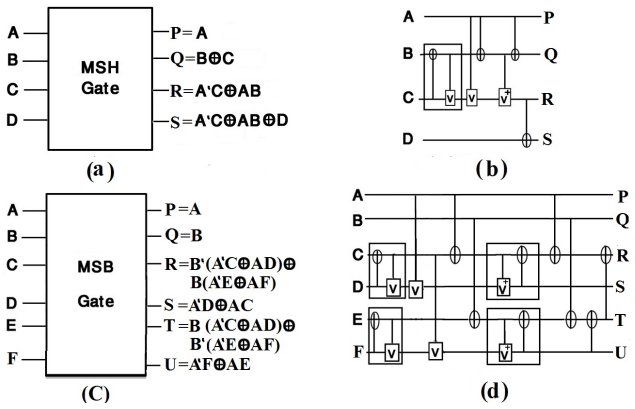

Fig. 2. MSH Gate:(a) Block Diagram (b) Quantum Realization (Cost 6) MSB Gate:(c) Block Diagram (d) Quantum Realization (Cost 12)

both 4. Authors in [3] used FRG and F2G gates whose quantum cost are 5 and 2, respectively. They did not improve significantly in terms of number of gates used and unit delay. Authors in paper [5] used BSP, FG, HNFG, FRG, TG gates to implement logic elements of reversible FPGA. But they did not show any generalize structure of look up table. The design is not optimized in terms of numbers of gates, garbage outputs and quantum costs. Moreover, they did not provide any generalized circuit for fault tolerant reversible FPGA. The theoretical explanations and algorithm of the design are also absent in [5].

III. Proposed Reversible Fault Tolerant logic elements of FPGA

This section illustrates the design layout, algorithms, theoretical properties and working principles of the proposed logic elements used in LUT based reversible fault tolerant FPGA.

A. Proposed Reversible Fault Tolerant Gates

We propose two new reversible fault tolerant gates along with their block diagram, quantum realization and truth tables.

1) MSH (Mubin-Sworna-Hasan) Gate: Fig. 2(a) and 2(b) represent the proposed MSH gate (input vector is [A,B,C, D] and output vector is [P, Q, R, S]) with its quantum realization.

2) MSB (Mubin-Sworna-Babu) Gate: Block diagram and quantum realization of the proposed MSB gate (input vector is [A,B,C, D, E, F] and output vector is [P, Q, R, S, T, U]) are represented in Fig. 2(c) and Fig. 2(d), respectively.

B. Proof of Fault Tolerance Property of Proposed Gates

Table I demonstrates the unique one to one correspondence and parity preservation between input (A, B, C and D) and output (P, Q, R and S) of the proposed MSH gate. For any input combination, corresponding output combination shows the same parity (that is A⊕B⊕C⊕D = P⊕Q⊕R⊕S) as well as reversibility of the proposed gate.

Similarly, Table II clarifies the fault tolerance property and the unique mapping between input (A, B, C, D, E and F) and output (P, Q, R, S, T and U) of the proposed MSB gate.

TABLE I
TRUTH TABLE OF PROPOSED MSH GATE

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

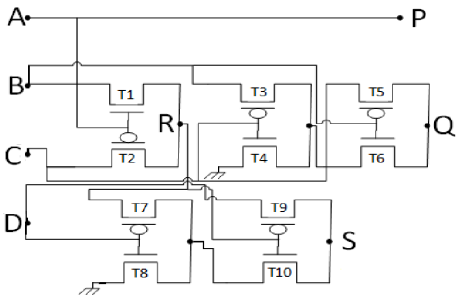

Fig. 3. Transistor Realization of the Proposed MSH Gate

C. Physical Implementation of Proposed Gates

Transistor realizations of the proposed MSH gate and MSB gate are demonstrated in Fig. 3 and Fig. 4 which require 10 and 12 transistors, respectively. Now, for MSH gate, suppose, when inputs of A=0, B=1, C=1 and D=0 then output P is 1, transistor T1 is off and transistor T2 is on, so C is propagated to R. Since C is 1, output at R is also 1. Again, both of B and C is 1, transistors T3 and T5 are off and T4 and T6 are on and pass the GND (Ground) voltage or 0 to Q. Similarly, R=1 and D=0, so transistors T7 and T10 are on, transistors T8 and T9 are off and output found at S is 1. So, finally for input combination (A=0, B=1, C=1 and D=0), output combination found is (P=0, Q=0, R=1 and S=1) as tabulated in Table I which proves the correctness of the physical implementation of the proposed MSH gate. On the other hand, let us consider MSB gate and for input combination (A=0, B=1, C=0, D=1, E=0 and F=1), transistors T2, T4, T5, T8, T10 and T11 are off and T1, T3, T6, T7, T9 and T12 are on. Input of D and F will be propagated to the output of S and U respectively and S=1 and U=1 since both D and F is 1. Again, transistors T3 and T9 will propagate the input of C and E to the transistors T6 and T12, respectively and outputs obtained at R and T are 0 since both of C and E is 0. Finally, the generated output is (P=0, Q=1, R=0, S=1, T=0 and U=1) for input combination (A=0, B=1, C=0, D=1, E=0 and F=1). It can be also shown for other input combinations as well to prove the correctness of the physical implementation of the proposed MSB gate.

TABLE II
TRUTH TABLE OF PROPOSED MSB GATE

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

D. Proposed Reversible Fault Tolerant D-latch, Master Slave Flip Flop and 4×1 Multiplexer

Fig. 5(a) presents the architecture of proposed reversible fault tolerant D-latch using one MSH gate to produce the desired output, S=$Clk'.feedback \oplus Clk.D$ with only two garbage G_1 and G_2 and one constant input. Fig. 5(b) illustrates the architecture of proposed reversible fault tolerant write enable master-slave flip flop using one FRG, two F2G and

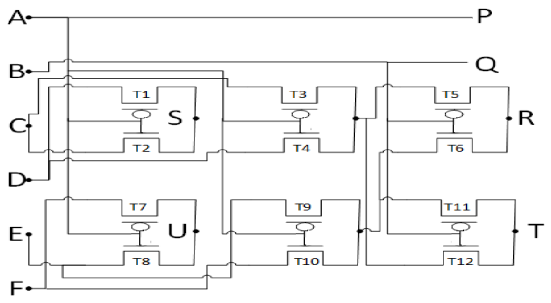

Fig. 4. Transistor Realization of the Proposed MSB Gate

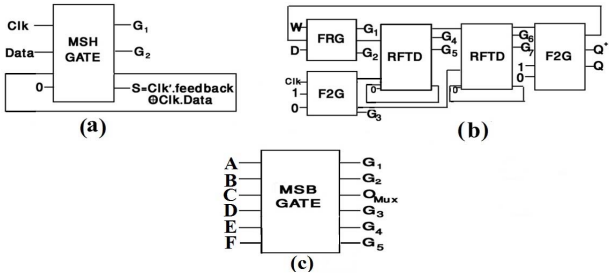

Fig. 5. Block Diagram of Proposed Reversible Fault Tolerant (a) D-latch (b) Master slave flip flop (c) 4×1 Multiplexer

two reversible fault tolerant D-latch (which described earlier) to produce the desired output, Q and Q^+.We use one proposed MSB gate to form a reversible fault tolerant 4×1 Mux having the Equation 1 and our proposed RFT multiplexer obtain the least garbage output which is clarified in Property 1. Block diagram is exhibited Fig. 5(c).

$$O_{MUX} = \overline{A}\,\overline{B}\,C + \overline{A}BE + A\overline{B}D + ABF \quad (1)$$

Property 1: A $4n$ to n RFT Multiplexer requires at least $(4n+1)$ garbage outputs and no constant inputs where n is the number of data bits.

Lemma 1: The proposed n-input reversible fault tolerant D-latch for an FPGA requires at least n reversible fault tolerant gates, $2n$ garbage and $6n$ quantum cost. ■

Lemma 2: The proposed n-input RFT write enable master slave flip-flop for an FPGA requires at least $5n$ RFT gates, $7n$ garbage and $21n$ quantum cost. ■

E. Proposed Reversible Fault Tolerant n-input Look Up Table

An 8×1 RFT multiplexer is proposed using two MSB and one FRG gates and an 16×1 RFT multiplexer using five MSB gates which are actually an 3-input and 4-input LUT as illustrated in Fig. 6(a) and Fig. 6(b), respectively. In Fig. 6, $1/0$ have been stored using our proposed D-latches. Similarly we can design an n-input LUT with our proposed architecture.

F. Proposed Reversible Fault Tolerant CLB of FPGA

We have designed the components of 4-LUT CLB of Xillinx FPGA [10] in reversible fault tolerant way. By accumulating

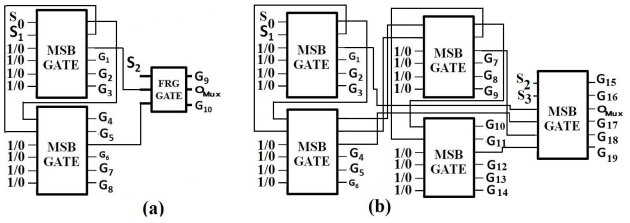

Fig. 6. Block Diagram of Proposed RFT (a) 3-input LUT (b) 4-input LUT

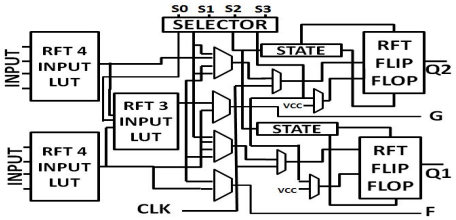

Fig. 7. Block Diagram of Proposed Reversible Fault Tolerant CLB

the components together, reversible fault tolerant FPGA is shown in Fig. 7 and Algorithm 1 is also provided for designing reversible fault tolerant CLB of FPGA.

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
```

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

We have used DSCH simulation tool [7] to implement a full adder circuit using our proposed reversible fault tolerant CLB architecture which is illustrated in Fig. 8 where Block 1 computes the sum $(A \oplus B \oplus C)$ of three input bits (A, B and C) and Block 2 computes the carry $(AB + AC + BC)$.

While full adder logic was assimilated in our architecture, we have computed the delay and power consumption. This implementation is scalable and hence can be generalized for n-bit adder is represented in Fig. 9.

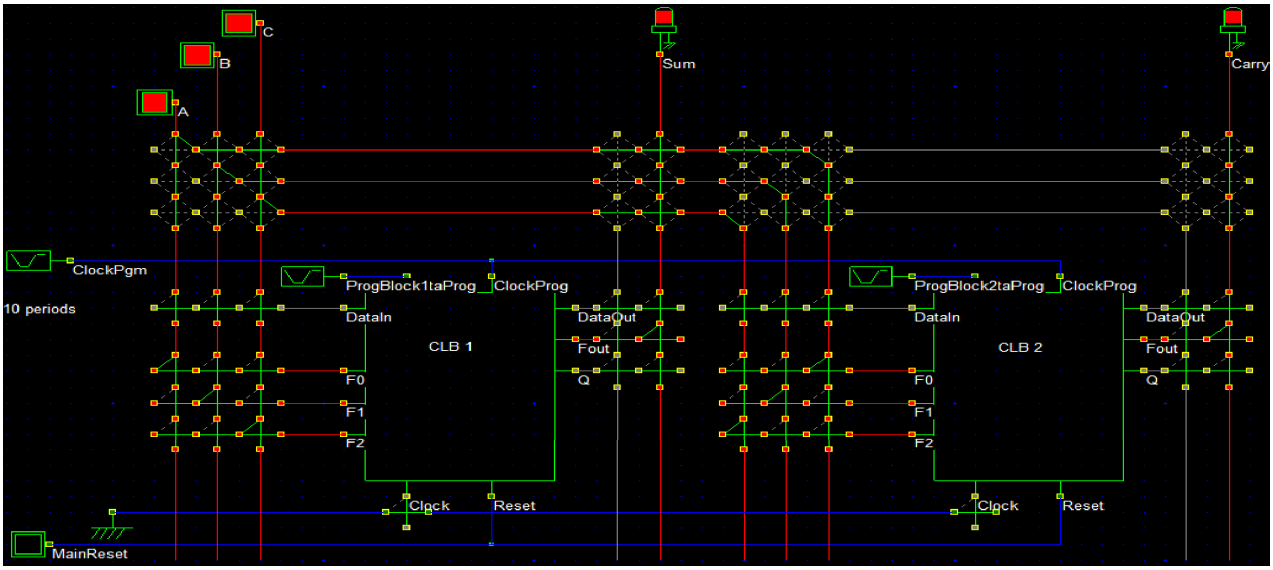

Fig. 8. Design of a Full Adder

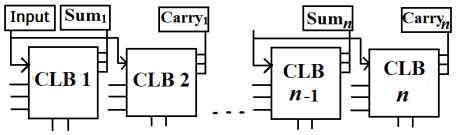

Fig. 9. Generalized Block Diagram for an n-Bit Adder Circuit

V. Simulation Results and Performance Analysis

The simulation of the proposed design is done using Microwind DSCH 2.7 software with technology node of 0.12 μm [7] using a computer with the Intel(R) Core (TM) i3 CPU with 2.10 GHz Clock Speed and 2.00 GB RAM. The simulation results show that the proposed circuits give correct outputs for all possible combinations of inputs which is shown in Fig. 10. Table III, Table IV, Table V, Table VI and Table VII demonstrate the comparative results of different reversible fault tolerant components of FPGA. The proposed reversible fault tolerant 3-input LUTs require 3 gates with 34 transistors, 29 quantum cost and 3 unit delay whereas the the best known existing [3] circuits require 7 gates with 70 transistors, 35 quantum cost and 7 unit delay. On the other hand, the proposed CLB requires 18 gates with 192 transistors and 208 quantum cost whereas the best known existing [3] CLB needs 44 number of gates with 424 transistors and 208 quantum cost. The Graphical analysis of power-delay product for the proposed and existing Adder is elucidated in Fig. 11 which interprets the enhancement of our proposed design. Finally, Table VIII depicts the superiority of our design in terms of power and delay.

VI. Future Work

The work can be extended to design the Block RAM logic, since after LUT Logic, Block RAMs are one of the most widely used blocks in FPGA.

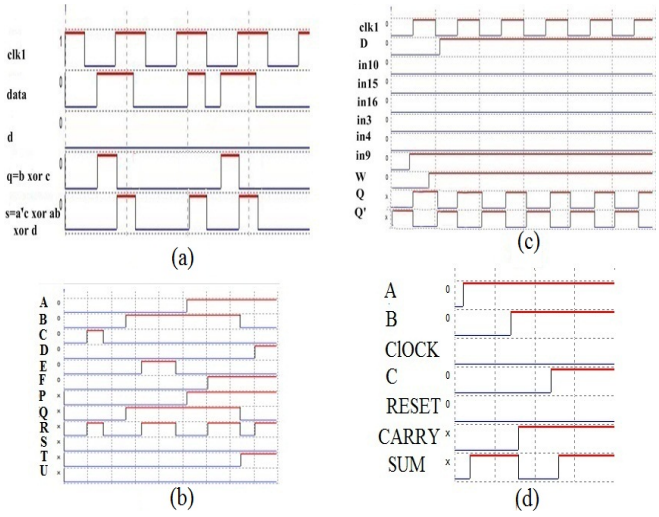

Fig. 10. Simulation Result of Proposed Reversible Fault Tolerant (a) D-Latch (b) Master-Slave Flip Flop (c) 4×1 Multiplexer (d) Full Adder Implemented by FPGA

TABLE III
COMPARATIVE ANALYSIS OF D-LATCHES

D-Latches	Number of Gates	Transistors	Quantum Cost	Unit Delay
Existing[3]	2	16	7	2
Existing[4]	2	19	9	2
Existing[5]	2	25	8	2
Existing[8]	1	22	9	1
Existing[9]	2	20	10	2
Proposed	1	10	6	1

TABLE IV
COMPARATIVE ANALYSIS OF FLIP-FLOPS

Flip-Flops	Number of Gates	Transistors	Garbage	Quantum Cost	Unit Delay
Existing[3]	7	54	7	23	6
Existing[4]	7	60	9	25	7
Existing[5]	7	72	9	25	7
Proposed	5	38	7	21	5

TABLE V
COMPARATIVE ANALYSIS OF 4×1 MULTIPLEXER

Multiplexers	Number of Gates	Transistors	Garbage	Quantum Cost	Unit Delay
Existing[3]	3	30	5	15	3
Existing[4]	9	78	11	57	6
Existing[5]	7	70	11	57	7
Proposed	1	12	5	12	1

TABLE VI
COMPARATIVE ANALYSIS OF LUTS

LUT SIZE	3-input LUT				
	Number of Gates	Transistors	Quantum Cost	Garbage	Unit Delay
Existing[3]	7	70	35	10	7
Existing[4]	19	166	119	24	19
Existing[5]	15	150	119	24	15
Proposed	3	34	29	10	3
LUT SIZE	4-input LUT				
Existing[3]	15	150	75	19	15
Existing[4]	39	342	243	50	39
Existing[5]	31	310	243	50	31
Proposed	5	60	60	19	5
LUT SIZE	5-input LUT				
Existing[3]	31	310	155	36	31
Existing[4]	79	694	491	102	79
Existing[5]	63	630	491	102	63
Proposed	11	120	125	36	11

TABLE VII
COMPARATIVE ANALYSIS OF CLB

Flip-Flops	Number of Gates	Transistors	Garbage	Quantum Cost	Unit Delay
Existing[3]	44	424	57	208	44
Existing[4]	104	610	133	630	104
Existing[5]	84	906	133	630	84
Proposed	18	192	57	170	18

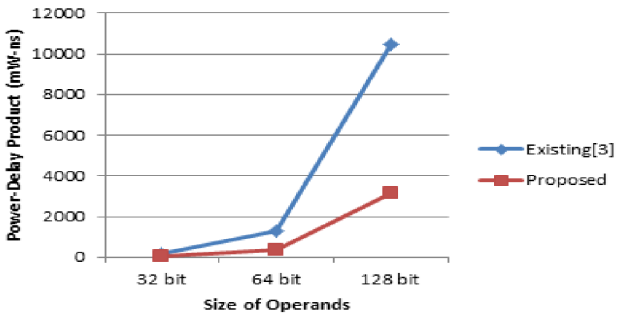

Fig. 11. Graphical Analysis of Power-Delay Product of RFT Adder

TABLE VIII
COMPARATIVE ANALYSIS OF RFT n-BIT ADDER

Adder	Power(mw)				
	1 bit	4 bit	16 bit	64 bit	n bit
Existing[3]	0.0082	0.0328	0.1312	0.5248	$0.0082n$
Proposed	0.0032	0.0128	0.0512	0.2048	$0.0032n$
Adder	Delay(ns)				
	1 bit	4 bit	16 bit	64 bit	n bit
Existing[3]	77.88	311.52	1246.08	4984.32	$77.88n$
Proposed	59.53	238.12	952.48	3809.92	$59.53n$

VII. Conclusion

This paper presented the design methodologies of reversible fault tolerant LUT (Look-Up Table) based configurable logic block (CLB) of FPGA (Field Programmable Gate Array). An algorithm for designing the reversible fault tolerant LUT based logic block for an FPGA had been proposed. In addition, two new reversible fault tolerant gates namely MSH (Mubin-Sworna-Hasan) and MSB (Mubin-Sworna-Babu) gate are presented. Finally, the designs of different components of the proposed FPGA using the MSH and MSB gates are elucidated. A design of an n-bit adder using the proposed LUT based FPGA had been also proposed. It is important to note that, for n-bits, the proposed FPGA based adder requires $0.0032n$ mW power and $59.53n$ ns delay, whereas the existing FPGA based adder requires $0.0082n$ mW power and $77.88n$ ns delay. Since FPGAs are useful in digital hardware design, the proposed reversible fault tolerant LUT based FPGA will be useful in many applications for designing low power circuits, such as complex arithmetic circuits [4,5].

References

[1] R. Landauer,"Irreversibility and heat generation in the computing process." *IBM J. Res. Dev.*, vol.5,no. 3, pp. 183-191, 1961.

[2] C.H. Bennett,"Logical reversibility of computation". *IBM J. Res. Dev.*, vol.17, no. 6, pp. 525-532, 1973.

[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] A.S.M. Sayem, M.M.A. Polash, H.M.H. Babu, "Design of a reversible logic block of field programmable gate array", *Silver Jubilee Conference on Communication Technologies and VLSI Design, VIT University, Vellore, India*, 2009.

[5] M.M.A. Polash, S. Sultana, "Design of a LUT-based reversible filed programmable gate array". *J.Comput.*, vol. 2, pp. 103-108, 2010.

[6] A.S.M. Sayem, S.K. Mitra, "Efficient approach to design low power reversible logic blocks for field programmable gate arrays." *IEEE international conference on Computer Science and Automation Engineering, Shanghai, China*, pp. 251-255, 2011

[7] The simulation software Microwind can be found at - *"http://microwind.net/download/Setup-Packages/MicrowindLite35.exe"*

[8] Ravi, T., S. Ranjith, and V. Kannan. "A Novel Design of D-Flip Flop Using New RR Fault Tolerant Reversible Logic Gate."*International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 2, February 2013)*

[9] Bruce, J. W., Mitchell A. Thornton, L. Shivakumaraiah, P. S. Kokate, and X. Li. *"Efficient adder circuits based on a conservative reversible logic gate."* In VLSI, 2002. Proceedings. IEEE Computer Society Annual Symposium on, pp. 74-79. IEEE, 2002.

[10] Xilinx 4000 series CLB url: *"www.xilinx.com/support/documentation/data-sheets/4000.pdf"*