Design of a Compact Fault Tolerant Adder/Subtractor Circuits Using Parity Preserving Reversible Gates

Ankur Sarker
Dept. of Computer Science and Engineering,
University of Dhaka, Dhaka-1000, Bangladesh, ankur.sum@gmail.com

Avishek Bose
Dept. of Computer Science and Engineering,
University of Dhaka, Dhaka-1000, Bangladesh
avishek.csedu.res@gmail.com

Shalini Gupta
Dept. of Computer Science and Engineering,
Dhaka City College, National University, Bangladesh
shalinigupta0001@gmail.com

***Abstract*— Reversible logic has drawn great attention in recent years due to its emerging propagation in diverse range of areas. In this paper, we present a novel approach to unite addition and subtraction operations; circuits that perform addition/subtraction operations using fault tolerant reversible gates with fault detection capability. Adder and subtractor are basic building blocks of any Arithmetic Logic Unit; in this manner we first present the concept of merging those two circuits into one logical block. Then we introduce all possible approaches to construct fault tolerant united addition-subtraction circuit for not only reducing the number of gate but also minimizing quantum cost and garbages of circuit at a meaningful level. We demonstrate three types of half-adder/subtractor circuits and four types of full-adder/subtractor circuits. Again, we depict an algorithm based on our novel concept and we also present simulations on our proposed circuits. Besides, the comparative analysis of our proposed compact method shows our proposed circuit outperform than existing circuit as highest improvements of proposed circuits are 33.33% for garbage output, 26.66% for quantum cost and 50% for gate count. Finally, overall significance of our proposed designs is presented in conclusion.**

Keywords*— Reversible logic, fault tolerant reversible circuits, fault tolerant gates, fault tolerant adder/subtractor circuit.

I. INTRODUCTION

Researchers like Landauer have shown that for irreversible logic computations, each bit of information loss generates $kT\times ln2$ joules of heat energy, where k is the Boltzmann's constant and T is the absolute temperature at which computation is performed [1]. Bennett showed that $kT\times ln2$ energy dissipation would not occur, if a computation is carried out in a reversible way [2], since the amount of energy dissipated in a system bears a direct relationship to the number of bits erased during computation. Reversible circuits are those circuits which do not lose any information. Reversible computation in a system can be performed only when the system comprises of reversible gates. These circuits can generate unique output vector from each input vector, and vice versa because of one to one mapping.

Checking the parity value of the input and output vectors is one of the most widely used methods for fault detection in digital system. It's most common use is to detect errors in storage or transmission of data whereas most of arithmetic and logic operations do not preserve the parity value of its input and output vectors. If arithmetic and logical operations are performed in such a way that it preserves the parity value from input to output throughout the computation, no intermediate checking is required to find faulty behavior. Such results can be forwarded to subsequent modules on the data path, while they are checked in a manner that is off the computation's critical path, and should therefore not subject to stringent performance or reversibility requirements. Any erroneous result tends to propagate through the downstream modules without a danger of corrupting additional information in the absence of multiple compensating faults.

Addition and subtraction operations are fundamental building block. Several reversible adder and subtractor circuits have been proposed with minimum number of quantum cost, gate count and constant input in [6], [7], [13] and [15]. Our study emphasizes on construction of viable adder/subtractor circuits in terms of cost and latency, which consist of fault tolerant reversible gates with fault detection mechanism.

This paper organizes as follows: Section II presents the previous concept of reversible logic and depicts some of the popular reversible gates, Section III shows the parity preserving property of fault tolerant gate to detect fault. Section IV, illustrates the one and only existing research based on our proposed topic. Section V presents our proposed methods of fault tolerant full adder/subtractor circuit as well as half adder/subtractor briefly along with an algorithm and simulation results. Next, the efficiency of our proposed methods is verified by a comparative study which is presented in Section VI. Finally, Section VII depicts conclusion to draw the future plan based on our current research.

II. BACKGROUND STDUIES

In this section, we present the basic definitions and properties which are related to reversible logic.

Definition 2.1. Let the input vector be I_V and output vector be O_V, where $I_V = (I_1, I_2, ..., I_n)$, $O_V = (O_1, O_2, ..., O_n)$ and $I_V \leftrightarrow O_V$. A $n\times n$ reversible gate is n-inputs and n-outputs circuit that produces a unique output pattern for each possible input pattern [6].

Example 2.1. Fig. 1 shows a $n\times n$ reversible gate. Similarly, 2×2 Feynman gate [shown in Fig. 2] is also a 2×2 reversible gate with two inputs and two outputs [8].

Definition 2.2. Unused outputs of a reversible gate are known as garbage outputs. These are only used to maintain the reversibility.

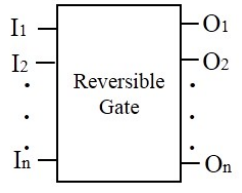

Fig. 1. $n\times n$ Reversible Gate.

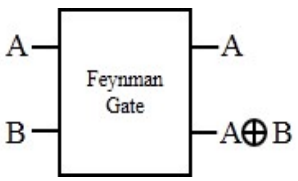

Fig. 2. 2×2 Feynman Gate.

978-1-4799-6288-4/14/$31.00 ©2014 IEEE

Example 2.2. When a Feynman gate is used for *EX-OR* operation of two inputs, extra one output is generated at the output part of the Feynman gate in addition to the *EX-OR* output. One extra output is known as garbage output [5].

*Definition 2.3.*Quantum cost refers to the cost of the circuit in terms of primitive gates such as C-NOT, V and V+ in quantum logic. It realizes the circuit by primitive gates and number of required gates is equal to the cost of that circuit.

Example 2.3. Quantum cost of Fenyman gate is 2, Toffoli gate [3] is 5.

*Definition 2.4.*Quantum depth of any logical circuit is cost of highest path from input variable to output variable.
Example 2.4. For several outputs, the quantum depth is equal to the quantum cost of longest path from outputs to inputs [14].

III. Fault Tolerant Reversible Gate

Fault tolerant reversible gates are capable of detecting fault by preserving the parity values of input and output vector So that, the parity values of input vector and output vector are same. Parity preserving property is one of the error detection methods in digital logic systems. Gates which are parity preserving must satisfy the equation: $A \oplus B \oplus C = P \oplus Q \oplus R$; where *A, B, C* are inputs and *P, Q, R* are outputs [12]. Fredkin gate [3], Feynman Double gate (F2G) [11] and MIG gate [14] are popular fault tolerant gates because they preserves input and output parity.

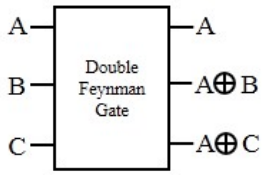

Fig. 3. 3×3 Feynman Double Gate.

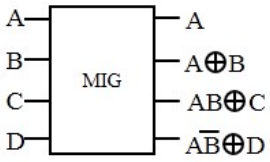

Fig. 4. 4×4 MIG Gate.

Feynman Double gate is a *3×3* reversible gate as shown in Fig. 3. It would be shown as: $I_v = (A, B, C)$ and $O_v = (P=A, Q=A \oplus B, R=A \oplus C)$; where I_v and O_v are input and output vectors, respectively. Here the control input *'A'* defines a second C-NOT operation [11]. It has the quantum cost of 2.
Also, in Fig. 4, the Modified IG (MIG) gate [14] is depicted, MIG gate can be defined as: $I_v = (A, B, C, D)$, $O_v = (P=A, Q=A \oplus B, R=AB \oplus C, S=A\bar{B} \oplus D)$, here I_v and O_v are input and output vectors, respectively. Quantum cost of MIG gate is 7.

IV. Existing Method for Fault Tolerant Adder Subtructor Circuit

A. Existing Fault Tolerant Full Addition-Subtraction Circuit

The one and only fault tolerant combined adder/subtractor circuit that we have found is proposed by P. Kaur [7]. This design requires high performance metrics such as more number of gates, high quantum cost, more garbage outputs and more constant input than our proposed designs.

V. Proposed Methods

The concept of merging addition and subtraction operations are presented in this section. This section also presents two fault tolerant reversible gates and our proposed fault tolerant adder/subtractor circuits with appropriate figures, an algorithm and simulation results using software.

A. Basic Concept

This subsection depicts the basic concept of addition and subtraction to construct the fault tolerant circuit adder/subtractor circuit.

a) A half-adder circuit performs addition operation of A and B; where, A, B are input variables thus, equations of the summation and carry are as follows: Sum = A⊕B, Carry = AB. Similarly, full-adder circuit performs addition operation of A, B and C; where these three are input variables so, the equations of summation and carry using full-adder circuit are as follows:

$Sum=A \oplus B \oplus C$, $Carry=AB \oplus AC \oplus BC=C(A \oplus B) \oplus AB$.

b) Unlike adder circuit, a half-subtractor circuit performs subtraction operation of *A* and *B;* where, *A,* B are input variables. So, equations of borrow and difference are as follows: $Diff=A \oplus B$, $Borrow=\bar{A}B$. Similarly, full-subtractor circuit performs subtraction operation of *A, B* and *C;* where *A, B* and *C* are input vectors. Thus, the equations of borrow and difference are as follows: $Diff=A \oplus B \oplus C$, $Borrow=\bar{A}B \oplus \bar{A}C \oplus BC=C\overline{(A \oplus B)} \oplus \bar{A}B$.

B. Novel Fault Tolerant Reversible Gates

a) Using the concept of previous section, we design two fault tolerant reversible gates to perform addition and subtraction operations in a single gate level representation. The first one is *5×5* fault tolerant reversible HASFT gate where $I_v = (S, A, B, C, D)$, $O_v = (P=A, Q=D \oplus B(A \oplus C), R=B \oplus C, S=D \oplus B\overline{(A \oplus C)}, T=B \oplus D \oplus E)$. The block diagram and quantum representation of this gate is shown in Fig. 5(a) and 5(b) respectively. The quantum cost for this gate is 10.

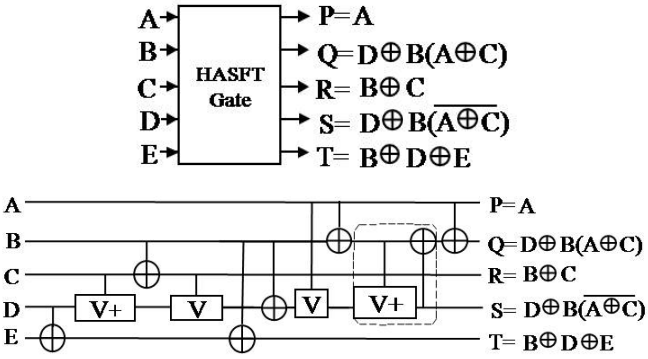

Fig. 5 (a), (b). Novel Fault Tolerant *5×5* HASFT Gate and its Quantum Representation respectively.

Next, we design another fault tolerant reversible *5×5* FASFT gate where where $I_v = (A, B, C, D, E)$, $O_v = (P=A, Q=B, R=A \oplus B \oplus C, S=A(B \oplus C) \oplus BC \oplus D, T=A\overline{(B \oplus C)} \oplus B\bar{C} \oplus E)$. The block diagram and quantum representation are presented in Fig. 6(a) and 6(b) respectively. The quantum cost of this gate is 10.

978-1-4799-6288-4/14/$31.00 ©2014 IEEE

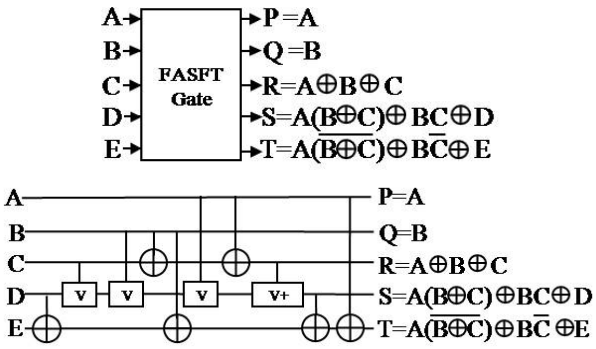

Fig. 6 (a), (b). Novel Fault Toleant 5×5 FASFT Gate and its Quantum Representation respectively.

Lemma 1: HASFT Gate is a parity preserving or fault tolerant reversible gate.

Proof. The input and output vectors of HASFT gate are as follows: $I_v = (A, B, C, D, E)$, $O_v = (P=A, Q=D\oplus B(A\oplus C), R=B\oplus C, S=D\oplus B(\overline{A\oplus C}), T=B\oplus D\oplus E)$, respectively.

Input parity HASFT gate is = A⊕B⊕C⊕D⊕E……. (i).

Output parity of HASFT gate is

= A⊕D⊕B(A⊕C)⊕B⊕C⊕D⊕B$\overline{(A\oplus C)}$⊕B⊕D⊕E

=A⊕B(A⊕C)⊕C⊕B(A⊕C⊕1)⊕D⊕E

=A⊕B(A⊕C)⊕C⊕B(A⊕C)⊕B⊕D⊕E

=A⊕(A⊕C)(B⊕B)⊕C⊕B⊕D⊕E

=A⊕B⊕C⊕D⊕E,

which is equal equation (i), the input parity. Thus, from the above explanations, it is proved that *HASFT* maintains the parity value from input vector to output vector. Next, using the truth table of *HASFT* Gate, it's easy to state that *HASFT* gate maintains the reversibility between input and output vectors.

Lemma 2: FASFT Gate is a parity preserving or fault tolerant reversible gate.

Proof. The input and output vectors of *FASFT* are as follows: $I_v = (A, B, C, D, E)$, $O_v = (P=A, Q=B, R=A\oplus B\oplus C, S=A(B\oplus C)\oplus BC\oplus D, T=A(\overline{B\oplus C})\oplus B\bar{C}\oplus E)$.

Input parity FASFT gate is = A⊕B⊕C⊕D⊕E……(i)

Output parity of FASFT gate is

= A⊕B⊕A⊕B⊕C⊕A(B⊕C)⊕BC⊕D⊕A$\overline{(B\oplus C)}$⊕ $B\bar{C}$⊕E

= C⊕A(B⊕C)⊕BC⊕D⊕A$\overline{(B\oplus C)}$⊕ $B\bar{C}$⊕E

= C⊕A(B⊕C)⊕BC⊕D⊕A(B⊕C⊕1)⊕ $B\bar{C}$⊕E

= C⊕A(B⊕C)⊕BC⊕D⊕A(B⊕C)⊕A⊕ $B\bar{C}$⊕E

= (B⊕C)(A⊕A)⊕B(C⊕$\bar{C}$)⊕C⊕D⊕A⊕E

= B⊕C⊕D⊕A⊕E

= A⊕B⊕C⊕D⊕E which is equal to equation (i).

Thus, it is proved that *FASFT* gate is parity preserving. Now, the truth table of FASFT gate ensures the reversibility of it. Thus, it is fault tolerant reversible gate.

C. Fault Tolerant Half Addition-Subtraction Circuits

From discussion of Section 4.A on the addition and subtraction operations, we can easily merge those two, (as only output *Carry* and output *Borrow* are different) and construct a combined circuit which can perform the addition and subtraction operations using the same architecture. If fault tolerance is applied in the combined circuit, then it preserves the parity value of the input and output vectors

We construct Design I, a fault tolerant half-adder/subtractor circuit using one Feynman Double gate and two Fredkin gates where Feynman Double gate is used to produce *Sum/Diff* and Fredkin gates are used to produce *Carry/Borrow* signal, quantum cost of this circuit is 12 which is also shown in Fig. 7(a) and 7(b) respectively with its equivalent quantum representation. This circuit requires fewer numbers of constant input, garbage output, and delay than previous one. And the Design II, another construction of half-adder/subtractor circuit consists of only two gates (MIG gate and Fredkin gate)is shown in Fig. 8 with quantum representation of this combination, it has same numbers of constant input, garbage output like previous design but delay is reduced significantly. Quantum cost of this circuit is also 12. Here, MIG gate produces all outputs and Fredkin gate works as a switch which is controlled by *Add/Sub* signal.

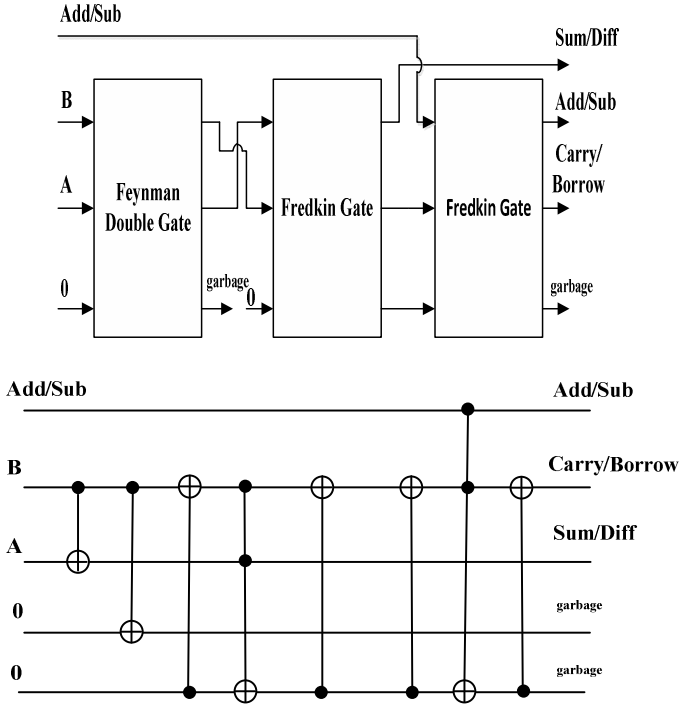

Fig. 7 (a), (b). (Design I) Fault Tolerant Half-Adder/Subtractor Circuit using Reversible Gates and its Quantum Representation.

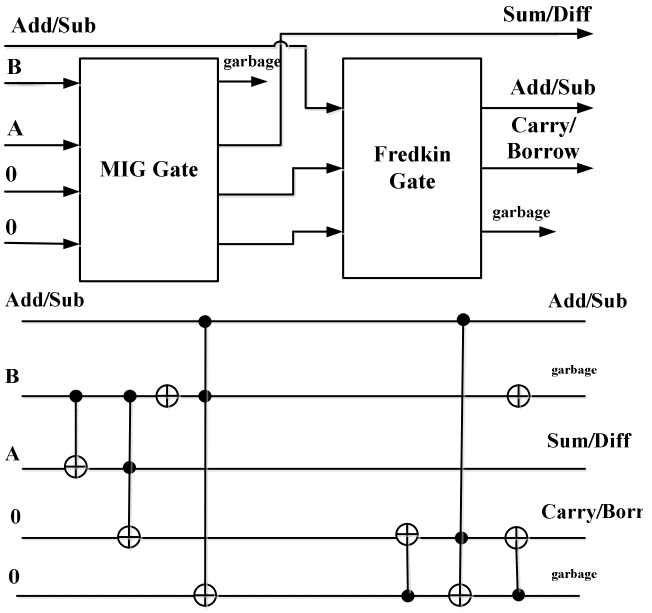

Fig. 8. (Design II) Fault Tolerant Half-Adder/Subtractor Circuit using Reversible Gates and Its Quantum Representation.

978-1-4799-6288-4/14/$31.00 ©2014 IEEE

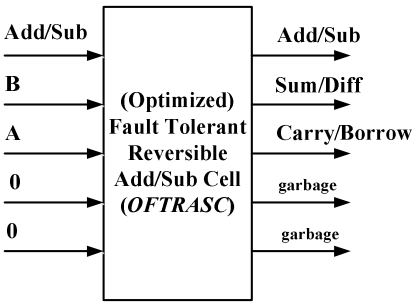

Fig. 9. (Optimized) Fault Tolerant Reversible Add/Sub Cell.

As, the numbers of inputs and outputs of Design I and Design II are same; we can use same representation for those two circuits. Fig. 9. shows the general representation of those two half-adder/subtractor circuits which is termed as *(Optimized) Fault Tolerant Reversible Add/Sub Cell (OFTRASC)*.

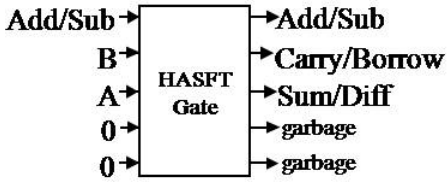

Fig. 10. (Design III) Fault-Tolerant Half-Adder/Subtractor Circuit using Newly Design 5×5 HASFT Gate.

Finally, we present our newly design using 5×5 HASFT gate as half-adder/subtractor circuit where two constant inputs are used with given three inputs and it produces three outputs *Add/Sub*, *Carry/Borrow* and *Sum/Diff* with two garbages. Fig 10. shows the Design III, the final fault tolerant half-adder/subtractor circuit.

D. Fault Tolerant Full Addition-Subtraction Circuits

Using the previously described half-adder/subtrator circuits (Design I, II and III), we can easily construct fault tolerant full-adder/subtractor circuits. In first approach to construct a full-adder/subtractor, we use two *(Optimized) Fault Tolerant Reversible Add/Sub Cells (OFTRASC)* with one additional Feynman Double gate to produce the final *Carry/Borrow*. As this design approach is presented using two previously described method (I and II), we have considered it as design IV & V. The quantum cost of this represented circuit is 26. Fig. 11 shows this circuit representation. In this circuit construction, Feynman Double gate is used to produce the full adder/subtractor's *Carry/Borrow* signal using *Carry/Borrow* signal of each individual *OFTRASC*.

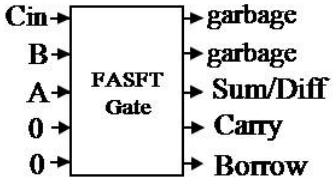

Fig. 13. FASFT gate as Fault-Tolerant Full-adder/subtractor Circuit Without Select Input *Add/Sub*.

The Design VI uses two HASFT gates with one additional Feynman Double gate; it means the quantum cost of this circuit is 22. Fig. 12 shows this circuit representation. Next, in Design VII, we use our fault tolerant FASFT gate without select input *Add/Sub*, as a result the quantum cost of this circuit is 10. Fig 13 shows that FASFT gate is used as a full adder. Also, Design VII shows another fault-tolerant full-adder/subtractor circuit where one FASFT gate and one Fredkin gate are used where quantum cost is 15, two constant inputs and three garbage outputs. Fig. 14 shows this circuit representation. Finally, Fig. 15 shows a parallel *n*-bit binary adder/subtractor circuit where any kind of proposed full-adder/subtractor circuits would be used which reduces quantum cost, delay and others at a significant level.

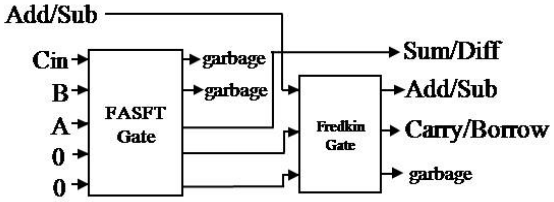

Fig. 14. (Design VII) Fault-Tolerant Full-adder/subtractor Circuit with Select Input *Add/Sub* by Using one FASFT Gate and one Fredkin Gate.

E. An Algorithm for Proposed Full-Addition/Subtraction Operation with Fault Detection

This algorithm presents the procedures for performing addition and subtraction operations with fault detection using our proposed approaches. The run time complexity of this algorithm is *O(n)* if *n* is equal to bit length of input operands.

ALGORITHM 1. Full Addition/Subtraction Operation with Fault Detection

Input: Given control signal- Add/Sub, given three inputs- $A_1, A_2 \ldots A_n$; $B_1, B_2 \ldots B_n$; $C_1, C_2 \ldots C_n$.
Output: Summation/difference between given inputs- $Sum/Diff_1, Sum/Diff_2 \ldots Sum/Diff_n$; carry/borrow out of given inputs- $Carry/Borrow_1, Carry/Borrow_2 \ldots Carry/Borrow_n$; parity values of input and output- $Parity_1, Parity_2$.

```
begin
  Parity1←Parity2← 0;
  J← 1;
  repeat
    Parity1 ← Exclusive-or of all inputs ⊕Parity1;
    Partial Sum/Diff ← Aj⊕Bj;
    if(Add/Sub =1)
      then
        Partial Carry/Borrow1 ←AjBj;
        Partial Carry/Borrow2 ← PartialSum/DiffCj;
    else
      Partial Carry/Borrow1 ←AjBj;
      Partial Carry/Borrow2 ← Partial Sum/ DiffCj;
  end
  Sum/Diffj ← Partial Sum/Diff⊕Cj;
  Carry/Borrowj ← Partial Carry/Borrow1⊕Partial Car ry/Borrow2;
  Parity2 ← Exclusive-or of all outputs ⊕Parity2;
  until j ≤ n;
  if (Parity1=Parity2)
  then
    "OK";
  else
    "Error Occurred";
End
```

F. Simulation Result

In order to verify the functional correctness of our proposed design approaches, we simulate using *Java* programming language. The input and output are represented by integer data structures and all reversible fault tolerant gates are represented

by different classes. As like for Fredkin gate, we construct *Fredkin Gate* class, which would be shown as like Fig. 16. Through the assignment of different objects, it can realize the connections between different reversible gates, which make the data in the program, are same as in the actual circuits. The simulated results from the program are as like as the results in original circuits. So, the simulation result can be the base of testing the validation of circuits.

```
class FredkinGate {
    int A, B, C, P, Q, R;
    void output() {
        P = A;
        Q = ((~A) & B) ^ (A & C);
        R = ((~A) & C) ^ (A & B);

    }
}
class FeynmanDoubleGate {
    int A, B, C, P, Q, R;
    void output() {
        P = A;
        Q = A ^ B;
        R = A ^ C;
    }
}
class MIGGate {
    int A, B, C, D, P, Q, R, S;
    void output() {
        P = A;
        Q = A ^ B;
        R = (A & B) ^ C;
        S = (A & (~B)) ^ D;
    }
```

Fig. 16. Code Level Representation of *Fredkin*, *Feynman* and *MIG* Gates.

Fig. 17 shows the simulation result of addition and subtraction operations for all proposed designs, where which operation (addition or subtraction) should be performed is controlled by *Add/Sub* input signal. Fig. 18 shows the simulation result of 4-bit full-addition operation implemented using Design VI. First, each 4-bit input is taken in string data structure format and then converted into single bit input and performed single bit addition operation and *Carry* signal is forwarded to the next block. Finally, each full-adder's outputs are cascaded and print result into string data structure as like input signals.

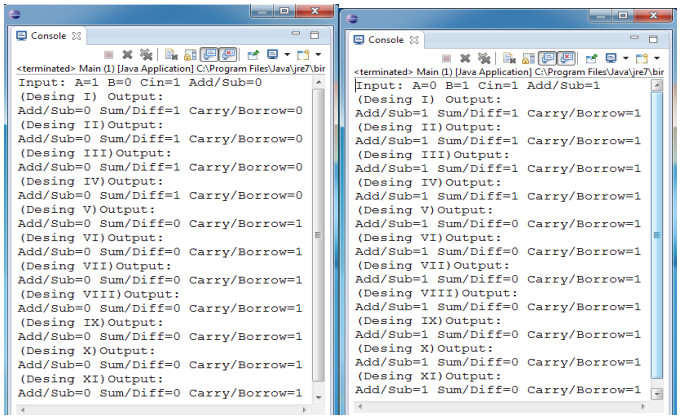

Fig. 17. Performing Addition and Subtraction Operation by Different Proposed Designs.

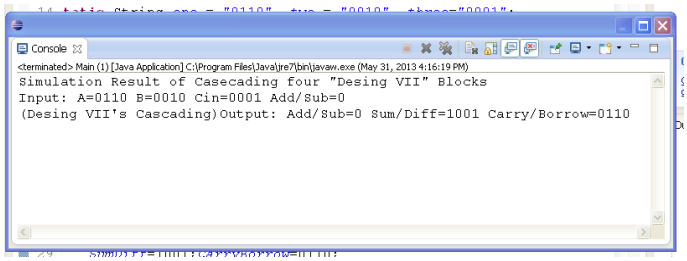

Fig. 18. 4-bit full-addition operation by cascading four Design VII blocks.

We also simulated our proposed fault tolerant or parity preserving reversible gates and methods using Microwind DSCH 3.5 software (*http://www.microwind.net/dsch.php*) on a computer which has the Intel(R) Core(TM) i3-2310M CPU with 2.10GHz Clock Speed and 2.00 GB RAM. Fig. 19 shows the simulation result of Design III (half adder/subtractor using HASFT gate) and Fig. 20 shows the simulation results of Design VII(full adder/subtractor using FASFT gate), which ensure the correctness of our proposed methodologies.

G. Theorem

Theorem 5.1. Our proposed designs could perform both addition and subtraction operations using same architecture.

Proof. Our proposed methods consist of several reversible gates, two numbers which should be used for desired operation and one control signal- *Add/Sub*. First, consider half-addition-subtraction operation. When *Add/Sub* is low, Design III perform $A \oplus B$ and AB produces output; Design I and Design II both use Fredkin gate as controlled switch and fed out only $A \oplus B$ and AB. On the other hand, when *Add/Sub* is high, Design III produce $A \oplus B$ and $\overline{A}B$; Design I and Design II both fed out only $A \oplus B$ and $\overline{A}B$. Second, for full-addition/subtraction operation, only one additional Ex-Or gate along with two half adder/subtractor circuits is enough to perform desired operation, either addition operation or subtraction operation. *Add/Sub* signal indeed is used as controller to dominate which operations should be performed. This verifies the fulfillment of our proposed methods. □

VI. Comparative Analysis

We propose three types of half-adder/subtractor circuits and four types of full-adder/subtractor circuits using different combinations of fault tolerant reversible gates with the realization concept of combined addition/subtraction operation. We summarize our proposed fault tolerant adder/subtractor circuits in Table 1. Next, the comparative study with existing design [7], which is shown in Table 2 shows that our design methodologies are scalable and outperforms than the existing design procedures.

Table 1. Summary of Proposed Fault Tolerant Adder/Subtractor.

Proposed Circuits	Constant Input	Garbage Output	Quantum Cost	Gate Count	Quantum Depth	Total Logical Operations □ □
Design I	2	2	12	3	12	8α+6β+4μ
Design II	2	2	12	2	12	6α+5β+3μ
Design III	2	2	10	1	10	2α+7β+μ
Design IV	5	6	26	7	26	16α+14β+8μ
Design V	5	6	26	5	26	12α+12β+6μ
Design VI	4	5	22	3	22	4α+16β+2μ
Design VII	2	3	15	2	15	8α+10β+4μ

□ It represents total number of fault tolerant logical operations; α, β and μ represents *logical And*, *logical Ex-Or* and *logical Not* Operations, respectively.

In Table 2, the comparison with existing circuit [7] is presented, where we mark out efficiency of our proposed compact design. Suppose we use our proposed designs to construct a 4-bit parallel adder/subtractor. It verifies the overall improvements of our proposed designs.

Table 2. The Comparison with Existing Fault Tolerant Full-Adder/Subtractor Circuits[7] and Our Compact Adder/Subtractor Circuit.

Fault Tolerant Design	Constant Input	Garbage Output	Quantum Cost	Gate Count
Existing [15]	2	4	18	3
Existing [7]	2	4	19	3
Design VII	2	3	15	2
Improvements	0%	33.33%	26.66%	50%

For considering quantum cost, Design VII outperforms than all other fault-tolerant full-adder/subtractor circuits.

VII. CONCLUSIONS

We have presented the unification concept of addition and subtraction operations and proposed a novel approach to design combined adder/subtractor circuits using fault tolerant reversible logic gates. We have also designed two novel fault tolerant reversible gates with lowest possible complexity to perform addition and subtraction operations. These proposed designs could perform addition and subtraction operations using same architecture. First, we have designed three fault tolerant reversible half-adder/subtractor circuits. Then, we have depicted four fault tolerant full-adder/subtractor circuits. Our proposed circuits achieve the considerable unit reduction for considering quantum cost, constant input, garbage output, delay and others. For considering *n*-bit full-adder/subtractor circuit, the existing design requires 9*n* constant inputs, 11*n* garbage outputs, 30*n* quantum cost, 9*n* gate count and 9*n* delay. However, one of our designs requires 2*n* constant inputs, 3*n* garbage outputs, 15*n* quantum cost, 2*n* gate count and 2n delay. It means our design procedures are scalable and outperformer than existing design. We have also verified our proposed design's correctness by software simulation and then we have compared our designs with existing research on same kind of fault tolerant/parity preserving circuit. As reversible logic is applicable to diverse range of areas including nano-scale computing, low power computing modules and quantum computing,DNA computing, etc., our proposed design methodology would also be extensible in those fields.

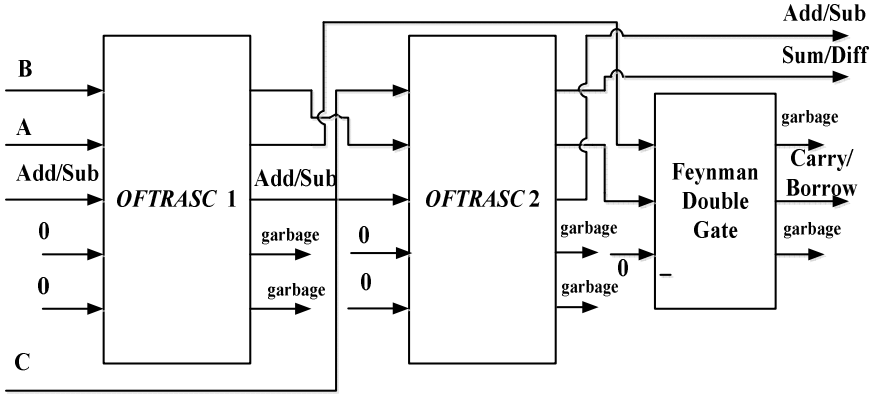

Fig. 11. (Design IV & V) Fault tolerant full-adder/subtractor circuit using *OFTRASC*s.

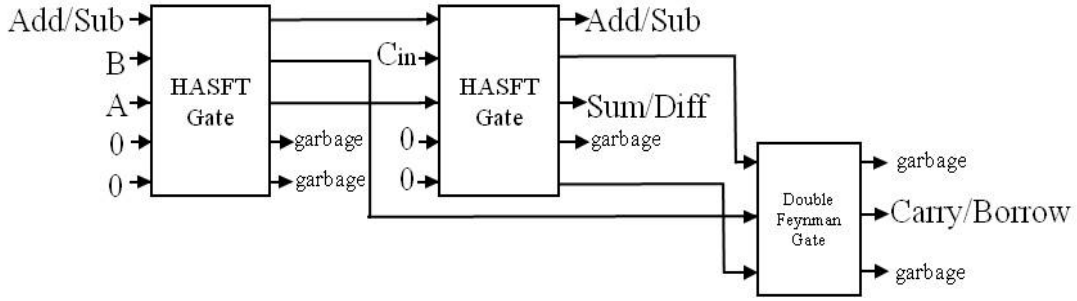

Fig. 12. (Design VI) Fault tolerant full-adder/subtractor circuit using Design III.

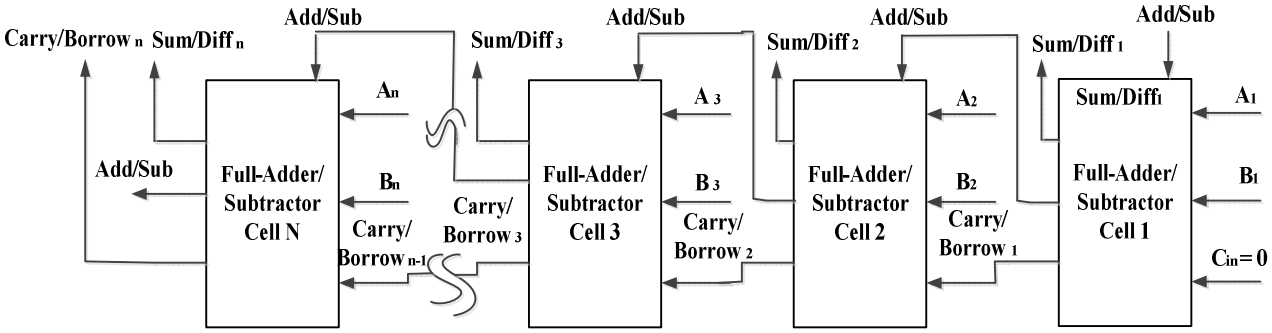

Fig. 15. *n*-bit parallel binary adder/subtractor circuit.

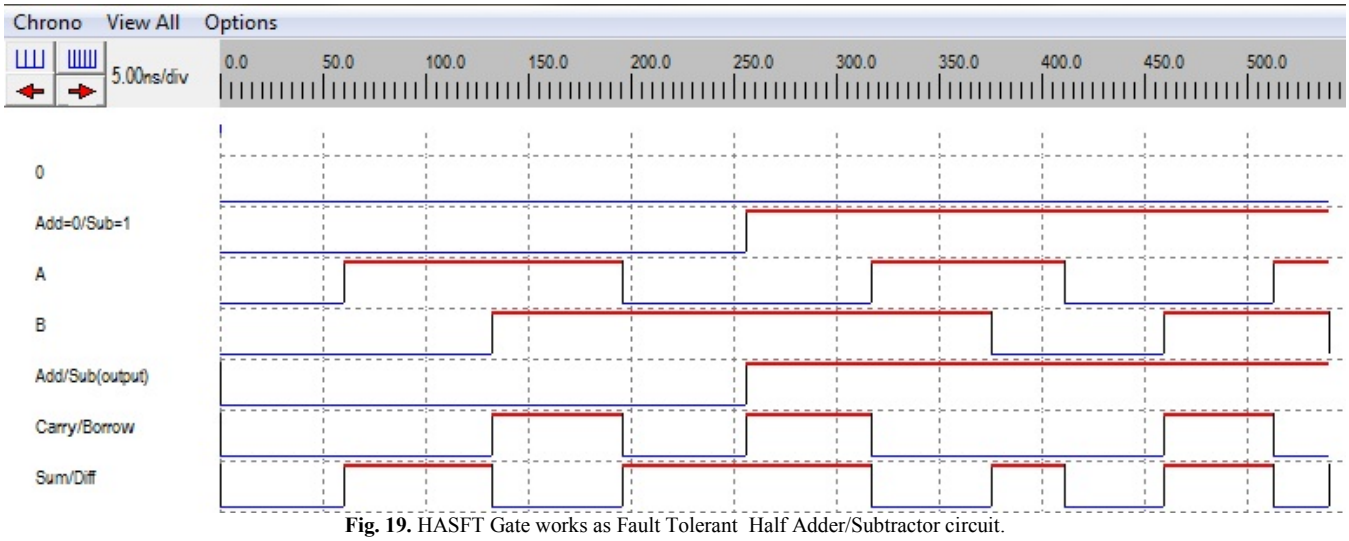

Fig. 19. HASFT Gate works as Fault Tolerant Half Adder/Subtractor circuit.

Fig. 20. FASFT Gate Works as Fault Tolerant Full Adder/Subtractor circuit.

REFERENCES

1. Landauer, Rolf. "Irreversibility and heat generation in the computing process.", IBM journal of research and development 5, no. 3 (1961): 183-191.
2. Bennett, Charles H. "Logical reversibility of computation." IBM journal of Research and Development 17, no. 6 (1973): 525-532.
3. Fredkin, Edward, and Tommaso Toffoli. Conservative logic. Springer London, 2002.
4. Toffoli, Tommaso. Reversible computing. Springer Berlin Heidelberg, 1980.
5. Leporati, Alberto, Claudio Zandron, and Giancarlo Mauri. "Simulating the Fredkin Gate with Energy-Based P Systems." J. UCS 10, no. 5 (2004): 600-619.
6. Babu, Hafiz Md Hasan, Md Rafiqul Islam, Syed Mostahed Ali Chowdhury, and Ahsan Raja Chowdhury. "Synthesis of full-adder circuit using reversible logic." In VLSI Design, 2004. Proceedings. 17th International Conference on, pp. 757-760. IEEE, 2004.
7. NG, Prashanth, M. B. Anandaraju, and Naveen KB. "Design of Efficient Reversible Fault tolerant Adder/Subtractor." International Journal of Computer Applications 74 (2013).
8. Feynman, Richard P. "Quantum mechanical computers." Foundations of physics 16, no. 6 (1986): 507-531.
9. Peres, Asher. "Reversible logic and quantum computers." Physical review A 32, no. 6 (1985): 3266.
10. Margolus, Norman H. Physics and computation. No. MIT/LCS/TR-415. MASSACHUSETTS INST OF TECH CAMBRIDGE LAB FOR COMPUTER SCIENCE, 1988.
11. Haghparast, Majid, and Keivan Navi. "A Novel Fault Tolerant Reversible Gate for Nanotechnology Based Systems." American Journal of Applied Sciences5, no. 5 (2008).
12. Parhami, Behrooz. "Fault-tolerant reversible circuits." In Signals, Systems and Computers, 2006. ACSSC'06. Fortieth Asilomar Conference on, pp. 1726-1729. IEEE, 2006.
13. Rangaraju, H. G., U. Venugopal, K. N. Muralidhara, and K. B. Raja. "Low power reversible parallel binary adder/subtractor." arXiv preprint arXiv: 1009.6218(2010).
14. Islam, Md Saiful, M. M. Rahman, Zerina Begum, and M. Z. Hafiz. "Low Cost Quantum Realization of Reversible Multiplier Circuit." Information technology journal 8, no. 2 (2009).
15. Kaur, Jaspreet, and Harpreet Kaur. "Synthesis and Designing of Reversible Adder/Subtracter Circuits." *Synthesis* 3.5 (2014).