Research Article

Reversible Logic-Based Fault-Tolerant Nanocircuits in QCA

Bibhash Sen, 1 Siddhant Ganeriwal, 1 and Biplab K. Sikdar 2

1 Department of Computer Science and Engineering, National Institute of Technology, Durgapur 713209, India
2 Department of Computer Science and Technology, Bengal Engineering and Science University, Shibpur 711103, India

Correspondence should be addressed to Bibhash Sen; bibhash.sen@gmail.com

Received 28 April 2013; Accepted 26 May 2013

Academic Editors: D. Rossi and P. Wachulak

Copyright © 2013 Bibhash Sen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Parity-preserving reversible circuits are gaining importance for the development of fault-tolerant systems in nanotechnology. On the other hand, Quantum-dot Cellular Automata (QCA), a potential alternative to CMOS, promises efficient digital design at nanoscale. This work targets design of reversible ALU (arithmetic logic unit) in QCA (Quantum-dot Cellular Automata) framework. The design is based on the fault tolerant reversible adders (FTRA) introduced in this paper. The proposed fault tolerant adder is a parity-preserving gate, and QCA implementation of FTRA achieved 47.38% fault-free output in the presence of all possible single missing/additional cell defects. The proposed designs are verified and evaluated over the existing ALU designs and found to be more efficient in terms of design complexity and quantum cost.

1. Introduction

Reversible logic has attractive perspective of constructing digital devices that can realize computing unit with almost zero power dissipation. Landauer [1] proved that for irreversible computations, each bit of information loss generates $k_B T \ln 2$ joules of heat energy. The energy $E_{\text{bit}}$ required for a binary transition is given by SNL (Shannon-Von Neumann-Landauer) expression in [1] as follows:

$$E_{\text{bit}} \geq E_{\text{SNL}} = k_B T \ln 2 = 0.017 \text{ eV},$$

where $k_B$ is Boltzmann constant and $T = 300$ K. This is the minimum energy to process a bit. Bennett [2] showed that a zero power dissipation in logic circuit is possible only if the circuit is composed of reversible logic gates. Since QCA circuits are clocked information preserving systems, the energy dissipation of QCA circuits can be significantly lower than $k_B T \ln 2$. This feature favours the introduction of QCA technology in reversible logic design.

Though, reversibility recovers bit loss, but it is not able to detect bit error in circuit. Fault-tolerant reversible circuits are capable of preventing errors at outputs. If the system itself made of fault-tolerant components, then the detection and correction of faults become easier and simple. In communication and many other systems, fault tolerance is achieved by parity. Therefore, parity-preserving reversible circuits will be the future design trends to the development of fault-tolerant reversible systems in nanotechnology.

On the other hand, QCA (Quantum-dot Cellular Automata) is considered to be promising in the field of nanotechnology due to their extremely small sizes and ultralow-power consumption [3]. The QCA is based on encoding binary information in the charge configuration of quantum-dot cells. The interaction between cells is coulombic and provides the necessary computing power. The fundamental unit of QCA-based design is the 3-input majority gate [3]. Majority gate with inverter, called MI, is used to realize the QCA designs because of its functional incompleteness.

Significant contributions have been made in the literature towards the design of arithmetic units in [4–6]. Also, fault-tolerant full adder circuits are explored in [7–10]. However, there have only been a few efforts towards designing reversible fault-tolerant ALUs.

This motivates us to design a fault-tolerant reversible ALU architecture considering QCA technology. A cost-effective realization of a fault-tolerant reversible adder (FTRA) is first introduced. It is then utilized to synthesize the desired fault-tolerant reversible arithmetic logic unit which outperforms the efficiency of existing designs in terms of design complexity and quantum cost. The major contributions of this work
around reversible QCA architecture can be summarized as follows:

(i) realization of generic fault-tolerant reversible adder having parity-preserving logic with cost effective quantum cost;

(ii) the presented fault-tolerant reversible adder block is used to realize different arithmetic circuit such as full adder, subtractor, ripple carry adder, and carry-skip logic;

(iii) synthesis of a fault-tolerant reversible arithmetic logic unit (ALU) using proposed adder;

(iv) application of the proposed adder in QCA nanotechnology with effective fault tolerance of 47.38% in the presence of all possible single missing/additional cell defects.


2. Preliminaries

Reversible Logic. A logic gate is reversible if the mapping of its inputs to outputs is bijective; that is, every distinct input yields a distinct output, and the number of inputs is equal to the number of outputs [12]. An important cost metrics in reversible logic circuits is the quantum cost. The cost of every $2 \times 2$ gate is the same; that is, unity and the cost of $1 \times 1$ gate are zero [13]. Any reversible logic can be realized using primitive quantum gates, like $1 \times 1$ NOT gates and $2 \times 2$ reversible gates, such as Controlled-$V$, Controlled-$V^+$, and CNOT gate (Figure 1). The quantum cost of a reversible gate can be calculated by counting the numbers of primitive quantum gates used in implementing it. The fan-outs and feedback paths are not permitted in reversible logic.

QCA Basics. A QCA cell consists of four quantum dots positioned at the corners of a square (Figure 2(a)) and contains two free electrons [3]. The electrons can quantum mechanically tunnel among the dots and settle either in polarization $P = -1$ (logic 0) or in $P = +1$ (logic 1) as shown in Figure 2(b). Timing in QCA is accomplished by the cascaded clocking of four distinct and periodic phases [3]. The basic structure realized with QCA is the 3-input majority gate, $MV(A, B, C) = Maj(A, B, C) = AB + BC + CA$ (Figure 2(c)). It can also function as a 2-input AND (2-input OR) logic by fixing one of the three input cells to $P = -1$ ($P = +1$). Inverter is realized in two different orientations as shown in Figure 2(d). In QCA-based logic implementations, two kinds of wire crossover, termed coplanar crossover and multilayer crossover, are possible. Figure 2(e) describes the coplanar wire crossing considering a 90° ($\times$-cell) and a 45° ($+\times$-cell) structures.

Fault-Tolerant Logic. Fault tolerance enables a system to operate properly in the event of the failure of some its components. If the system itself is made up of fault-tolerant components, then the detection and correction of faults become easier and simple. A fault-tolerant (FT) reversible gate is also called conservative gate [7]. The hamming weight of its inputs and outputs is equal. Let the input and output vectors of any fault-tolerant gate be $I_v = I_0, I_1, \ldots, I_{n-1}$ and $O_v = O_0, O_1, \ldots, O_{n-1}$, where

(i) $I_v \langle$Bijective$\rangle O_v$,

(ii) $I_v \oplus I_1 \oplus \cdots \oplus I_{n-1} = O_0 \oplus O_1 \oplus \cdots \oplus O_{n-1}$.

Few fault-tolerant reversible $3 \times 3$ gate-like Feynman double gate (F2G), Fredkin (FRG), NFT, and so forth and $4 \times 4$ gate like MIG are already investigated.

3. Related Work

In [14], the feasibility of the parity-preserving approach to design of reversible logic circuits was explored. Few new fault-tolerant reversible logic gates were proposed in [8]. Fault-tolerant full adder circuits are explored in [7–10]. In all these researches, no such single logic unit was identified to implement fault-tolerant full adder. Also, the fault tolerance
capability of those logic gates was not explored quantitatively. Significant contributions have been made in the literature towards the design of arithmetic units in [4–6]. However, there have only been a few efforts towards designing reversible fault-tolerant ALUs. A new fault-tolerant method based on majority multiplexing (Maj-MUX) is proposed for QCA in [15]. But the application of reversible logic in QCA to achieve fault tolerance was not identified (which is of primary interest to us in this work).

4. Proposed Fault-Tolerant Adder

In communication and many other systems, fault tolerance is achieved by parity. A parity-preserving reversible gate, when used with an arbitrary synthesis strategy for reversible logic circuits, allows any fault that affects no more than a single logic signal to be detectable at the circuit’s primary outputs [14]. In [10], a minimum number of garbage outputs and constant inputs for a fault-tolerant reversible full adder circuit are specified as 3 and 2. Keeping in mind to have minimum number of garbage outputs and constant inputs, a new fault-tolerant reversible adder (FTRA) is proposed here (Figure 3(a)) followed by design of different adder-based circuits. The input to output mapping of FTRA structure is 

\[ P = A, Q = A \oplus B, R = A \oplus B \oplus C \oplus D, S = (A \oplus B)(C \oplus D)(AB \oplus D), \]

and 

\[ T = (A \oplus B)(C \oplus D)(\overline{A}B \oplus D) \oplus E, \]

where A, B, C, D, and E are inputs and P, Q, R, S, and T are outputs. It is a 5 × 5 reversible gate with a quantum cost of 8 (Figure 3(b)). The proposed reversible FTRA gate is parity preserving. The reversibility and parity preserve nature of the proposed FTRA gate can be verified from Table 1.

A fault-tolerant reversible full adder and subtractor circuit using the newly proposed FTRA gate is shown in Figure 4(a). The design uses only one FTRA gate, has a quantum cost of 8, produces 3 garbage outputs, and uses 2 constant inputs only. No such single logic gate is found which can implement a fault-tolerant reversible full adder with such quantum cost and less complexity. Design capability of the proposed adder is further analyzed by implementing a 4-bit fault-tolerant ripple carry adder (Figure 4(b)) and carry skip adder (Figure 4(c)). A comparative performance analysis of the proposed design with existing designs is reported in Table 2 with enviable gate count besides other parameters. The results shows that the proposed design is much more effective than the existing designs.

The advantage of our method is in the implementation of this logic at gate level. Thus, once the required gates have been designed and an appropriate synthesis framework has been established, fault-tolerant implementation requires no extra expenditure in design or verification effort.

Figure 1: Elementary quantum logic gates: (a) NOT, (b) exclusive OR, (c) square root of NOT (SRN), and (d) Hermitian matrix of SRN.

Figure 2: QCA basics.
(a) Block diagram fault-tolerant adder

A
B
C
D
E
P = A
Q = A ⊕ B
R = A ⊕ B ⊕ C ⊕ D
S = (A ⊕ B)(C ⊕ D) ⊕ (AB ⊕ D)
T = (A ⊕ B)(C ⊕ D) ⊕ (A'B ⊕ D) ⊕ E

(b) Quantum implementation of fault-tolerant adder

**Figure 3:** Proposed fault-tolerant reversible adder.

(a) Fault-tolerant full adder and subtractor

(b) Fault-tolerant ripple carry adder

(c) Fault-tolerant carry-skip adder

**Figure 4:** Implementation of different logic circuit with proposed FTRA.
Figure 5: Fault-tolerant ALU with proposed FTRA.

Figure 6: Implementation of n-bit fault-tolerant ALU.

Table 2: Performance analysis of different fault-tolerant full adders.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>[7]</th>
<th>[8]</th>
<th>[9]</th>
<th>[10]</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gate count</td>
<td>5</td>
<td>6</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Quantum cost</td>
<td>25</td>
<td>18</td>
<td>14</td>
<td>11</td>
<td>8</td>
</tr>
<tr>
<td>Garbage outputs</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Constant inputs</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Logical calculations</td>
<td>$8\alpha + 16\beta + 8\delta$</td>
<td>$12\alpha + 8\beta + 4\delta$</td>
<td>$8\alpha + 6\beta + 2\delta$</td>
<td>$9\alpha + 4\beta + 3\delta$</td>
<td>$13\alpha + 4\beta + \delta$</td>
</tr>
</tbody>
</table>

$\alpha$: “two-input EXOR calculation”; $\beta$: “two-input AND calculation”; $\delta$: “NOT calculation.”
4.1. Design of Fault-Tolerant Arithmetic Logic Unit (ALU). The reversible fault-tolerant 1-bit ALU is designed with one FTRA gate, two Fredkin gates, and four Double-Feynman gates. Thus, the design uses a total of 7 gates and has a quantum cost of 26 (Figure 5). It produces 5 garbage outputs and uses 4 constant inputs. The various operations performed by the proposed FTRA-ALU are shown in Table 3. The total logical operations performed is $21\alpha + 10\beta + 5\delta$. The main advantage of the proposed ALU logic is its programmable feature; that is, just programming the constants $C_1$, $C_2$, and $C_3$, different ALU functions are implemented. Besides its fault tolerance, this programmable feature adds more flexibility in reversible ALU. Figure 6 shows the design of an n-bit fault-tolerant ALU which is synthesized by cascading the 1-bit ALU module explored in Figure 5.

Table 3: Different function of ALU.

<table>
<thead>
<tr>
<th>$C_0$</th>
<th>$C_1$</th>
<th>$C_2$</th>
<th>Output</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$AB$</td>
<td>AND</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>$A + B$</td>
<td>NOR</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>$A \overline{B}$</td>
<td>NAND</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>$A + B$</td>
<td>OR</td>
</tr>
<tr>
<td>1</td>
<td>$X$</td>
<td>0</td>
<td>$A \oplus B \oplus C$</td>
<td>ADD</td>
</tr>
<tr>
<td>1</td>
<td>$X$</td>
<td>1</td>
<td>$A \oplus B \oplus C$</td>
<td>SUB</td>
</tr>
</tbody>
</table>

5. FTRA Gates in QCA Computing

To demonstrate the application of the proposed fault-tolerant design approach to reversible adder in emerging nanotechnologies, Quantum-dot cellular automata (QCA) technology is considered because reversible logic has potential applications in QCA computing. The QCA implementation of the proposed FTRA gate is shown in Figure 7. The design requires 27 MVs and has a delay of 12 clock zones. In the following section, the fault tolerance capability is established quantitatively. FTRA simulation shown in Figure 8 is performed by fixing $D$ and $E$ to 0. In this simulation result, $R$ generates sum, and $S$ propagates carry.

6. Fault Tolerance of Proposed Logic Gate

In QCA manufacturing, defects can occur during the synthesis and deposition phases, although defects are most likely to take place during the deposition phase [16]. Researchers have shown that QCA cells are more susceptible to missing and additional QCA cell defects [17]. In additional cell defect, an additional cell is deposited on the substrate. The missing cell defect occurs due to the missing of a particular cell. Researchers have been addressing the design and test of QCA circuits assuming the single missing/additional cell defect model.
### Figure 8: Simulation result of fault-tolerant QCA adder.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>A, B, C</td>
<td>P, Q, R, S, T</td>
</tr>
</tbody>
</table>

#### Detailed Inputs and Outputs:

- **A** Max: $1.00e + 000$
  Min: $-1.00e + 000$

- **B** Max: $1.00e + 000$
  Min: $-1.00e + 000$

- **C** Max: $1.00e + 000$
  Min: $-1.00e + 000$

- **P** Max: $9.54e+ 000$
  Min: $-9.54e+ 000$

- **Q** Max: $9.54e+ 000$
  Min: $-9.54e+ 000$

- **R** Max: $9.54e+ 000$
  Min: $-9.54e+ 000$

- **S** Max: $9.54e+ 000$
  Min: $-9.54e+ 000$

- **T** Max: $9.54e+ 000$
  Min: $-9.54e+ 000$

### Figure 9: HDLQ modelling of QCA-based fault-tolerant adder.
Table 4: Fault pattern of proposed fault-tolerant adder.

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
<th>Fault pattern</th>
<th>Success rate</th>
<th>FT%</th>
</tr>
</thead>
<tbody>
<tr>
<td>a₀</td>
<td>a₀</td>
<td>a₁₆, a₁₂, a₀, a₂, a₁, a₀, a₁₂, a₁, a₀</td>
<td>32/78</td>
<td>41.02</td>
</tr>
<tr>
<td>a₁</td>
<td>a₁</td>
<td>a₁₇, a₁₃, a₁, a₁, a₀, a₁, a₀</td>
<td>34/78</td>
<td>43.59</td>
</tr>
<tr>
<td>a₂</td>
<td>a₇</td>
<td>a₃, a₈, a₇, a₆, a₅, a₄, a₃, a₂, a₁, a₀</td>
<td>38/78</td>
<td>48.72</td>
</tr>
<tr>
<td>a₃</td>
<td>a₆</td>
<td>a₁₂, a₈, a₇, a₆, a₅, a₄, a₃, a₂, a₁, a₀</td>
<td>38/78</td>
<td>48.72</td>
</tr>
<tr>
<td>a₄</td>
<td>a₄</td>
<td>a₁₈, a₁₁, a₄, a₆, a₅, a₄, a₃, a₂, a₁, a₀</td>
<td>37/78</td>
<td>47.43</td>
</tr>
<tr>
<td>a₅</td>
<td>a₅</td>
<td>a₁₀, a₇, a₆, a₅, a₄, a₃, a₂, a₁, a₀</td>
<td>39/78</td>
<td>50.00</td>
</tr>
<tr>
<td>a₆</td>
<td>a₃</td>
<td>a₁₅, a₃, a₂, a₁, a₀</td>
<td>40/78</td>
<td>51.28</td>
</tr>
<tr>
<td>a₇</td>
<td>a₇</td>
<td>a₀, a₁₆, a₁₇, a₁₉, a₁₉, a₁₆, a₁₇, a₁₆, a₁₅, a₁₄</td>
<td>41/78</td>
<td>52.56</td>
</tr>
<tr>
<td>a₈</td>
<td>a₈</td>
<td>a₁₆, a₁₂, a₁₈, a₁₈, a₁₇, a₁₆, a₁₅, a₁₄, a₁₃, a₁₂</td>
<td>42/78</td>
<td>53.78</td>
</tr>
</tbody>
</table>

Average FT = fault tolerance
47.38

In the proposed work, the QCA layout of the FTRA gate is converted into the corresponding hardware description language notations using the HDLQ Verilog library [18]. The HDLQ design tool, Verilog equivalent for QCA, consists of a Verilog HDL library of QCA devices, that is, MV, INV, FO, Crosswire (CW), and L-shape wire with fault injection capability. The HDLQ-modeled design of the FTRA gate is shown in Figure 9, and an exhaustive testing of the HDLQ model of the FTRA gate is conducted with 32 input patterns in the presence of all possible single missing/additional cell defects. The design is simulated using the Verilog HDL simulator in the presence of faults to determine the corresponding outputs.

Testing of the FTRA gate generated 78 unique fault patterns at the output, as shown in Table 4. Due to huge volume of data, the fault pattern table is partially shown. In the fault patterns study shown in Table 4, a₁ is the five-bit pattern with an equivalent decimal value of i. For example, a₀ represents 00000 (decimal 0), and a₃ represents 11111 (decimal 31). In Table 4, first two columns present the correct behaviour of FTRA gate, that is, for a particular input vector corresponding expected output vector. Each row of the Table 4 represents the output generated after the fault injection of different modules of the HDLQ model (Figure 9), for example, for given input a₀, after the fault injection of different modules, the generated fault pattern is a₀, a₁₆, a₁₲, a₁₂, a₁. From those fault patterns, we observed that there are average 47.38% successful patterns that produce the correct output, even when there is a fault.

7. Conclusions

This work presents a novel architecture of fault-tolerant reversible adder (FTRA) gate. Experimental results establish the fact that the proposed FTRA achieved significant improvements in reversible circuits over the existing ones. A reversible arithmetic logic unit is synthesized based on the FTRA proposed. This is first attempt to synthesize a fault-tolerant full adder/subtractor using only single reversible logic block (FTRA) with optimal quantum cost to avoid wire-crossing bottleneck. Also, the application of this fault-tolerant logic in QCA nanotechnology gets an extra advantage in fault-tolerant computing with effective 47.38% fault-free output in the presence of all possible single missing/additional cell defects.

Though, the clocking structure beneath the QCA cell layer is also very important and nontrivial research issue.

Acknowledgment

The authors would like to thank Mr. Samik Some, UG student of CSE department, NIT Durgapur, for his valuable help in testing the design using HDLQ tool.

References


Submit your manuscripts at http://www.hindawi.com