Online Testable Efficient Latches for Molecular QCA Based on Reversible Logic

Debajyoty Banik

Abstract Quantum computer is very advanced technology in upcoming era and can capable to solve any complex problem with very fast computation power. The proposed models for quantum computation are quantum dot cellular automata (QCA). The molecular QCA has the tendency to high error rates. In case of molecular quantum dot cellular automata, the main objective of circuit design is the reduction of circuit area with wanted functional behavior. In this article, we propose the efficient design of online testable latches based on reversible logic for molecular QCA that is very much cost-effective respect to circuit area and other parameters. We use reversible gates having conservative property; i.e., capability for producing the equal number of 1s in output bits as input bits. So, conservative logic gates are subset of parity preserving reversible gate. Fault patterns of used conservative logic gate are analyzed for single stuck-at faults in molecular QCA circuit. Our proposed latches are able to examine single stuck-at fault, missing/additional QCA cell defect online, including permanent or transient fault in molecular QCA and efficient respect to circuit area. The designs of QCA layouts for various latches are presented and verified using QCA Designer and the Verilog HDL library of QCA devices is used to present HDLQ design tool.

1 Introduction

Molecular QCA is most demandable emerging nanotechnology having small shaped component, ultra low power consumption and high clock frequency [1, 2]. Molecular QCA cell can define the logic states, such as logic 0 or logic 1, by depending upon the electrons' location in it. Due to the tendency of high error rate in nanotechnology, it is necessary to check the error frequently in the circuit. And it is possible only when the circuit has online testable capability. Defect in molecular QCA can occur in various phases, such as synthesis, deposition, and runtime; missing/additional defects take

D. Banik (✉)
Indian Institute of Technology Patna, Patna, India
e-mail: debajyoty.pcs13@iitp.ac.in

A. K. Singh et al. (eds.), *Design and Testing of Reversible Logic*, Lecture Notes in Electrical Engineering 577, https://doi.org/10.1007/978-981-13-8821-7_11

place in synthesis and deposition phase [3]. Stuck-at fault and transient fault are more likely to take place in runtime for external unwanted energy or internal cell defect.

We introduce an efficient design of online testable latches for molecular QCA. The molecular QCA has various application of reversible computation. The reversible computation is possible only if the primary elements of the system are reversible gate. A reversible gate computes the function with bijective in manner. A reversible gate is n×n gate which has same number (n) of inputs and outputs with bijective mapping within input and output vectors. Landauer has proved that $k_B T ln2$ Joules energy is dissipated for every bit of information loss which is possible only irreversible logic [4]. Where, k_B is the Boltzmann constant and T represents the operating temperature. This energy dissipation Will not be occurred with reversible computation, firstly proposed by Bennett [5]. The previous exertion for applying reversible logic in molecular QCA is described in [6], in which reversible logic testing properties are described in a 1-D array of QCA cells.

The conservative reversible gates are used as basic component of the latches to provide online testability in the circuits and to reduce energy dissipation.

Conservative logic belongs to the logic family having a power to generate same number of 1s in the input and the output signals [7].

Definition 1: Bit conserving circuit is referred as conservative logic circuit. It must hold same number of 1s in input and output signals.

Definition 2: A circuit with is conservative in nature with having the reversibility property is known as conservative reversible logic circuit.

Definition 3: All high signals at the input lines of the conservative logic network are the cause to produce all high signals at the output lines, and all low signals at the input lines of the conservative logic network are the cause to produce all low signals at the output lines. Figure 1 shows the structure of very well-known reversible conservative Fredkin gate.

Here, the fault patterns for single stuck-at fault are analyzed. We get a clear idea from the fault patterns (for different type of faults) that when single fault will occur in the conservative logic gate it must mismatch the parity and specifically mismatch the number of 1s in output lines with input lines. Based on the above property we design online testable latches using conservative Fredkin gate. In existing paper, irreversible latches for molecular QCA are proposed without having online testable capability [8, 9]. Concurrently testable latches are proposed with decidedly less-cost efficient design which needs more molecular QCA cells to design [10]. So, the main contribution in this literature is to design various latches having online testable capability with cost-effective nature which needs few molecular QCA cells and minimal garbage lines.

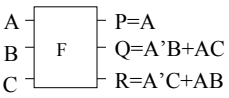

Fig. 1 Conservative reversible Fredkin gate

This literature is arranged by the following way: Sect. 2 describes background work related to our work including the basics, cost metric of molecular QCA and also turn the attention to related work. Section 3 presents conservative reversible gate and online testing of molecular QCA. Various online testable reversible latches in molecular QCA framework are designed in Sect. 4. Implementation and Various simulation results of the proposed online testable latches in molecular QCA framework to verify our designs are shown in Sect. 5. Single stuck-at fault patterns are shown in tabular format and normal circuit simulation result is mentioned in graphical format to verify the designs in this section. Finally, Sect. 6 describes conclusions along with upcoming works.

2 Background

Various type of faults may occur in molecular QCA, like stuck-at fault [11], missing/additional cell defect which is more likely during synthesis and deposition phase for miss placement of the QCA cell [3], etc. It also has the tendency for permanent faults as well as transient faults caused by radiation, thermodynamic impact, and other effects, like the variation of power within excited and ground states is very small [12]. So, any type of single stuck-at fault, single cell defect (missing/additional) including permanent fault or transient fault detection are important. Our proposed designs covered all of such faults. Thus the proposed designs are significant.

2.1 Basics of molecular Quantum Cellular Automata Computing

QCA cell which is nothing but coupled dot system, is the basic unit of molecular quantum dot cellular automata nanotechnology circuit. Each cell contains four dots at the vertices of the square. Among four dots two electrons can quantum mechanically tunnel in this QCA cell. Due to electrostatic repulsion, the electrons are occupying diagonal positions, which is mentioned in Fig. 2a. Positions of electron pair in the QCA cell are caused to polarize the QCA cell either in logic 0 ($P = -1$) or in logic 1($P = +1$), is mentioned in Fig. 2b.

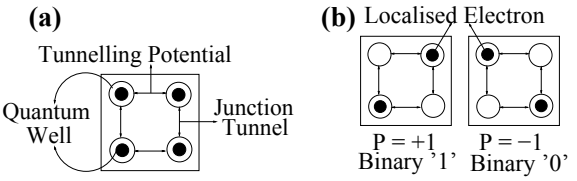

Fig. 2 QCA cell: **a** QCA cell formation. **b** Bi-polar QCA cell

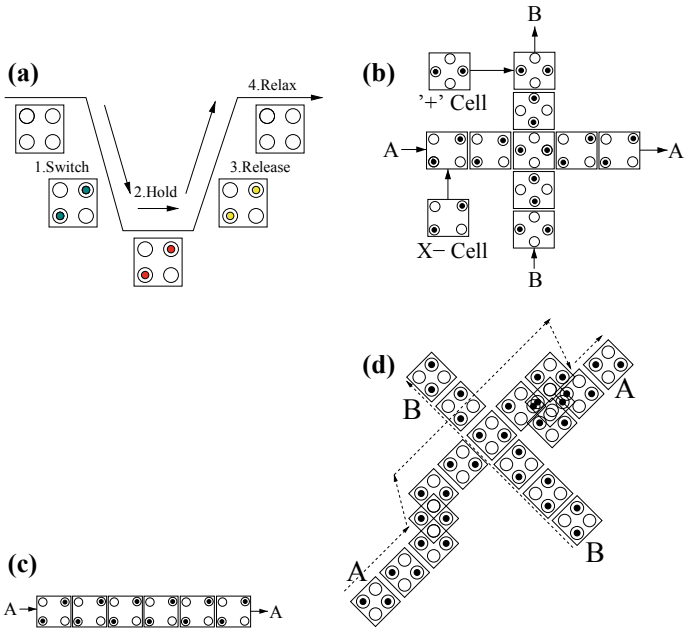

Fig. 3 **a** Clocking. **b** Coplanar wire-crossing. **c** Wire. **d** Multilayer wire-crossing

The four distinct periodic phases cascaded clocking is accomplished timing in QCA [4]. Figure 3a displays four distinct periodic phases are consist of switch, hold, release, and relax. Wire crossing in the molecular QCA is a remarkable achievement. The nature of the QCA cell makes wire-crossing more vulnerable. Undesirable crosstalk between wires may be introduced here. Wire-crossing determination is done either using different clock zones at different wires or using rotated QCA cells at one of the wires by coplanar approach. Considering 90° (X cell) and 45° (+ cell) structures form the coplanar wire crossing which is mentioned in Fig. 3b. The crossover can also be realized with multilayered wires. Instead of the coplanar strategy, multilayer strategy is more powerful [13]. Multilayer implementation of wire-crossing with same type of cells is mentioned in Fig. 3d.

Primary and the basic device of QCA computing is MV or majority voter, which is mentioned in Fig. 4. Property of majority voter is modeled as MV(A, B, C) = AB + BC + CA, which is a function to produce output from the majority of inputs A, B, and C. All other basic functions like OR, AND can be realized with majority voter. Two-input OR or a two-input AND gate can be realized through the fixation of a input cell at MV to p = +1 or p = −1, respectively. Realization of OR gate, AND gate is shown in Fig. 5.

INVERTER is another important logic device, which can be realized with many ways in QCA computing. Pictorial representation is mentioned in Fig. 6. Signal in QCA circuit may transfer through either inverter chain (Fig. 6c) or binary wire (Fig. 3c).

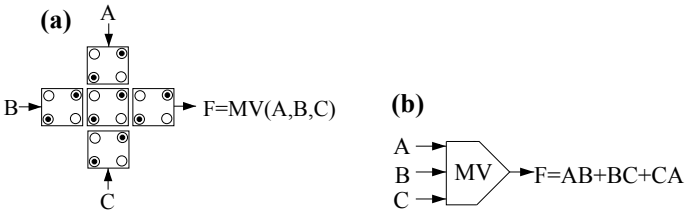

Fig. 4 Majority voter

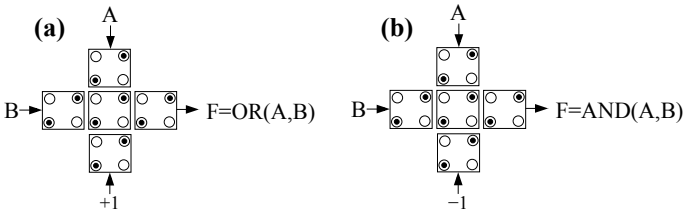

Fig. 5 **a** OR gate. **b** AND gate

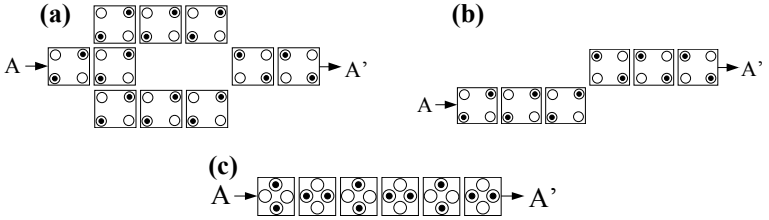

Fig. 6 **a** Inverter [type-1]. **b** Inverter [type-2]. **c** Inverter chain [type-3]

2.2 Cost Metric

Any circuit in molecular QCA framework must be designed with some cost-effective manner, which depends on number of QCA cell, majority voter, clock cycle is used to design the QCA circuit. We should reduce the cost metric to design efficient circuit.

Molecular QCA Cell

The QCA cell is primary element of the molecular quantum dot cellular automata. The QCA cell count should be minimum for reducing the circuit area. This is the most important cost metric because the main objective to design the circuit in any framework minimizes the area.

Majority Voter

In order to design the circuit with basic gates, majority voter is important to realize AND and OR gates. Minimum number of majority voters is considered as an efficient design to reduce the circuit area.

Clock Zone

The clock zone is another important cost metric because it estimates the time delay of the circuit. Thus, to make efficient circuit design we should minimize the delay or minimize the clock zone.

2.3 Related Work

Lent et al. proposed molecular QCA firstly in 1993 [14]. The first physically implemented molecular QCA produced Al islands and tunnel junctions. The experiment is done at 10mK [15]. As the circuit in molecular QCA having tendency to high error rate, so researchers were trying to concentrate on testing and the testing of molecular QCA first addressed in 2004 [3]. Here, the QCA devices defect characterization has been demonstrated, and also described how the difference between the testing of molecular QCA and the CMOS. QCA defects were modeled at molecular level for combinational circuits in [16]. It characterized faults in respect to single additional/missing cell defect on different QCA devices (i.e., INV, fan-out, MV, L-shape wire, cross wire, etc.). The test generation for molecular QCA described in [11]. For the QCA reversible logic circuit, single additional/missing cell defects were proposed in [6]. It concludes that reversible 1-D array must be C-testable. The fault-tolerant molecular QCA designs using the modular redundancy with the shifted operands, were shown in [17]. Considering various faults and wire delay, defect is modeled and SR latch is presented. The sequential circuits in QCA is presented based on the SR latch [8, 9]. The coplanar crossing in molecular QCA was presented and proved that wires with rotated cells can be considered as thermally more stable cells in [18]. The single stuck-at fault testing for combinational circuit is presented in [19]. Testing of reversible circuits is shown in [20–24] but they have not test circuit in of molecular framework. Though testable latches in QCA framework for missing/additional cell defect was done in [10], but it does not target to test stuck-at fault and it is not much efficient design respect to circuit area, time delay, etc.

Latches and flip-flops are the primary ingredients of various registers. Nowadays, researchers incorporate registers also for transliteration [25] to increase the processing speed. The automated transliteration and translation [26] are booming topics for now [27, 28]. Energy efficient fault- tolerant architecture is designed for wireless

sensor networks [29]. Moreover, researchers introduced testable features into the sequential circuits which makes it more interesting [30–32]. In future, researchers may incorporate the proposed scheme in their applications [33] to improve efficiency.

3 Conservative Reversible Gate and Online Testing of Molecular QCA

The conservative reversible gate is a unique type of parity preserving logic which belongs to conservative reversible logic family. It has capability to form same number of 1s in output signals as the input signals. We shall discuss the conservative logic with case study of Fredkin gate. If anyone is interested to design the circuit with another conservative logic gate then they can also easily make the circuit online testable. The Fredkin gate is 3 × 3 conservative gate with reversible property. I_v = {A, B, C} and O_v = {A, A'B⊕AC, A'C⊕AB} having quantum cost 5; first proposed by Fredkin and Toffoli [7]. This is mentioned in Fig. 1. As the Fredkin gate belongs to the conservative reversible logic family. So, it is capable to produce equal number of 1's in input lines as output lines which can be cleared from behavioral analysis of the Fredkin gate which shown in Table 1. It should be noted that any type of single permanent or transient fault will cause to miss the conservative property as well as parity preserving property. Thus, depending upon the conservative property of the Fredkin gate, we can easily detect the fault online. Launder four-phase clocking scheme is most popular to design circuit in molecular QCA framework and our proposed designs are based on this clocking scheme. Figure 7 represents the Fredkin gate structure in molecular QCA.

The Fredkin gate is considered as basic building block of proposed latches based on the reversible logic in molecular QCA framework, having 6 FOs, 2 INVs, 6 MVs, 8 LSs, 5 CWs, and to implement it 233 QCA cells with 4 clock zone are needed.

In this work, we use Fredkin gate behalf of conservative logic gate. The Verilog library of the hardware description language notations for QCA layout is used to

Table 1 Behavioral study of the Fredkin gate

A	B	C	P	Q	R
Low	Low	Low	Low	Low	Low
Low	Low	High	Low	Low	High
Low	High	Low	Low	High	Low
Low	High	High	Low	High	High
High	Low	Low	High	Low	Low
High	Low	High	High	High	Low
High	High	Low	High	Low	High
High	High	High	High	High	High

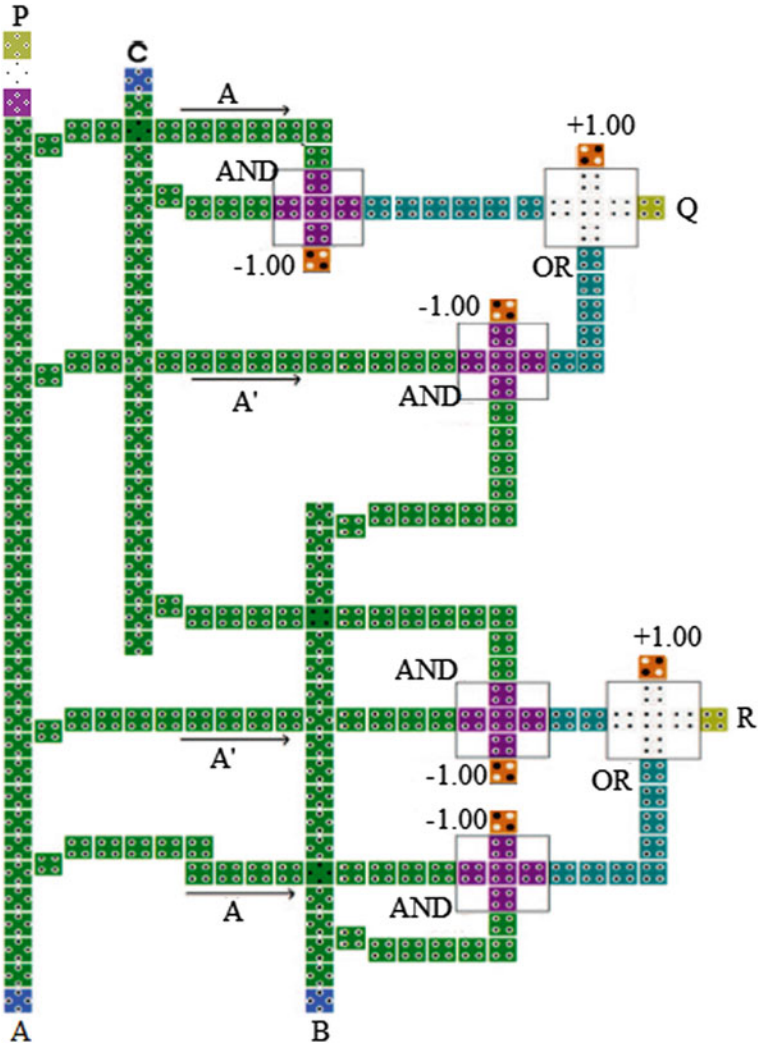

Fig. 7 QCA layout of Fredkin gate

show all possible single stuck-at fault and single cell defect (missing/additional) in majority voters (MJs), CWs, INVs, L-shape wires (LSs), FOs. The Verilog HDL library of molecular QCA devices is used as HDLQ design tool i.e., MV, FO, L-shape wire, Cross wire (CW), INV having ability to inject the faults [34]. Design HDLQ model for the Fredkin gate is mentioned in Fig. 8.

Verilog HDL simulator helps to simulate the proposed design and this is successfully done for the appearance of all the possible single stuck-at faults to determine corresponding erroneous outputs. 43 fault patterns were produced through the entire

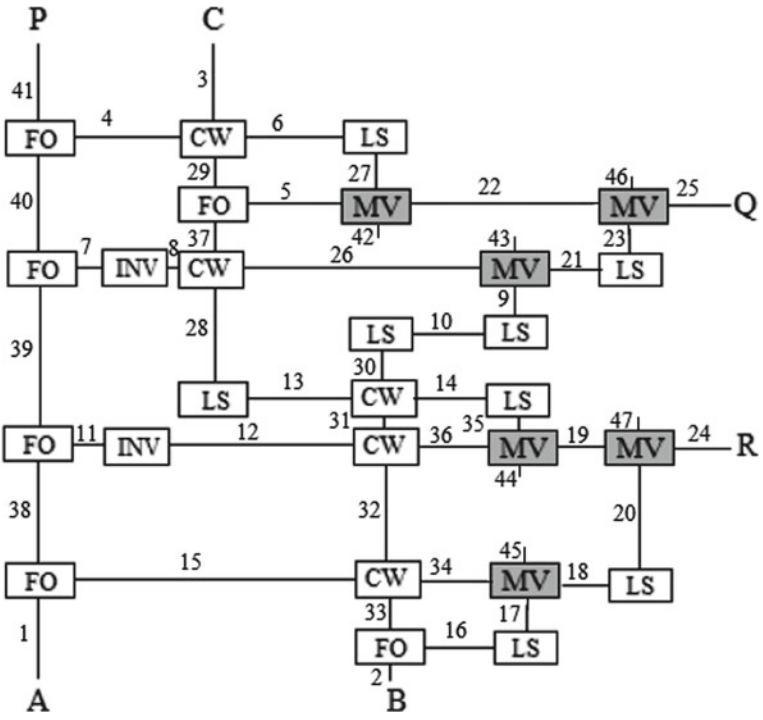

Fig. 8 Modeling for the Fredkin gate with QCA layout. INV describes inverter in QCA, FO illustrates QCA fanout device, MV describes QCA majority voter, LS shows QCA L-shape wire, and CW illustrates QCA cross wire

test for single stuck-at 0 and 45 fault patterns for single stuck-at 1. The fault patterns for single stuck-at 0 fault is mentioned in Tables 2 and 3, fault patterns for single stuck-at 1 fault is mentioned in Tables 4 and 5. In these tables, k represents the three-bit pattern of corresponding decimal value of roman k. s0 describes bit pattern 000 (decimal 0), a1 denotes bit pattern 001 (decimal 1), and so on. From the fault pattern table, it is clear that there must be mismatch in conservative nature if there exits any stuck-at fault. Conservative gate also mismatch in parity if there exist single cell defect (missing/additional) is occurred [10]. So, the fault patterns indicate the presence of fault, if any conservative discrepancy (i.e., number of 1s in input vector is unequal to number of 1s in output vector) is found between the input and output vectors. More generally parity mismatch helps to find the faults. So, any permanent or transient faults can be detected by parity mismatch property of conservative logic gate. Finally, it can be claimed that the Fredkin gate based on molecular QCA is online testable gate for its parity preserving property.

Table 2 Stuck-At 0 fault patterns in Fredkin gate (first 22 lines)

InV	EOV	Fault patterns																					
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	i	i	i	0	i	i	i	i	i	i	i	i	0	0	0	i	i	i	i	0	i	i	i
ii	ii	ii	0	ii	ii	ii	ii	ii	0	0	0	ii	ii	ii	ii	ii	ii	ii	ii	ii	ii	0	ii
iii	iii	iii	i	ii	iii	iii	iii	iii	i	i	i	iii	ii	ii	ii	iii	iii	iii	iii	ii	iii	i	iii
iv	iv	0	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv
v	vi	i	vi	iv	iv	iv	iv	vi	vi	vi	vi	vii	vi	vi	vi	vi	vi	vi	vi	vi	vi	vi	iv
vi	v	ii	iv	v	v	v	v	vii	v	v	v	v	v	v	v	iv	iv	iv	iv	v	iv	v	v
vii	vii	iii	vi	v	v	v	v	vii	vii	vii	vii	vii	vii	vii	vii	vi	vi	vi	vi	vii	vi	vii	v

InV = Input vector, and EOV = Expected output vector

Table 3 Stuck-At 0 fault patterns in Fredkin gate (remaining lines)

InV	EOV	Fault patterns																					
		23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	46	47	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
i	i	i	0	i	i	i	0	0	i	i	i	i	i	0	0	0	i	i	i	i	i	0	
ii	ii	0	ii	0	0	ii	ii	ii	0	0	0	0	ii	ii	ii	ii	ii	ii	ii	ii	0	ii	
iii	iii	i	ii	i	i	iii	ii	ii	i	i	i	i	iii	ii	ii	ii	iii	iii	iii	iii	i	ii	
iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	0	0	0	0	iv	iv	
v	vi	vi	vi	iv	vi	iv	vi	iv	vi	vi	vi	vi	vi	vi	vi	vi	i	0	0	ii	iv	vi	
vi	v	v	iv	v	v	v	v	v	v	v	v	v	iv	v	v	v	iii	iii	i	i	v	iv	
vii	vii	vii	vi	v	vii	v	vii	v	vii	vii	vii	vii	vi	vii	vii	vii	iii	iii	i	iii	v	vi	

InV = Input vector, and EOV = Expected output vector

Table 4 Stuck-At 1 fault patterns in Fredkin gate (first 22 lines)

InV	EOV	Fault patterns																					
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
0	0	iv	ii	i	0	0	0	0	0	ii	ii	0	0	i	i	0	0	0	i	i	i	ii	ii
i	i	vi	iii	i	iii	i	iii	i	i	iii	iii	0	i	i	i	i	i	i	i	i	i	iii	iii
ii	ii	v	ii	iii	ii	ii	ii	0	ii	ii	ii	ii	ii	iii	iii	iii	ii	ii	iii	iii	iii	ii	ii
iii	iii	vii	iii	iii	iii	iii	iii	i	iii	iii	iii	ii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii
iv	iv	iv	v	vi	iv	vi	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	v	v	v	v	v	vi	vi
v	vi	vi	vii	vi	vi	vi	vi	vi	vi	vi	vi	vi	vii	vi	vi	vi	vii	vii	vii	vii	vii	vi	vi
vi	v	v	v	vii	v	vii	v	v	vii	v	v	v	v	v	v	v	v	v	v	v	v	vii	vii
vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii

InV = Input vector, and EOV = Expected output vector

Table 5 Stuck-At 1 fault patterns in Fredkin gate (remaining lines)

InV	EOV	Fault patterns																						
		23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
0	0	ii	i	ii	0	0	i	i	ii	ii	ii	ii	0	i	0	i	iv	iv	iv	iv	0	ii	i	0
i	i	iii	i	iii	i	iii	i	i	iii	iii	iii	iii	i	i	i	i	vi	vii	vii	v	iii	iii	i	i
ii	ii	ii	iii	ii	ii	ii	iii	iii	ii	ii	ii	ii	iii	iii	ii	iii	iv	iv	vi	vi	ii	ii	iii	iii
iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	iii	vi	vii	vii	vii	iii	iii	iii	iii
iv	iv	vi	v	vi	iv	iv	iv	vi	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	iv	vi	iv	iv	v
v	vi	vi	vii	vi	vi	vi	vi	vi	vi	vi	vi	vi	vi	vi	vii	vi	vi	vi	vi	vi	vi	vi	vii	vii
vi	v	vii	v	vii	vii	v	v	vii	v	v	v	v	v	v	v	v	v	v	v	v	vii	vii	v	v
vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii	vii

InV = Input vector, and EOV = Expected output vector

4 Design of Online Testable Latches in molecular QCA Framework

In this part, we design efficient online testable latches based on minimum number of online testable conservative reversible Fredkin gate with minimal number of QCA cells, few majority voters, and significant clock zones.

Lemma *Any circuit based on online testable gates is also online testable.* □

Proof Online testable capability denotes the power of fully fault coverage runtime. Here, we consider the single stuck-at fault. So, the test vector should cover all zeroes and all ones in the input lines and output lines. The fault cover achieves with either to generate all combination of outputs with some characteristic or adding extra line which defines the correctness of the circuit. Now, if the testable advantage achieves with the first characteristic then there is no problem to test each block, because it can be tested with comparing specified characteristic for every combination of input which is applied from preceding block. If the testable advantage achieves with the mentioned second characteristic then each block must generate error detection signal for each combination of provided input to that block. Finally, after getting green signals from every block it can easily confirm to have no fault in the circuit.

Lemma *Proposed designs of sequential circuit based on multilayer approach are online testable.* □

Proof Multilayer based Fredkin gate in molecular QCA is online testable, proved in section III. According to Lemma 1, circuit based on Fredkin gate is online testable. As, it is sequential circuit, its input depends on output of previous state. According to conservative principle, generate output must have same number of 1's as applied 1's in input lines in fault-free case. Thus, after comparing equality in generated number of 1's at output with applied number of 1's including previous state signal to input

of the Fredkin gate, and consider $\overline{Q}$ as extra line which must be opposite of Q, we can claimed that the generated signal is error free and the circuit is fault free.

4.1 D Latch (Positive Level Triggered)

Equation of the level triggered D latch (positive):

$$Q_{n+1} = (D \wedge E) \vee (\overline{E} \wedge Q_n)$$

where, D and E are input bit and enable line, respectively. Q_n and Q_{n+1} are primary outputs of the present state and the next state, respectively. As per the level triggered D latch (positive) characteristic equation, it is cleared the straightaway reflection of D passes to the output signal when E is high, as $Q_{n+1} = D$. Positive level triggered D latch stays identical in its past state when the enable signal is low; as $Q_{n+1} = Q_n$. Figure 9 shows the D latch (positive level triggered). Though fan-out is restricted at reversible circuit whereas fan-out is acceptable in molecular QCA. Thus, Fig. 9 is valid for low power QCA.

As Fredkin gate has capability to detect fault online for its parity preserving characteristic, so the proposed D latch with the Fredkin gate also has the capability to detect fault online. To test the D latch, it is necessary to considered Q_n, E and D signals for input vector and G1, G2, Q_n as output vector; if parity mismatch is found within input and output vectors then it can be claimed that there is fault in the circuit. For testing the D latch (positive level triggered) with output Q and $\overline{Q}$, we only need to check same criteria and we should check additional $\overline{Q}$ bit which must be inverse of Q line if there exists no fault.

4.2 D Latch (Negative Level Triggered)

It will directly transfer the input signal D to Q line, when value of E is low; else estate will remain unchanged as before. The negative level triggered D latch can be modeled as,

$$Q_{n+1} = (D \wedge \overline{E}) \vee (E \wedge Q_n)$$

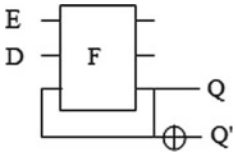

Fig. 9 Positive level triggered D latch with output Q and $\overline{Q}$

From the characterize equation, it is clear that D latch (negative level triggered) is mapped to Fredkin gate because of its MUXing capability, is mentioned in Fig. 10.

The proposed negative level triggered D latch also has capability to detect fault online. For testing, it is necessary to consider Q_n, E, and D signals for input vector and G1, G2, and Q signals as output vector. If parity mismatch is found within input and output vectors, then it can be claim that there is fault in the circuit because of its parity preserving property. For testing the D latch (negative level triggered) with output Q and $\overline{Q}$, only need to check same criteria and we should check additional $\overline{Q}$ bit which must inverse of Q bit for no fault.

4.3 T Latch

In this section, we design online testable T latch in molecular QCA framework. T latch is explained bellow:

$$Q_{n+1} = (T \oplus Q_n)E + \overline{E}Q_n$$

According to characterize equation of T latch, it is mapped to the conservative Fredkin gate. We design the online testable T latch with two Fredkin gates only. Our proposed design of online testable T latch with output lines Q and $\overline{Q}$ is mentioned in Fig. 11. To test the T latch online, it is necessary to consider Q_n, $\overline{Q}_n$, T, and E signals as input vector and G1, G2, G3, Q_{n+1}, and $\overline{Q}_{n+1}$ signals as output vector. Now if there is any mismatch in parity between the input and output vectors then this will be considered as faulty circuit.

4.4 JK Latch

In this section, online testable JK latch in QCA framework with two Fredkin gate is presented. The JK latch can be described as,

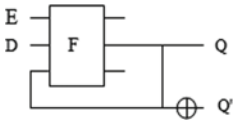

Fig. 10 Negative level triggered D latch

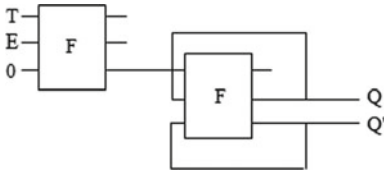

Fig. 11 Proposed design of T latch based on Fredkin gate

$$Q_{n+1} = (J\overline{Q}_n + \overline{K}Q_n)E + \overline{E}Q_n$$

From the characterize equation, it is clear that the JK latch can be realized using D latch or SR latch, but by this methodology circuit complexity will high, so alternative circuit diagram for JK latch is mentioned in Fig. 12.

The proposed JK latch based on the Fredkin gate also has capability to detect fault online. To test the JK latch, we should considered J, $\overline{K}$, E, and Q_n signals as input vector and G1, G2, G3, Q_{n+1} signals as output vector; if parity mismatch is found within input and output vectors, then it can be claimed that there is fault in the JK latch. For testing the JK latch with output Q and $\overline{Q}$, only need to check same criteria and we should check additional $\overline{Q}$ bit which must opposite of Q bit in fault-free case.

4.5 SR Latch

In this section, we propose the design of online testable SR latch in molecular QCA framework based on Fredkin gate. The characteristic equation of SR latch can be established as,

$$Q_{n+1} = (S + \overline{R}Q_n)E + \overline{E}Q_n$$

The characteristic equation can be mapped to D latch (positive level triggered) as $DE + \overline{E}Q_n$, where $D = (S + \overline{R}Q_n)$, and both of the equations can be realized using Fredkin gate. Our architecture for the testable SR latch is mentioned in Fig. 13.

Proposed SR latch based on the conservative Fredkin gate also have capability to detect fault online. To test the SR latch, we should consider all input bits including Q_n as input vector, and G1, G2, G3, G4, G5, G6, Q as output vector. If parity mismatch is found within input and output vectors, then it can be claimed to have fault in the SR latch. For testing the SR latch with output Q and $\overline{Q}$, only need to check same criteria and we should check additional $\overline{Q}$ bit which must opposite of Q bit in fault-free case.

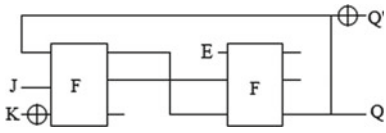

Fig. 12 JK latch based on Fredkin gate

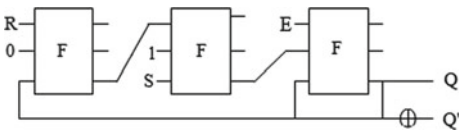

Fig. 13 SR latch based on Fredkin gate

5 Implementation to Functionally Verify The Proposed Multilayer Latches in Molecular QCA Framework

Verification of every design is done by using the QCADesigner version 2.0.3 [35]. Following parameters are used for the bistable approximation, cell size = 18 nm, radius of effect = 41 nm, convergence tolerance = 0.001000, number of samples = 182 800, clock amplitude factor = 2.000, clock high = 9.8 e^{-22}, clock low = 3.8 e^{-23}, relative permittivity = 12.9, maximum iterations per sample = 1000, and layer separation = 11.5000 nm. In the QCA layouts, each multilayer Fredkin gate will produce output by one clock cycle delay since Fredkin gate is designed with four clock zones, as shown in Fig. 7. The simulated waveform of the multilayer conservative Fredkin gate is mentioned in Fig. 14 by which it verifies the functionality with one delay of the circuit. The applied inputs to multilayer Fredkin gate at clock zone 0 and the produced outputs from Fredkin gate at clock zone 4. The simulation result is exactly same as truth table of the Fredkin gate. It verifies the correctness of the proposed multilayer implementation of Fredkin gate. The important fact is that all the designs are practical as well as usable, since it was verified using the QCADesigner simulator that generated signals are proper without degradation. To perform correctly, all signals should appear simultaneously at the majority gate [36]. All designs follow this characteristic.

5.1 Multilayer QCA Layout and Simulation of the Proposed Latches

All the latches are implemented with proposed multilayer Fredkin gate. The QCA layout of the triple layer D latch (positive level triggered) is mentioned in Fig. 15 and it is verified using QCADesiger version 2.0.3 which is same as expected truth table. To better understanding of conservative property, in Fig. 15 garbage outputs are managed with proper clock zone. G1 and G2 represent the garbage output. To verify the design, the simulated results of the proposed triple layer D latch (positive level triggered) are mentioned in Fig. 16. Here, arrows are used to verify the functionality of the proposed triple layer D latch (positive level triggered). Arrow A, B, and C show that when E is high then D will be reflected in Q_{n+1}; like arrow A and arrow C indicate that when E = 1 and D = 0 then $Q_n = 0$. And since E = 0 in the next cycle, Q_{n+1} will maintain its previous value as 0 (mentioned with lines D and F), Arrow B indicate that when E = 1 and D = 1 then Q = 1. And since E = 0 in the next cycle, Q_{n+1} will maintain its previous value of Q_n as 1 (mentioned with lines E). To design the triple layer D latch (positive level triggered), 292 QCA cells, 0.44 μm^2 area and four clock zones are needed. All the generated output will be delayed by one clock cycle as Fredkin gate has one clock cycle delay. The Table 6 which is compact form of Fig. 16, summaries the working functionality of proposed triple layer D latch (positive level triggered).

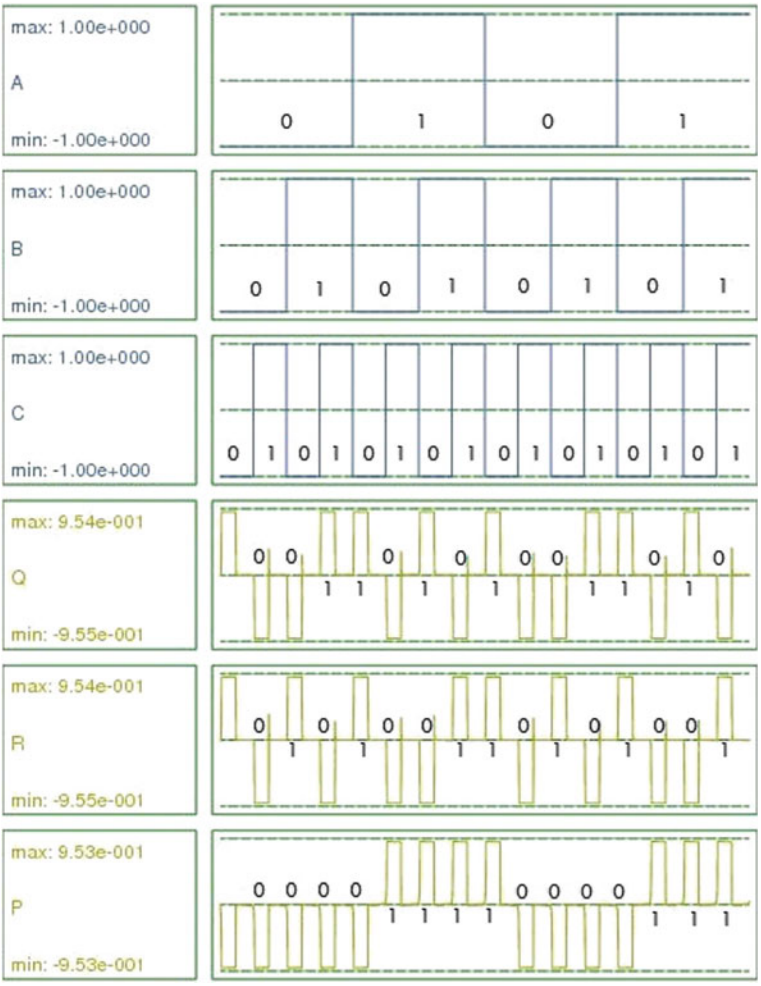

Fig. 14 Simulated results of multilayer Fredkin gate

Figure 17 shows multilayer representation of negative level triggered latch which is just opposite of positive level triggered latch. According to the characterize equation of negative level triggered latch, when E is low then D will be reflected in output Q_{n+1} and when E is high then the output Q_{n+1} will remain same as Q_n. To better understanding of conservative property, in Fig. 17 garbage outputs are managed with proper clock zone. G1 and G2 represent the garbage output. To verify the design, the simulated results of the proposed D latch (negative level triggered) are mentioned in Fig. 18. The trip of the arrows A and C indicate that when E = 0 and D = 0 then $Q_n = 0$. Since enable (E) = 1 in the next cycle, Q_{n+1} will maintain its previous value as 0 (mentioned with lines D and F), The trip of the arrow B indicates that when both

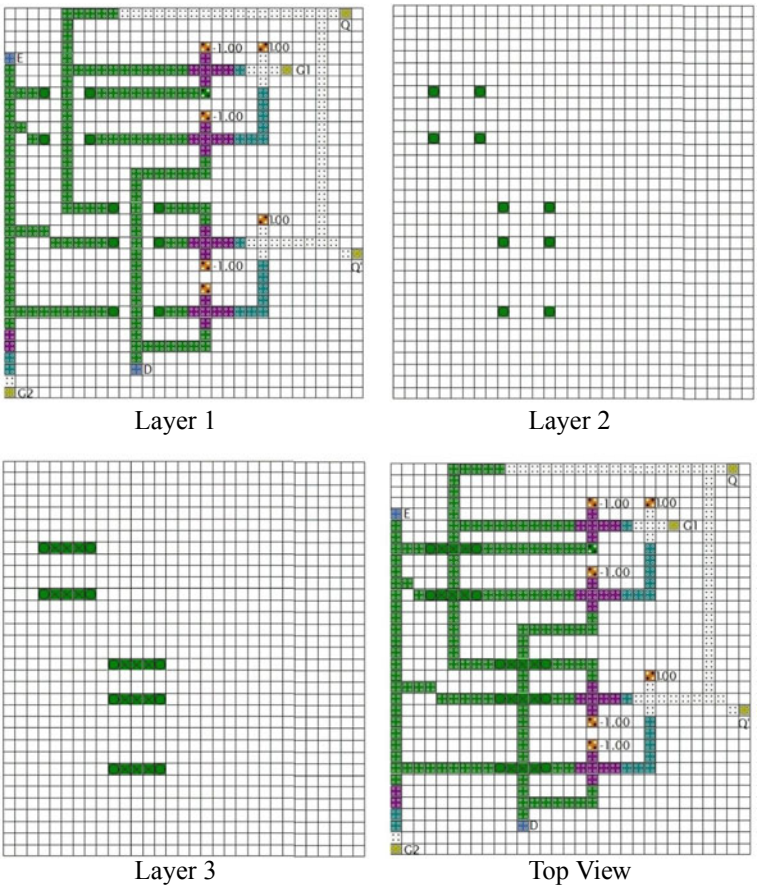

Fig. 15 Grid representation of proposed triple layer D latch (positive level triggered)

enable and the input line D are low then Q is based on previous state (i.e., high). And since E = 1 in the next cycle, Q_{n+1} will maintain its previous value of Q_n as 1 (mentioned with lines E). To design the triple layer D latch (negative level triggered), 272 QCA cells, 0.40 μm^2 area and four clock zones are needed. All generated outputs are delayed by a clock cycle. Table 7 which is compact form of Fig. 18, summaries the working functionality of proposed triple layer D latch (positive level triggered).

Figure 19 presents the top view representation of QCA layout and Fig. 20 show the simulation results of the triple layer T latch. To design the triple layer T latch, 628 QCA cells, 0.92 μm^2 area and eight clock zones are needed. Thus, after two clock cycles the correct output is generated. Table 8, summaries the working functionality of proposed triple layer T latch. The trip of arrow A represents toggling of Q_n at Q_{n+1} as when E = 1, T = 1, and $Q_n = 1$ then $Q_{n+1} = 0$. The Trip of the arrow B

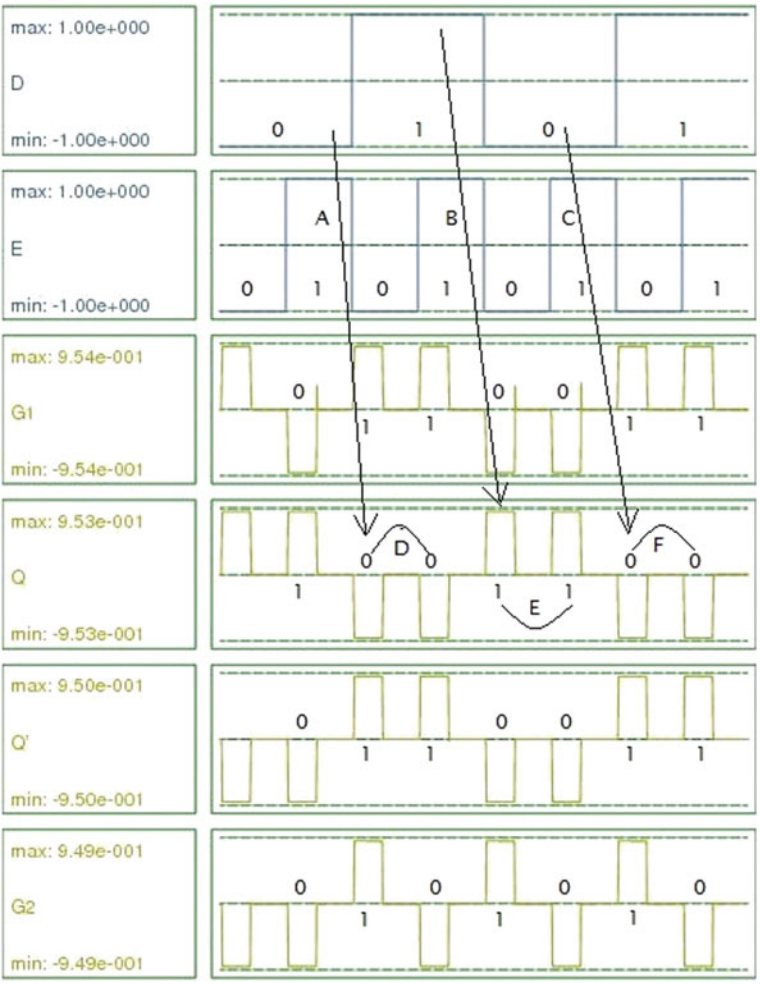

Fig. 16 Simulated output of the triple layer D latch (positive level triggered)

Table 6 Truth table of the D latch (positive level triggered)

Arrow	Input	Output (after one clock cycle)
A	E = High D = Low Q_n= High	Q_{n+1} = Low
B	E = High D = High Q_n = Low	Q_{n+1} = High
C	E = High D = Low Q_n = High	Q_{n+1} = Low
D	E = Low D = High Q_n = Low	Q_{n+1} = Low
E	E = Low D = Low Q_n= High	Q_{n+1} = High

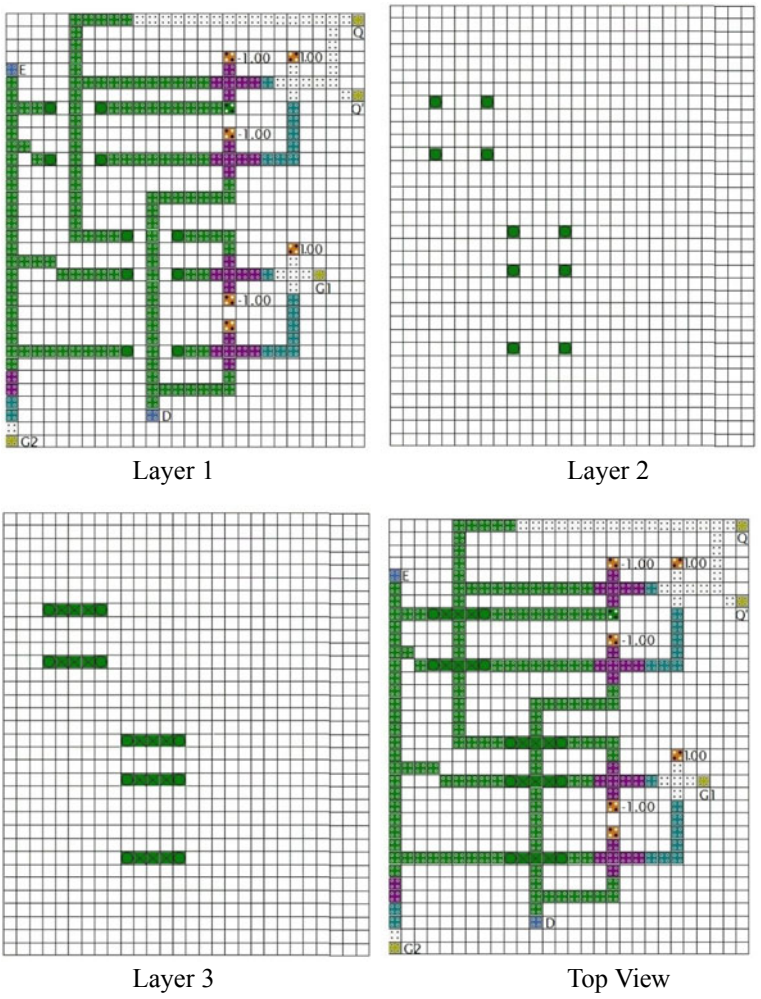

Fig. 17 Grid representation of proposed triple layer D latch (negative level triggered) with output Q based on multilayer Fredkin gate

represents when E = 0, T = 0, and $Q_n = 1$, then $Q_{n+1} = 1$. The trip of the arrow C represents when E = 1, T = 0, and $Q_n = 0$, then $Q_{n+1} = 0$.

Top view of the triple layer JK latch is represented in Fig. 21. The simulated results are shown in Fig. 22. To design the triple layer JK latch, 780 QCA cells, 1.50 μm^2 area and eight clock zones are needed, as two Fredkin gates are cascaded in series. Thus, the correct output will be generated after delay of two clock cycles. Table 9 which is compact form of Fig. 22, summaries the working functionality of proposed triple layer JK latch. The trip of arrow A represents when E = 1, J = 0, K = 1, and $Q_n = 0$, then $Q_{n+1} = 0$. The trip of arrow B represents when E = 1, J = 1, K = 0

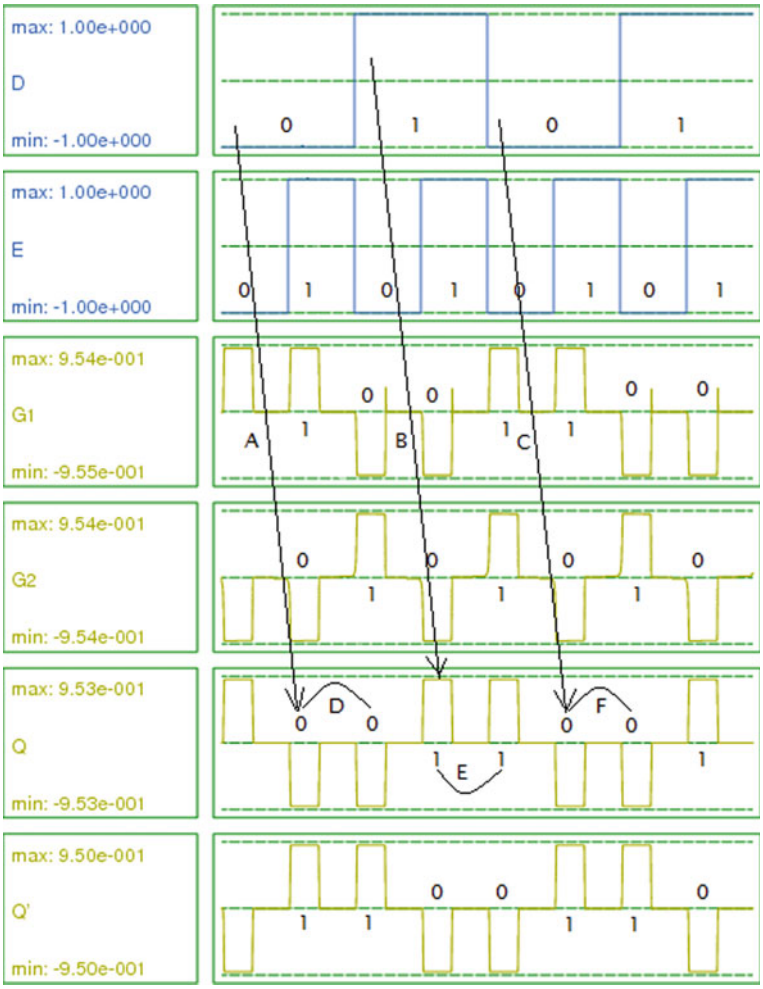

Fig. 18 Simulated results of triple layer D latch (negative level triggered)

Table 7 Truth table of the D latch (negative level triggered)

Arrow	Input	Output (after one clock cycle)
A	E = Low D = Low Q_n = High	Q_{n+1} = Low
B	E = Low D = High Q_n = Low	Q_{n+1} = High
C	E = Low D = Low Q_n = High	Q_{n+1} = Low
D	E = High D = Low Q_n = Low	Q_{n+1} = Low
E	E = High D = High Q_n = High	Q_{n+1} = High

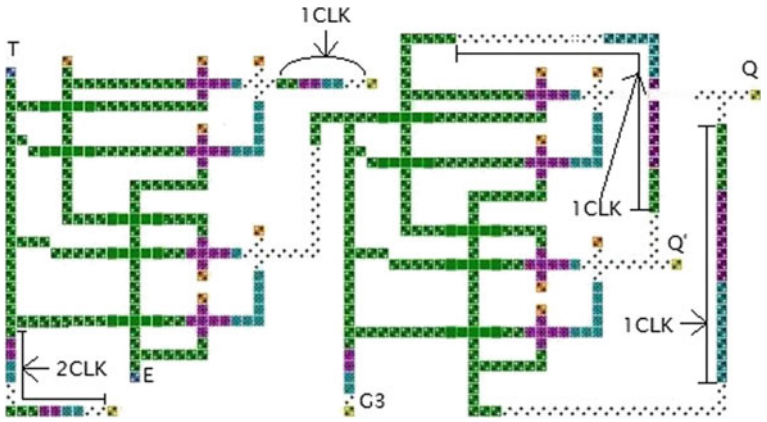

Fig. 19 Top view QCA layout of proposed triple layer T latch with output Q based on multilayer Fredkin gate

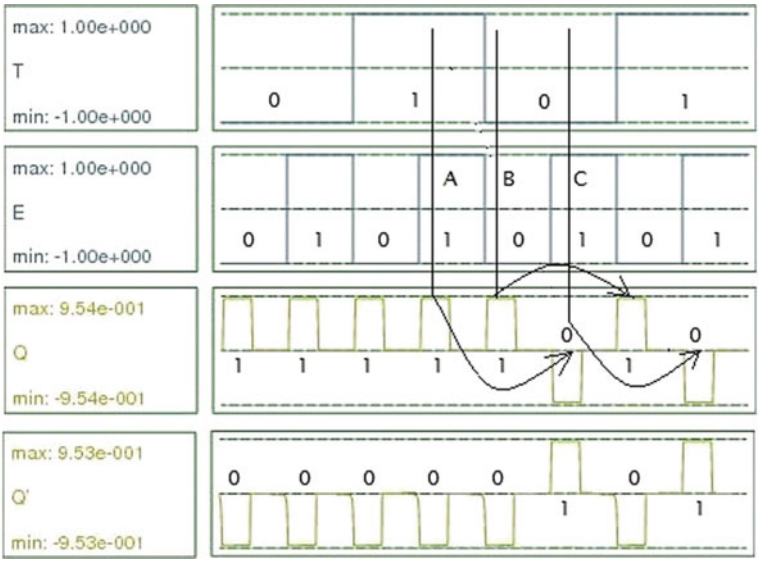

Fig. 20 Simulated results of triple layer T latch

Table 8 Truth table of the T latch

Arrow	Input	Output (after two clock cycles)
A	E = High D = High Q_n = High	Q_{n+1} = Low
B	E = Low D = Low Q_n = High	Q_{n+1} = High
C	E = High D = Low Q_n = Low	Q_{n+1} = Low

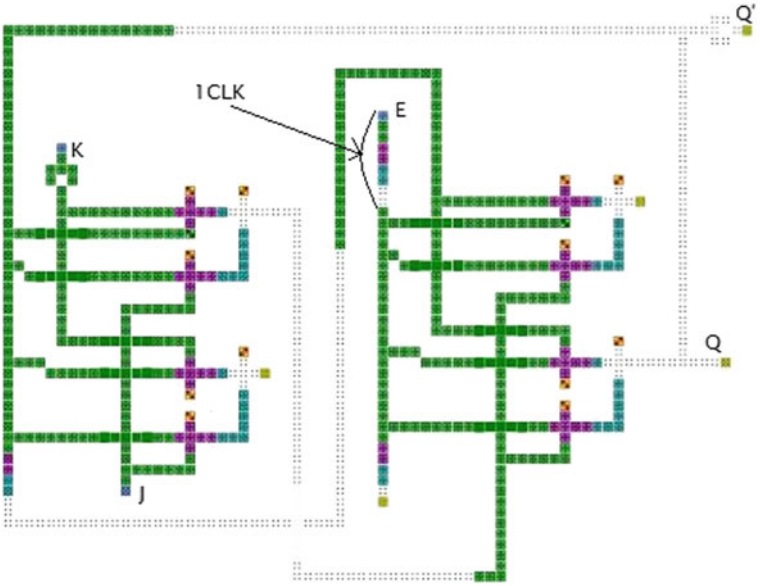

Fig. 21 Top view QCA layout of proposed triple layer JK latch with output Q using multilayer Fredkin gate

and $Q_n = 0$ then $Q_{n+1} = 1$. The trip of arrow C represents when E = 1, J = 1, K = 1, and $Q_n = 1$ then $Q_{n+1} = 0$. The trip of arrow D represents when E = 1, J = 0, K = 0, and $Q_n = 0$ then $Q_{n+1} = 0$. The trip of arrow E represents when E = 0, J = 1, K = 0, and $Q_n = 0$, then $Q_{n+1} = 0$.

Top view of QCA layout of the triple layer SR latch are presented in Fig. 23. The simulated results are shown in Fig. 24. To design the triple layer SR latch, 981 QCA cells, 2.17 μm^2 area and twelve clock zones are needed, as three Fredkin gates are cascaded in series. Thus, the correct output will be generated after delay of three clock cycles. Table 10 which is compact form of Fig. 24, summaries the working functionality of proposed triple layer SR latch. The trip of arrow A represents when E = 1, S = 1, R = 0 and $Q_n = 1$, then $Q_{n+1} = 1$. The trip of arrow B represents when E = 1, S = 0, R = 1and $Q_n = 1$ then $Q_{n+1} = 0$. The trip of arrow C represents when E = 0, S = 1, R = 1and $Q_n = 1$ then $Q_{n+1} = 1$. The trip of arrow D represents when E = 1, S = 0, R = 0, and $Q_n = 1$, then $Q_{n+1} = 1$.

Our multilayer latches are significant than the state-of-the-art latches in molecular QCA concerning with delays, Fredkin gates count, and the used number of QCA cells. A clock cycle delay is needed for each Fredkin gate in critical path. Comparison analysis of proposed multilayer latches in molecular QCA with existing one is mentioned in Table 11.

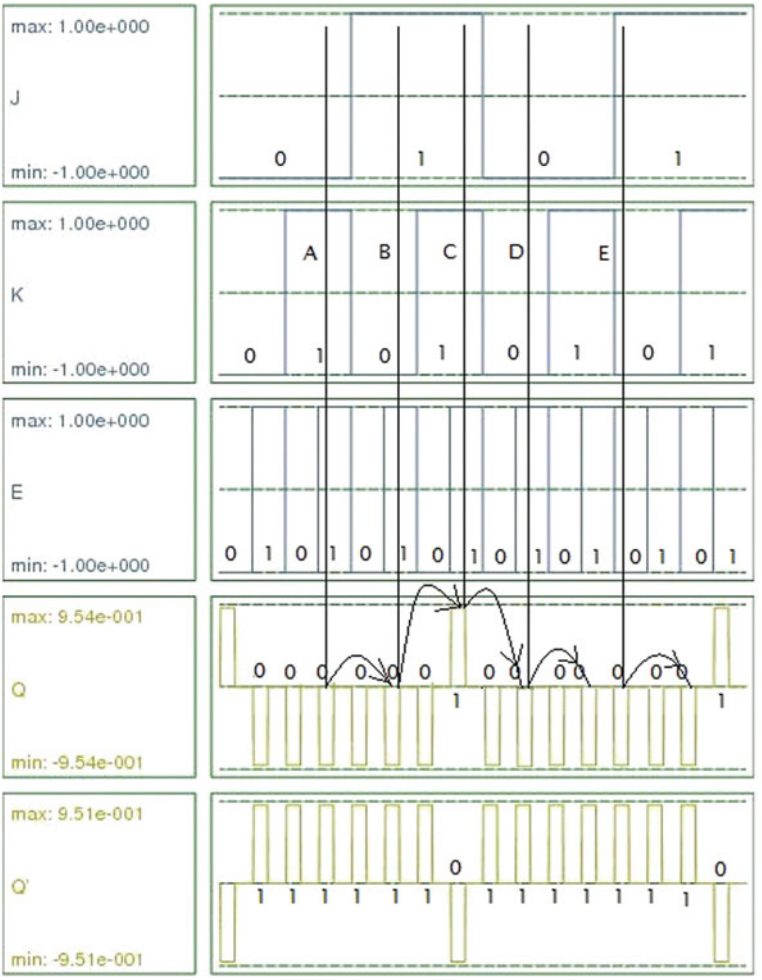

Fig. 22 Simulated results of triple layer JK latch

Table 9 Truth table of the JK latch

Arrow	Input	Output (after two clock cycles)
A	E = High J = Low K = High Q_n = Low	Q_{n+1} = Low
B	E = High J = High K = Low Q_n = Low	Q_{n+1} = High
C	E = High J = High K = High Q_n = High	Q_{n+1} = Low
D	E = High J = Low K = Low Q_n = Low	Q_{n+1} = Low
E	E = Low J = High K = Low Q_n = Low	Q_{n+1} = Low

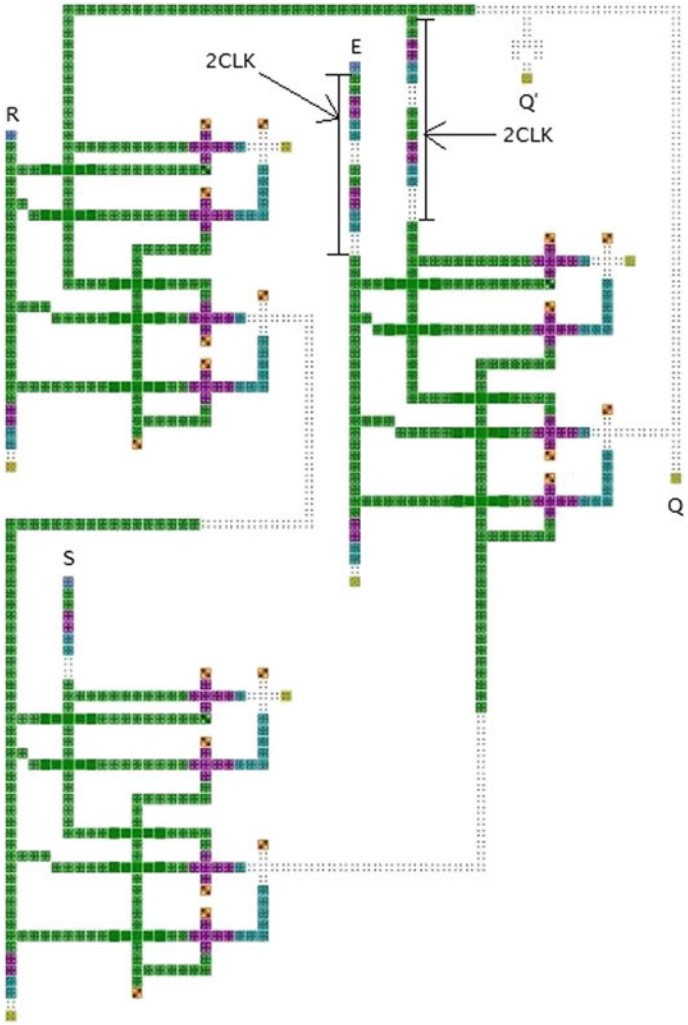

Fig. 23 Top view QCA layout of proposed triple layer SR latch with output Q based on multilayer Fredkin gate

6 Conclusions and Future Work

We propose online testable latches for molecular QCA using conservative logic (i.e., multilayer Fredkin gate). The proposed online testing approach concerning with parity preserving characteristics of the multilayer Fredkin gate without increasing the area of the circuit. It is beneficial for permanent fault as well as transient fault which can be detected by parity mismatch between inputs and outputs. $\overline{Q}$ may not be considered depending upon the prior information about circuit design, where

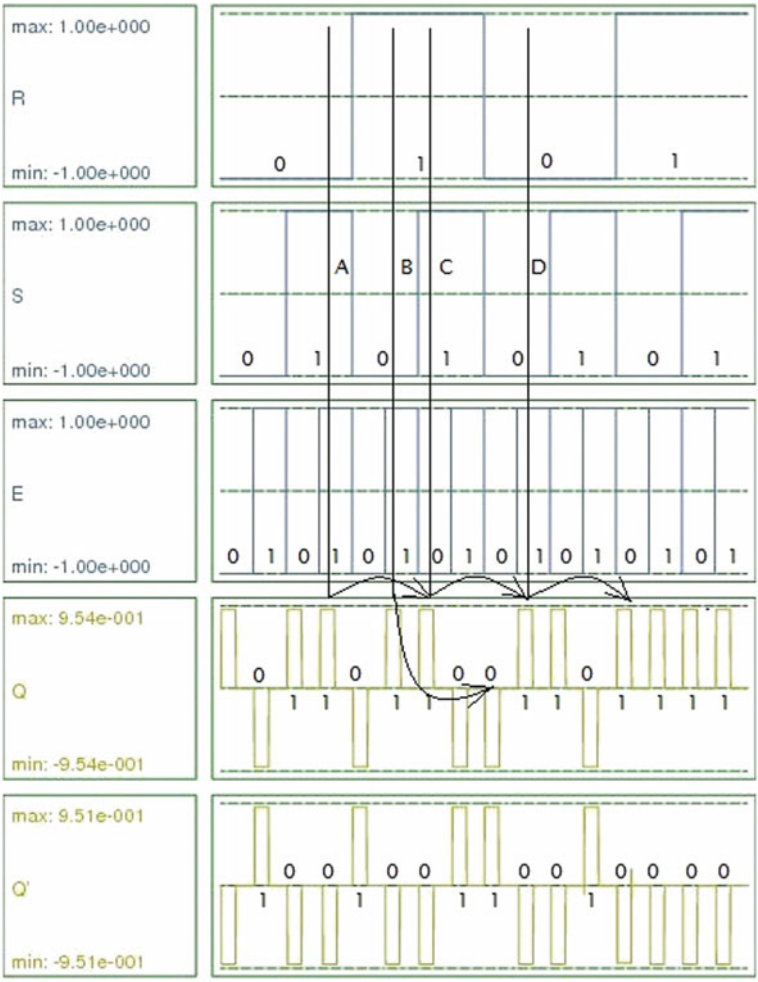

Fig. 24 Simulated results of triple layer SR latch

$\overline{Q}$ is introduced separately using NOT gate. For that case $\overline{Q}$ will be tested using knowledge of Q, as $\overline{Q}$ must be opposite of Q in fault-free case. The online testable designs for various multilayer latches, QCA layouts, and their simulation results are presented. The proposed methodology is applicable for the online detection of the single cell defect (missing/additional) model, single stuck-at fault model, or unidirectional faults. The proposed methodology is not appropriate to detect the bidirectional multiple faults, say expecting fault free output vector is {1110}, and due to bidirectional faults generated output vector is {1101} where parity of the input vector and output vector is same, thus fault is not detected. The input signals can be regenerated at output lines due to the reversible property. Thus, input vector

Table 10 Truth table of the SR latch

Arrow	Input	Output (after three clock cycles)
A	E = High S = High R = Low Q_n = High	Q_{n+1} = High
B	E = High S = Low R = High Q_n = High	Q_{n+1} = Low
C	E = Low S = High R = High Q_n = High	Q_{n+1} = High
D	E = High S = Low R = Low Q_n = High	Q_{n+1} = High

Table 11 Comparison analysis of proposed multilayer latches

	# of Fredkin gates in critical path		Required delays		# of QCA cells	
Various latches	In [10]	Proposed	In [10]	Proposed	In [10]	Proposed
D latch (positive level triggered) with output Q and Q'	2	1	2	1	598	292
D latch (negative level triggered) with output Q and Q'	None in literature	1	None in literature	1	None in literature	272
T latch with output Q and Q'	3	2	2	2	826	628
JK latch with output Q and Q'	4	2	4	2	1206	780
SR latch with output Q and Q'	4	3	4	3	1224	981

and output vector will not only preserve the parity, it must be exactly same. This characteristic can be implemented with our proposed approach to resolve the pitfall (detection of bidirectional multiple faults) of our proposed methodology in future, and also our proposed methodology can be implemented to design testable memory in molecular QCA framework. Finally, we can conclude that our proposed multilayer latches are implementation of online testable sequential circuit in molecular QCA framework with multilayer approach which is efficient respect to the area, delay, etc.

References

1. Tougaw P, Lent C (1994) Logical devices implemented using quantum cellular automata. J Appl Phys 75(3):1818–1825
2. Orlov AO, Amlani I, Bernstein GH, Lent CS, Snider GL (1997) Realization of a functional cell for quantum-dot cellular automata. Science 277:928–930
3. Tahoori MB, Huang J, Momenzadeh M, Lombardi F (2004) Testing of quantum cellular automata. IEEE Trans Nanotechnol 3(4):432–442
4. Landauer R (1961) Irreversibility and heat generation in the computational process. IBM J Res Dev 5:183–191
5. Bennett CH (1973) Logical reversibility of computation. IBM J Res Dev 17:525–532
6. Ma X, Huang J, Metra C, Lombardi F (2008) Reversible gates and testability of one dimensional arrays of molecular QCA. Springer J Electron Testing 24(1–3):297–311
7. Fredkin E, Toffoli T (1982) Conselvative logic. Int J Theor Phys 21:219–253

8. Momenzadeh M, Huang J, Lombardi F (2005) Defect characterization and tolerance of QCA sequential devices and circuits. In: Proceedings of the defect fault tolerance VLSI system, pp 199–207
9. Huang J, Momenzadeh M, Lombardi F (2007) Analysis of missing and additional cell defects in sequential quantum-dot cellular automata. Integr VLSI J 40(1), 503–515
10. Thapliyal H, Ranganathan N (2010) Reversible logic-based concurrently testable latches for molecular QCA. IEEE Trans Nanotech 9(1)
11. Gupta P, Jha NK, Lingappan L (2007) A test generation framework for quantum cellular automata circuits. IEEE Trans VLSI Syst 15(1)
12. Kummamuru RK, Orlov AO, Ramasubramaniam R, Lent CS, Bernstein GH, Snider GL (2003) Operation of a quantum-dot cellular automata (QCA), shift registers and analysis of errors. IEEE Trans Electron Devices 50(9):1906–1913
13. Schulhof G, Konrad W, Jullien G (2007) Simulation of random cell displacements in QCA. J Emerg Technol Comput Syst 3:1
14. Lent CS, Tougaw PD, Porod W, Bernstein GH (1993) Quantum cellular automata. Nanotech. 4:49–57
15. Orlov AO, Amlani I, Bernstein GH, Lent CS, Snider GL (1997) Realization of functional cell for QCA. Science 277, 928–930
16. Momenzadeh M, Ottavi M, Lombardi F (2005) Modeling QCA defects at molecular level in combinational circuits. In: Proceedings of the DFT VLSI System, Monterey, pp 208–216
17. Wei T, Wu K, Karri R, Orailoglu A (2005) Fault tolerant quantum cellular array (QCA) design using triple modular redundancy with shifted operands. In: Proceedings of the 2005 conference on Asia South Pacific design automation, Shanghai, China, pp 1192–1195
18. Bhanja S, Ottavi M, Pontarelli S, Lombardi F (2006) Novel designs for thermally robust coplanar crossing in QCA. In: Proceedings of the 2006 design, automation and test in Europe, Munich, Germany, pp 786–791
19. Sultana S, Imam SA, Radecka K (2006) Testing QCA modular logic. In: Proceedings of the 13th IEEE international conference on electronics circuits and systems, Nice, France, pp 700–703
20. Patel KN, Hayes JP, Markov IL (2004) Fault testing for reversible circuits. IEEE Trans Comput-Aided Design Integr Circuits Syst 23(8), 1220–1230 (2004)
21. Vasudevan DP, Lala PK, Parkerson JP (2006) Reversible-logic design with online testability. IEEE Trans Instrum Meas 55(2):406–414
22. Thapliyal H, Vinod AP (2010) Design of reversible sequential elements with feasibility of transistor implementation. In: Proceedings of the 2007 IEEE international symposium on circuits and systems, New Orleans, LA, pp 625–628
23. Chuang ML, Wang CY (2007) Synthesis of reversible sequential elements. In: Proceedings of the IEEE Asia south pacific design, automation in conference, Yokohama, Japan, pp 420–425
24. Rice JE (2006) A new look at reversible memory elements. In Proceedings of the international symposium on circuits and systems, Kos, Greece, pp 243–246
25. Banik D, Bhattacharyya P, Ekbal A (2016) Rule based hardware approach for machine transliteration: a first thought. In: 2016 Sixth international symposium on embedded computing and system design (ISED). IEEE
26. Banik D et al (2016) Can SMT and RBMT Improve each other's performance?-An experiment with English-Hindi translation. In: Proceedings of the 13th international conference on natural language processing
27. Sen S et al (2016) IITP English-Hindi machine translation system at wat 2016. In: Proceedings of the 3rd workshop on Asian translation (WAT2016)
28. Banik D, Ekbal A, Bhattacharyya P (2019) Machine learning based optimized pruning approach for decoding in statistical machine translation. IEEE Access 7:1736–1751
29. Mondal S, Banik D (2013) Energy efficient fault tolerant scheme for wireless sensor networks (EEFSWSN). IJCSET 3
30. Banik D, Mathew J, Rahamant H (2016) Testable reversible latch in molecular quantum dot cellular automata framework. In: India conference (INDICON), 2016 IEEE annual. IEEE

31. Banik D (2016) Design of low cost latches based on reversible quantum dot cellular automata. In: 2016 Sixth international symposium on embedded computing and system design (ISED). IEEE
32. Banik D (2016) Efficient methodology for testable reversible sequential circuit design. In: IEEE international conference on power electronics, intelligent control and energy systems (ICPEICES). IEEE
33. Banik D, Chowdhury RR (2016) Offline signature authentication: a back propagation-neural network approach. In: 2016 Sixth international symposium on embedded computing and system design (ISED). IEEE
34. Ottavi M, Schiano L, Lombardi F (2006) HDLQ: A HDL environment for QCA design. ACM J Emerg Technol 2(4):243–261
35. Walus K, Dysart T, Jullien GA, Budiman R (2004) QCADesigner: a rapid design and simulation tool for quantum-dot-cellular automata. Trans Nanotechnol 3(1):26–29
36. Kim K, Wu K, Karri R (2006) Quantum-dot cellular automata design guideline. IEICE Trans Fund E89-A(6), 1607–1614