Conservative QCA Gate (CQCA) for Designing Concurrently Testable Molecular QCA Circuits

Himanshu Thapliyal and Nagarajan Ranganathan
Department of Computer Science and Engineering,
University of South Florida, Tampa, FL, USA
E-mail : {hthapliy,ranganat}@cse.usf.edu

Abstract

Nanocircuits based on molecular QCA are prone to high error rates. In this paper, we present a novel conservative logic gate termed 'CQCA' (conservative QCA) to design concurrently testable circuits for molecular QCA. In conservative logic gates, there would be an equal number of 1s in the output as there would be on the input. Thus, conservative logic gates are parity preserving, that is, the parity of the input vectors is equal to the output vectors. CQCA is proposed in this work as molecular QCA is based on majority voting. We analyzed the fault patterns in existing popular conservative Fredkin gate and proposed CQCA gate due to single missing/additional cell defect in molecular QCA. We found that if there is a fault in molecular QCA implementation of Fredkin and CQCA gates, there is a parity mismatch between the input and the output; otherwise the input parity is same as output parity. Thus, any permanent and transient fault in molecular QCA can be concurrently detected if implemented with conservative Fredkin and CQCA gates. We applied novel method of using majority and minority voting to detect the fault in conservative gates. We propose to use CQCA gate compared to existing popular Fredkin gate as CQCA excels Fredkin gate in parameters of complexity(number of majority voter), speed and area. The results are well supported by synthesizing standard benchmark combinational functions. The QCA design of 2 pair 2 rail checker is also presented for the first time ever in literature. The design of QCA layouts and the verification of the designs are performed using the QCADesigner and HDLQ tools.

1. Introduction

The existing CMOS technology is reaching its limits beyond which the down scaling in feature size and proper working of the device is becoming extremely difficult. CMOS devices suffer from heat generation as they have to discharge all the stored energy when flipping from 1 to 0. Quantum dot cellular automata (QCA) is one of the emerging nanotechnologies in which it is possible to achieve circuit densities and clock frequencies much beyond the limit of existing CMOS technology. QCA has significant advantage in terms of power dissipation as it does not have to dissipate all its signal energy hence considered as one of the promising technologies to achieve the thermodynamic limit of computation [1-2]. The basic QCA logic devices comprise the majority voter (MV), the inverter (INV), binary wire and the inverter chain. Figure 1 shows the basic QCA cell and logic devices.

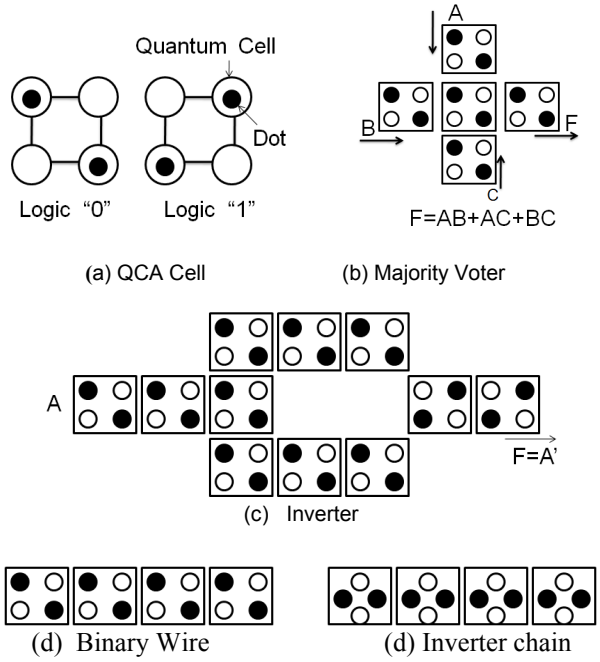


Figure 1. Basic QCA Devices

The Launder four phase clocking scheme is generally used in QCA design, and the present work is also based on this. Due to significant error rates in nano-scale manufacturing, nanotechnologies including

1063-9667/09 $25.00 © 2009 IEEE
DOI 10.1109/VLSI.Design.2009.75

QCA require extremely low device error rate [5].In manufacturing QCA, defects can occur in the synthesis and deposition phases. However, defects are more likely to take place during the deposition phase [6]. QCA devices are also prone to transient faults caused by thermodynamic effects, radiation and other effects, as the energy difference between the ground and the excited state is small [10,11]. Thus, in literature researchers have used the novel concepts such as reversible logic to improve the testability of molecular QCA [5].

In this work, we explore the concept of conservative logic gates as a means for designing concurrently testable circuits for molecular QCA. Conservative logic gates have equal number of 1s in the output as there would be on the input [15]. Thus, they are parity preserving, that is, the parity of the input is always equal to the parity of the output. As molecular QCA is based on majority voting, the design based on conservative logic will be completely different from conventional CMOS designs. In literature, conservative Fredkin gate is the most popular gate but we find that Fredkin gate is not suitable for all molecular QCA designs as the designs based on it requires more clocking zones, majority gates and area. Since QCA logic is based on majority voting, this led us to propose CQCA gate (conservative QCA gate). CQCA requires only two clocking zones compared to four clocking zones required by the Fredkin gate and it requires only two majority gates compared to six majority gates required by the Fredkin gate. The benefits of using CQCA over Fredkin gate in terms of area and delay is shown by synthesizing standard combinational benchmark functions. To demonstrate the effectiveness of conservative Fredkin and CQCA gates for concurrently testable molecular QCA design, we have done the fault pattern study of conservative Fredkin and CQCA gates due to single missing/additional cell defect in QCA. We found that when there is permanent fault due to above defects, there is parity mismatch between the input and the output of the conservative Fredkin and CQCA gates. Due to parity preserving property, any permanent and transient fault in molecular QCA can be concurrently detected. We demonstrated a novel strategy of using majority and minority voter gates to detect parity mismatch between the input and output of the 3 input 3 output conservative gates instead of XOR gates, as it is costly to implement XOR function in QCA compared to majority/minority voter. The QCA design of 2 pair 2 rail checker is also presented for the first time ever in literature. Thus, this work lays the foundation of concurrent testing of molecular QCA which is susceptible to high error rates.

The paper is organized as follows: Section 2 presents the QCA defects and related work; Section 3 presents existing and proposed conservative gates. Section 4 presents concurrent testing of molecular QCA using conservative logic gates; Section 5 shows the QCA design of 2 pair 2 rail checker; Section 6 presents the comparison between Fredkin and CQCA conservative gates; Section 7 provides the conclusions.

2. Background and related work

In manufacturing QCA, defects can occur in the synthesis and deposition phases. However, defects are more likely to take place during the deposition phase [6]. Researchers assume that QCA cells have no manufacturing defects and in metal QCA faults occur due to cell misplacement. These defects can be characterized as cell displacement, cell misalignment and cell omission [7]. Researchers have proved that molecular QCA cells are more susceptible to missing/additional QCA cell defects [8,17]. Additional cell defect is due to the deposition of an additional cell on the substrate while missing cell defect is due to loss of a particular cell

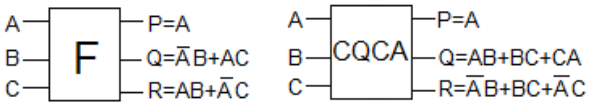
2.1. Related work

The testing of QCA is first time addressed in a seminal work in [6]. In [6], the defect characterization of QCA devices is investigated and is shown how the testing of QCA differs from conventional CMOS. In [8], the modeling of QCA defects at molecular level is done for combinational circuits. Fault characterization is done for single missing/ additional cell defect on different QCA devices such as MV, INV, fan-out, Crosswire and L-shape wire. In [7], test generation framework for QCA is presented. It is seen that additional test vectors can be generated for detecting QCA defects which remain undetected by stuck-at fault model. Bridging fault on QCA wires is also addressed. In [5], reversible logic is used to detect single missing/additional cell defects. It is seen that reversible 1D array is C-testable. In [14], fault-tolerant QCA designs are presented using triple modular redundancy with shifted operands. The strategy is proposed considering the wire delay and faults in wires in QCA.

3. Conservative gates for QCA

There is an existing popular conservative gate called Fredkin gate [15]. Fredkin gate is shown in Fig.2.a. Fredkin gate can be described as mapping (A, B, C)

to (P=A, Q=A'B+AC, R=AB+A'C), where A, B, C are input and P, Q, R are output, respectively. Fredkin gate produces the same number of 1s in the output as on the input. The QCA design of Fredkin gate is shown in Fig. 3 using four-phase clocking scheme, in which the clocking zone is shown by the number next to D (D0 means clock 0 zone, D1 means clock 1 zone and so on, MV in the figure represents majority voter). Thus, it can be seen that Fredkin gate has two level majority voter (MV) implementation and it requires 6 MVs to implement it.



(a). Fredkin gate (b). Proposed CQCA gate

Figure 2. Conservative QCA gates

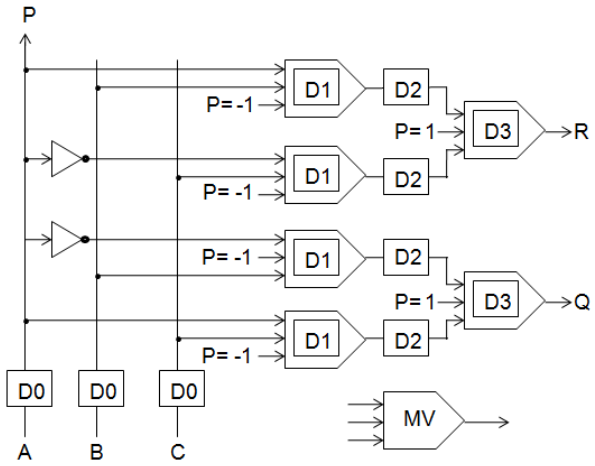


Figure 3. Fredkin gate QCA design (D0 to D3 represent clock zones 0 to 3)

Table 1. Truth table of CQCA gate

A	B	C	P	Q	R
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	0	1
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	1	0
1	1	1	1	1	1

3.1. Proposed CQCA gate

The existing conservative Fredkin gate is costly in molecular QCA implementation. This led us to propose novel conservative gate especially suiting molecular QCA MV (majority voter) based design and is termed CQCA (Conservative QCA gate). The input to output mapping of CQCA is: P=A; Q=AB+BC+AC [MV(A,B,C)]; R=A'B+A'C+BC [MV(A',B,C)], where A, B, C are input and P, Q, R are output, respectively. Figure 2.b shows the block diagram representation of CQCA gate. Table 1 shows the truth table of the CQCA gate. It shows that it has the same number of 1's in the input as in the output. Figure 4 shows the QCA implementation of CQCA gate. It is seen that the CQCA can be implemented with one level MV logic and requires only two MVs to implement it. A detailed comparison between CQCA and Fredkin gate is presented in Section 6.

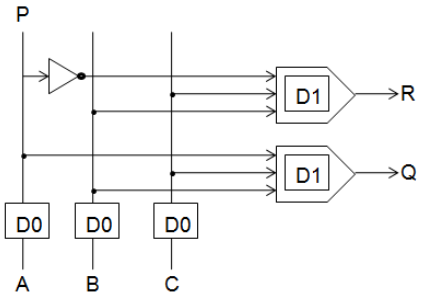


Figure 4. QCA implementation of CQCA gate

4. Concurrent testing of molecular QCA with conservative gates

To the best of our knowledge and as far as existing literature is concerned, concurrent testing for molecular QCA designs has never been addressed and the proposed work is the first attempt in this direction. In this work, our analysis is also based on missing/additional QCA cell defects. Figure 5 shows the QCA layout of the CQCA gate (the QCA layout of the Fredkin gate is described in [12]). We have modeled the Fredkin and CQCA gates QCA layouts, with the presence of all possible single missing/additional cell defects in MV, INV, fan-out, Crosswire and L-shape wire [8]. The modeling is done using HDLQ [16], a design tool which provides the Verilog HDL library of QCA devices, i.e., MV, INV, fan-out, Crosswire, L-shape wire with fault injection capability. The design is simulated in Verilog HDL simulator in the presence of faults to determine the corresponding output.

The exhaustive testing of the Fredkin and CQCA gates with 8 input patterns and all possible single missing/additional cell defects is done using the Active HDL simulator. The exhaustive testing for Fredkin gate generated 20 unique fault patterns. The

exhaustive testing of CQCA generated 8 unique fault patterns as shown in Table 2. In the fault patterns study in Table 2, ai is the 3 bit pattern having an equivalent decimal value of i, for example a0 represents 000 (decimal 0) and a7 represents 111(decimal 7).We carefully observe each fault pattern and found that in the occurrence of a fault, there is a parity mismatch between the output and the input of the Fredkin and CQCA gates (i.e., parity of the input vector is not equal to the output vector). This led us to conclude that Fredkin and CQCA gates can detect concurrently permanent fault by matching the parity. Since Fredkin and CQCA gates are logically parity preserving, they can detect also the transient faults. Hence, Fredkin and CQCA gates can concurrently detect permanent as well as transient fault based on parity preserving in molecular QCA. In CMOS circuits, parity match is checked as A⊕B⊕C=P⊕Q⊕R.. However, implementing the XOR gate in QCA is costly as the process requires 3 majority gates. In QCA, implementing A⊕B⊕C would require 6 majority gates and similarly P⊕Q⊕R would require 6 majority gates. Thus, comparing A⊕B⊕C=P⊕Q⊕R would require a total of 12 majority gates.

In order to check the parity mismatch we propose an alternative strategy to use majority voter for input vector and minority voter for output vector. Let D=MV(A,B,C) where D is the output of the majority gate and A,B,C are the input of the CQCA gate. Let S=mV(P,Q,R) where S is the output of the minority voter and P,Q,R are the output of the CQCA gate. Thus when there is no fault D will be complementary to S, and when there is a fault D will be same as S. The minority voter required can be designed by complementing the majority voter or as a novel design proposed in [13]. The proposed approach requires only 2 majority gates to compare the input and output of conservative CQCA gate. An example of the proposed approach is demonstrated for CQCA gate in Fig. 6 (The strategy is also applicable for Fredkin gate). The CQCA gate along with the majority and minority voter will be referred to as conservative testable block (CTB) in this paper. The reason for generating S and D as complementary in case of fault free condition is to make use of 2 pair 2 rail checker in comparing them. It can be argued that majority and minority voter used for generating D and S may have faults leading to incorrect results. Hence fault-tolerant majority gate are required, one of its design is described in [3]. The correct outputs D and S can be also be generated by using triple modular redundancy approach (TMR) for majority and minority voting [14]. The design of 2 pair 2 rail checker and its use to detect the fault is discussed in the next section.

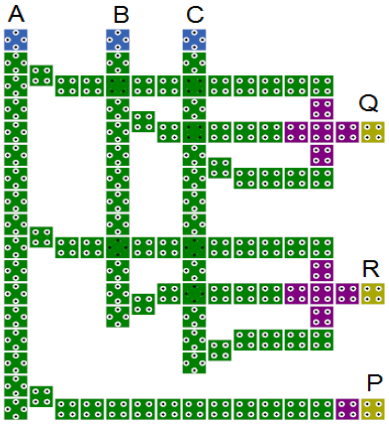


Figure 5. QCA layout of CQCA gate

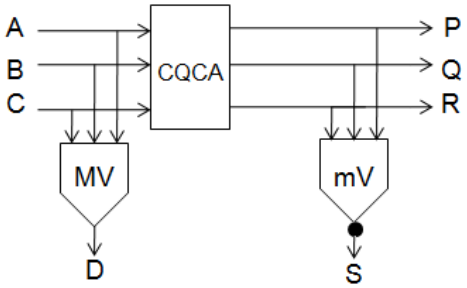


Figure 6. Conservative testable block

Table 2. Fault patterns in CQCA Gate

Input Vector	Fault Free	Fault Patterns							
		1	2	3	4	5	6	7	8
a0	a0	a0	a0	a0	a0	a0	a2	a0	a0
a1	a1	a3	a0	a3	a1	a1	a1	a0	a0
a2	a1	a3	a0	a1	a3	a3	a3	a1	a1
a3	a3	a3	a3	a1	a1	a3	a3	a3	a2
a4	a4	a4	a4	a6	a6	a4	a4	a4	a5
a5	a6	a4	a7	a6	a4	a4	a4	a6	a6
a6	a6	a4	a7	a4	a6	a6	a6	a7	a7
a7	a7	a7	a7	a7	a7	a7	a5	a7	a7

5. 2 pair 2 rail checker

The 2 pair 2 rail checker is required for testing that the output D and S generated by the majority and minority voting, respectively, are complementary or not. The error checking functions required in the 2 pair rail checker are E1=X0Y1+Y0X1 and E2= X0X1+Y0Y1; where X0/Y0 & X1/Y1 are complementary.

The 2 pair 2 rail checker produces the complementary output at E1 & E2 if the input passed to it are complementary. If the input are not complementary, the output E1 & E2 will be identical. We are also presenting a design of 2 pair 2 rail checker

based on MV QCA gate in Fig. 7. To the best of our knowledge, this is the first ever reported QCA design of 2 pair 2 rail checker. This design features 6 MV gates. Figure 8 shows the testing of conservative testable blocks (CTB) as described in Section 4 with 2 pair 2 rail checker. The output D and S (D1 and S1 in Fig. 8) of one testable block will be the input X0 and Y0 of the 2 pair rail checker. The other testable block output, D and S (D2 and S2 in Fig. 8) will form the other input X1 and Y1. Thus, 2 pair 2 rail checker can check 2 testable blocks at a time. The cascading of the 2 pair 2 rail checkers is done in a tree fashion, so that only the final 2 output are externally observable.

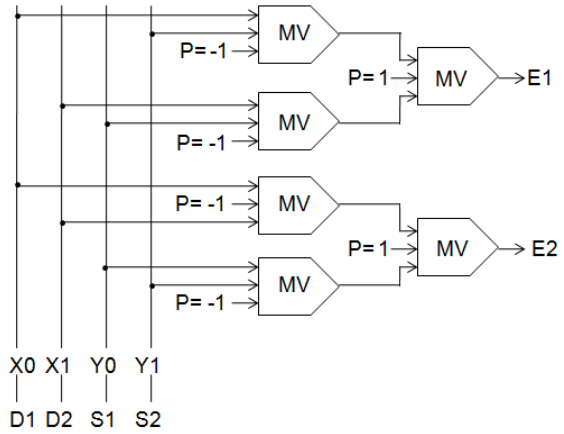


Figure 7. QCA design of 2 pair 2 rail checker (MV represents majority voter)

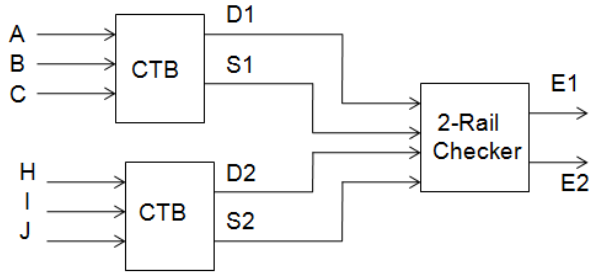


Figure 8. CTB testing using 2 pair 2 rail checker (2 CTBs can be tested with 1 checker)

6. Comparison of Fredkin and CQCA gate

Table 3 shows the comparison between the Fredkin and CQCA gates. In Table 3, since the number of clocking zones required to design CQCA conservative gate is less, it will be faster compared to Fredkin gate. The total number of QCA cell required in CQCA gate is only 47% of cells required by Fredkin gate and the area occupied by CQCA is only 29% of the area occupied by Fredkin gate (Fredkin gate requires 246 QCA cell with the area of 0.37 um^2 while CQCA requires 117 QCA cell with the area of 0.11um^2). Thus, the proposed CQCA gate excels Fredkin gate in all aspects.

6.1. Simulations for verification

The designs were verified using QCADesigner ver. 2.0.3 [9]. In the bistable approximation, we used the following parameters: cell size=18 nm, number of samples=182800, convergence tolerance=0.001000, radius of effect=41 nm, relative permittivity= 12.9, clock high=9.8e-22, clock low=3.8e-23, clock amplitude factor=2.000, layer separation=11.5000nm, maximum iterations per sample=1000.

Table 3. A comparison of Fredkin and CQCA

	Fredkin	CQCA
Clk Zs	4	2
MVs	6	2
Total Cells	246	117
Area	0.37um^2 , where L=0.4812um and W= 0.76984um	0.11um^2 , where L=0.30012 um W= 0.36454um

6.2. Comparison on benchmark functions

In order to have to have the comparison of the Fredkin and proposed CQCA gate for logic synthesis, we have implemented thirteen standard three variable Boolean combinational functions proposed in [4] for molecular QCA. These thirteen functions cover all the 256 Boolean functions for three variables. Table 4 shows the comparison between the two by synthesizing these 13 standard functions. It requires a total of 246 MVs and 136 clock zones to implement the standard functions using Fredkin gate. While it requires only 86 MVs and 62 clock zones when these standard functions are implemented with proposed CQCA gate. Thus implementing with CQCA achieves a reduction of 65% and 54.4% in terms of MVs and clock zones, respectively, which shows that CQCA gate performs better than Fredkin gate in terms of speed and area.

7. Conclusions

We propose the use of conservative logic gates to design concurrently testable circuits for molecular QCA. CQCA gate as presented shows that it is better than most popular Fredkin gate in terms of area and speed. The results are supported by synthesizing standard benchmark functions.

Table 4. Synthesis comparison of thirteen standard functions

	Standard Function	Fredkin Implementation			CQCA Implementation		
		# of Fredkin	# of MVs	Clk Zs	# of CQCA	# of MVs	Clk Zs
1	F=ABC	2	12	8	2	4	4
2	F=AB	1	6	4	1	2	2
3	F=ABC+AB'C'	3	18	12	3	6	4
4	F=ABC+A'B'C'	4	24	12	6	12	8
5	F=AB+BC	2	12	8	2	4	4
6	F=AB+A'B'C	5	30	16	5	10	8
7	F=ABC+A'BC'+AB'C'	6	36	16	6	12	6
8	F=A	1	6	4	1	2	2
9	F=AB+BC+AC	5	30	16	1	2	2
10	F=AB+B'C	1	6	4	3	6	4
11	F=AB+BC+A'B'C'	6	36	16	6	12	8
12	F=AB+A'B'	2	12	8	4	8	6
13	F=ABC+A'B'C+AB'C'+A'BC'	3	18	12	3	6	4
	Total	41	246	136	43	86	62

A novel strategy of majority and minority voting mismatch is proposed to detect fault in the input and output of the conservative gates for molecular QCA. The design of 2 pair 2 rail checker is presented for molecular QCA for the first time ever time in literature. In conclusion, the proposed CQCA gate is of great importance to fault susceptible molecular QCA nano-computing.

8. References

[1] A. O. Orlov, I. Amlani, G. H. Bernstein, C. S. Lent, and G. L. Snider, "Realization of a functional cell for quantum-dot cellular automata," Science,vol. 277, pp. 928–930, 1997.

[2] P. Tougaw and C. Lent, "Logical devices implemented using quantum cellular automata," J. Appl. Phys., vol. 75, no. 3, pp. 1818–1825, 1994.

[3] A. Fijany, B.N. Toomarian, "New Design for Quantum Dots Cellular Automata to Obtain Fault Tolerant Logic Gates", Journal of Nanoparticle Research, vol. 3, pp. 27-37, Feb. 2001.

[4] R. Zhang, K. Walus, W. Wang, and G. A. Jullien, "A method of majority logic reduction for quantum cellular automata," IEEE Trans. Nanotechnol., vol. 3, no. 4, pp. 443--450, Dec. 2004.

[5] X. Ma, J. Huang, C. Metra, F.Lombardi, "Reversible Gates and Testability of One Dimensional Arrays of Molecular QCA", Springer Journal of Electronic Testing, Vol.24, No. 1-3, pp. 297-311,June, 2008.

[6] M. B. Tahoori, J. Huang, M. Momenzadeh, and F. Lombardi, "Testing of quantum cellular automata," IEEE Trans. Nanotechnol., vol. 3, no.4, pp. 432–442, Dec. 2004.

[7] P.Gupta, N.K. Jha, L.Lingappan, "A Test Generation Framework for Quantum Cellular Automata Circuits", IEEE Trans. VLSI Syst., Vol. 15, no.1,pp. 24-36, Jan 2007.

[8] M. Momenzadeh, M. Ottavi, F. Lombardi, "Modeling QCA defects at molecular level in combinational circuits", Proc. DFT in VLSI Systems, Monterey, CA, USA, 3-5 Oct 2005, pp. 208–216.

[9] http://www.qcadesigner.ca/

[10] T. Tanamoto et al.,"Quantum effect device", United States Patent 5710436, Jan 1998.

[11] R. K. Kummamuru et al., "Operation of a Quantum-Dot Cellular Automata (QCA),Shift Registers and Analysis of Errors", IEEE TRANS. Electron Devices, Vol. 50, No. 9, pp 1906-1913, Sep. 2003

[12] H. Thapliyal and N. Ranganathan, "Testable Reversible Latches for Molecular QCA", Proc. IEEE NANO 2008, Arlington,TX, Aug 2008, pp. 699-702. (invited paper after peer review process)

[13] S. Roy, B.Saha and B.K Sikdar,"Design and Characterization of Minority Gate as a Universal Logic for Quantum-Dot Cellular Automata", Journal of Computational and Theoretical Nanoscience, Vol.3, No.5, pp. 684-695,Oct 2006.

[14] T. Wei,K. Wu, R.Karri and A. Orailoglu, "Fault tolerant quantum cellular array (QCA) design using Triple Modular Redundancy with shifted operands",Proc. ASPDAC 2005, Shanghai, China, Jan 2005, pp.1192-1195

[15] E. Fredkin, T Toffoli, "Conservative Logic", International Journal of Theor. Physics, 21(1982),pp.219-253.

[16] M. Ottavi, L. Schiano, and F. Lombardi, "HDLQ: A HDL Environment for QCA Design", ACM Journal on Emerging Technologies in Computing Systems, Vol. 2, No. 4, pp. 243–261, Oct. 2006.

[17] M. Momenzadeh, J. Huang and F. Lombardi, "Defect Characterization and Tolerance of QCA Sequential Devices and Circuits," Proc. Defect and Fault Tolerance in VLSI Systems, CA, Oct 2005. pp. 199-207.