Competent Universal Reversible Logic Gate Design for Quantum dot Cellular Automata

KUNAL DAS[1,4], DEBASHIS DE[2,3] AND MALLIKA DE[4]

[1]B.P.Poddar Institute of Management and Technology, 137, VIP Road, Kol-700052, India
[2]Department of Computer Science and Engineering,
West Bengal University of Technology, BF-142, Sector-I, Salt Lake City, Kolkata 700064, India
[3]School of Physics, University of Western Australia, M013,
35 Stirling Highway, crawley, Perth, WA 6009, Australia
[4]Department of Engg.and Technological Studies, Kalyani University, Kalyani 741235, West Bengal, India
kunaldasqca@gmail.com members.nanosociety.us/kunaldas

Abstract: - Quantum dot cellular Automata (QCA) is leading technology for alternative of CMOS design. Reversible Logic design is found to be Low power design which becomes emerging technology in Low power Nanotechnology era. In this work we devoted to design a Reversible Logic Gate which is a universal gate (known as URLG) and can be design with alternative of co-planer cross over wire. We introduce a passive type 3×3 tile that will also helpful for design the co-planner crossover wire, implements Reversible gate. We analyzed the application of passive type tile with the implementation of proposed URLG. The reversible gate design proposed in this paper which is proved to be universal by means of designing all classical logic gate i.e. AND, OR, NOT, NAND, NOR and XOR. During design of URLG we made attention is to reduce the no.'s of cells as well as to reduce the area which is found to be 0.16μm2 with conventional co-planner crossover wire and 0.19 μm2 area with new proposed passive 3×3 tile and compared to Fredkin gate it achieved 48%(on average) reduction in area (area of Fredkin gate is 0.37 μm2). We also tried made focus on 'Garbage Minimization' Problem. Here, we demonstrate that the design all Logic gate generate low garbage output as we make a comparison with NFT gate. We achieve 20% reduction in garbage count. The competent design of working Temperature, Fault Tolerant URLG is useful to implement all combinational Logic circuit as well as sequential circuit. Hence, it shall be innovative, effective with respect to previous report.

Keywords: Basic QCA, Majority Voter (MV), And-Or-Inverter (AOI), Reversible Logic Gate, Garbage Count.

1 Introduction

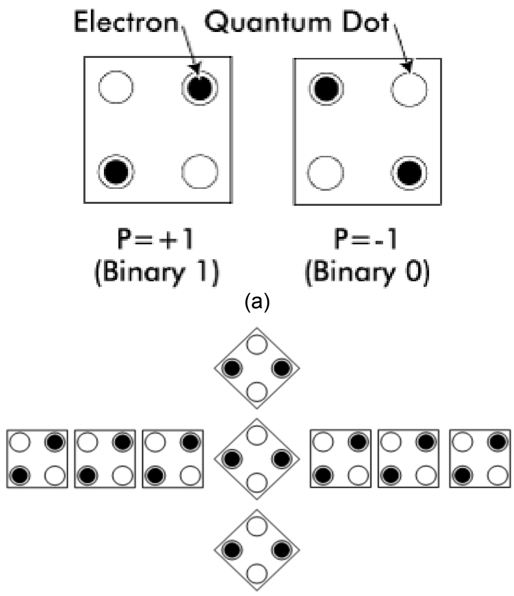
The Complementary Metal Oxide Semiconductor (CMOS) provides micro scale computing, with high density and low power very large-scale integration (VLSI) system. However such system has found many problems like high leakage current, high lithography cost, and limitation of speed in GHz range. As a result we have to find out a strong alternative of CMOS system within next 15 to 20 years. One of strong alternative technology is a Quantum-dot cellular automaton, [1][2]. Quantum-dot Cellular Automata (QCA) [3-8] first proposed in 1993 by University of Notre Dame researchers C.S. Lent et al. [2]. QCA is based upon the physics of cell-to-cell interaction due to the rearrangement of electrons positions [7-14]. Low power design is become primary goal of Very Large-Scale Integration (VLSI). Traditionally classical logic circuit which is found to be 'irreversible logic circuit' dissipate heat energy in an order KTln2 joules that the loss of per bit of information, where K (K=1.3806505*10-23JK-1) is Boltzmann's constant and T is absolute Temperature [15]. Bennett shows that in case of Reversible logic computation KTln2 joules energy will not dissipate [16]. Hence Reversible logic design naturally gets priority to design combinational as well as sequential circuit. In this Low Power design era Reversible circuit design is applicable for VLSI in CMOS, Quantum computing, DNA computing as well as Quantum dot cellular automata (QCA). In QCA [17][18], there are not much progress had been noticed, a few bunch of proposal had been found. In Quantum computing we found there are many proposal on Reversible Logic Gate (RLG) design like Feynman Gate [19], Toffili Gate [20], Fredkin Gate [21]. Very recently Haghparast and Navi Proposed NFT Gate for

Nanotechnology based system [22]. In this proposal, it mainly addressed 'Garbage Minimization' problem [23] for implementing all basic logic gate. In this paper, we made attention to design a Universal Reversible logic gate that must be able to implement all reversible basic gate like AND, OR, NAND, NOR, NOT and XOR gate. In this design we also addressed the 'Garbage Minimization' problem. As we know the more No's of Constant Input leads to more Garbage output, in this design of all logic gate we tried to reduce constant input as much as possible. Here we made a comparison with recently made NFT [22] and achieve around 20% of reduction in Garbage count.

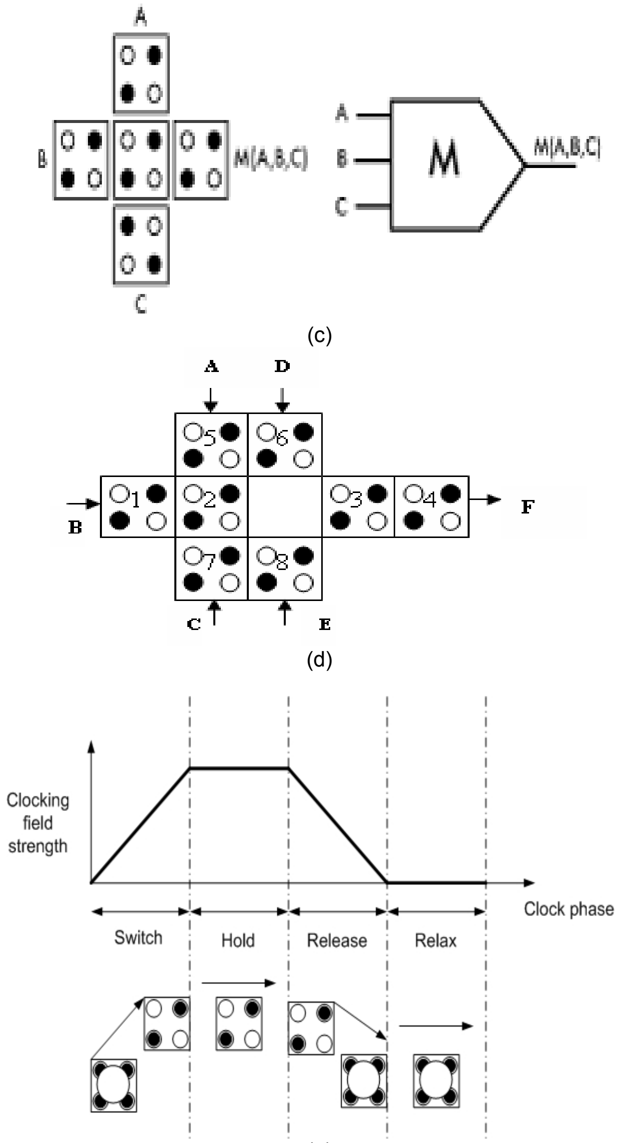
The rest of this paper is organized as follows. Section 2 is dedicated to brief review of QCA basic. The 3x3 tile Co-planner crossover wire is characterized in section 3. The Section 4 is to describe Reversible Logic gate. In the section 5 we describe proposed URLG gate, section 6 explain the simulation result. In the Section 7 we describe the Proposed URLG using 3×3 TILE and section 8 describe standard function implementation of proposed URLG. In section 9 we analyze temperature region and finally this paper is concluded in section 10.

2 QCA Basic

Quantum dot cellular automata consist of four quantum dots positioned at four corners of cell and two mobile electrons confined within the cell. In QCA logic state is determined by the polarization of electrons rather than voltage level as in CMOS technology. The two stable polarization of electrons P= +1.00 and P= -1.00 of a QCA cell represents logic '1' and logic '0' respectively, shown in fig.1.a.



(b)



(e)

Fig.1: QCA cell and basic gates (a) polarized cell (b) Two types of QCA wires (c) majority voter (d) AOI gate (e) four clocking phases.

2.1 QCA gates

The basic gates of Quantum dot cellular automata are described in previous proposal [1]. QCA wires can be either made up of 900 cells or 450 cells. 450 cells are used for coplanar wire crossings (Fig. 1.b). In case of Inverter, if place two cells at 450 with respect to each other they interact inversely. An array of QCA cells acts as a wire and is able to transmit information from one end to another, i.e., all the cells in the wire will switch their polarizations to follow that of the input or driver cell.

2.2 Majority Voter

Majority Voter (MV) is described as logic function MV (A, B, C) = AB + BC + CA. MV can be realized by 5 QCA cells, as shown in Fig.1.c. Logic AND

and OR functions can be implemented from the MV by setting an input permanently to a 0 or 1 value.

2.3. QCA Clocking

QCA required four phased clocking signal, shown in fig.1.e. The four phases are relaxed, switch, hold and release [12][13]. In the relax phase there is no inter-dot barrier. In the switch phase, barrier is slowly become high and cell attends definite polarity depending on the input. The polarity retains in the hold phase. The barrier is slowly getting lowered and cell release the polarity in the release phase.

2.4. AOI gate

In our previous report [24], we explored Novel And-Or-Inverter (AOI) a promising and effective in area [25],[26]. The nanostructure is shown in Fig.1.d. AOI gate has two control input (constant input) D and E, when D='+1.00' & E= '-1.00'or vice versa. Output F act like AND gate if C='-1.00'and if C='+1.00' output implies OR of two input A, B.
The Boolean function of AOI represents as AOI(A,B,C,D,E)=(D'E')+(D'+E')(AB+BC+CA)

3 3×3 Tile Co-Planer Crossover Wire

In this paper, we explore the effective 3×3 tile co-planer crossover wire. Fig.2.a represents the 3×3 tile structure, referred to as the QCA tile [27], realizing the majority function F=MV (A, B, C). The 9 cells of the 3×3 grid form the base of this tile structure. The choice of terminals A, B, C & F (cells) sets the functional behaviour of the tile. For example, when A, B & C are denoted as the inputs, then at the output F we receive a majority function. To set the tile as fan out branches, the terminal B is to be denoted as input and the rest others as output, as shown in fig.2.b. Similarly, the tile can function as a wire when only terminal B is set to input and the F as output.
It is operate as binary crossover QCA wire i.e. Top input data flows to right output and left input data flows to down output without interference each other.
The tile (3×3 grid) based approach has been proposed in [28] to optimize the area of a design as well as to implement versatile logic and interconnection functions.

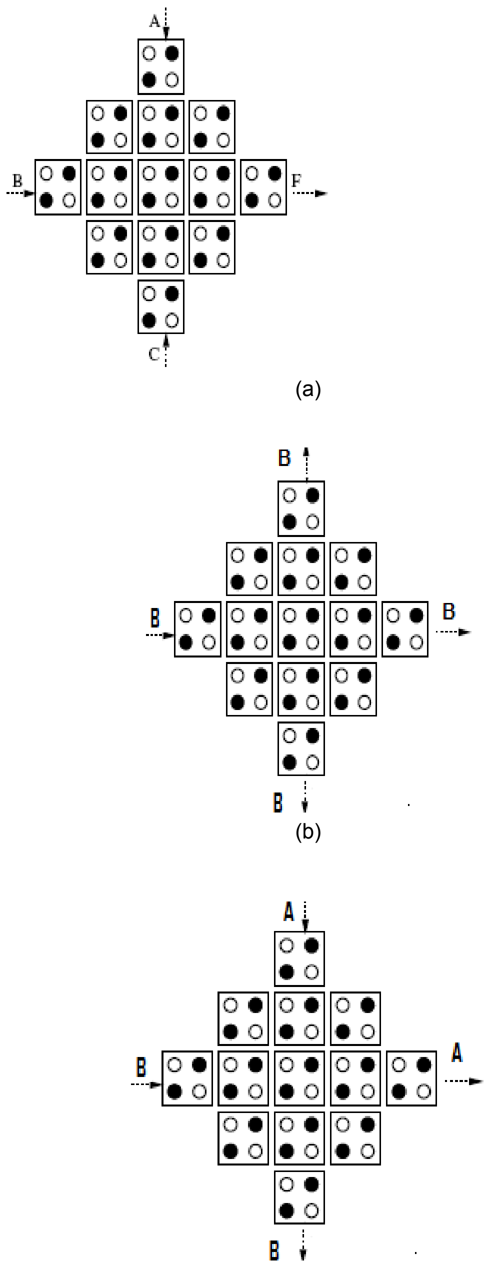


Fig.2: (a) Orthogonal tile with 3×3 grid, (b) 3×3 grid fan out connection, (c) crossover wire.

3.1. Top-down & left-right 3×3 tile co-planer crossover wire

In this paper we had proposed first time top-down & left-right 3×3 tile co-planer crossover wire. Our proposal top-down & left-right 3×3 tile co-planer crossover wire gate is described in Fig.2.A: (a) and its output result is shown in Fig.2.A: (b).It is operate as binary crossover QCA wire i.e. Top input data flows to down output and left input data flows to right output without interference each other.
No. of QCA cells required for this design is 20 and the area of top-down & left-right 3×3 tile co-planer crossover wire is 0.04µm2.We apply this concept in

our reversible logic design.

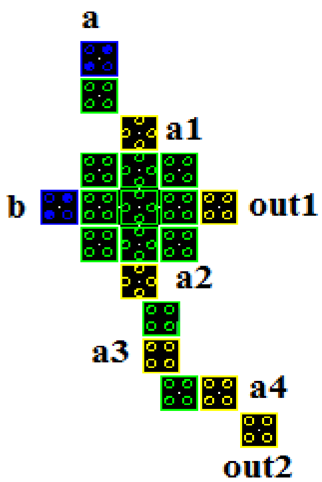


Fig.2.A: (a)QCA Layout for Top-down & left-right 3×3 tile co-planer crossover wire

Fig.2.A: (b) Simulation result of Fig.2.A

4 Reversible Logic Gate

To avoid energy dissipation in irreversible logic gate [17], RLG proved to be promising area of study [18]. There are several proposals had been made which are (A) Feynman (FG) [19] (B) Toffoli gate (TG) [20] (C) Fredkin gate (FRG) [21] (D) NFT Gate [22] commonly performed as reversible logic gates shown in Fig.3 to Fig.6.

Definition 1: If a reversible gate has k input and therefore k outputs, Input Vector Iv is mapped with output vector Ov such that mapping is bijective i.e. the one-to-one mapping between Iv and Ov. The corresponding reversible gate is known as RLG k*k gate.

Definition 2: Garbage output refers to the no. of output added to make n*k function reversible, which output is/are not used for further computations.

Definition 3: Constant Input preset value of input that were added to n*k function to make reversible.

Example1: A 2-input 2-output function given by formula (X, Y)→(X $\oplus$ Y,Y') or truth vector [1,2,3,0] is reversible.

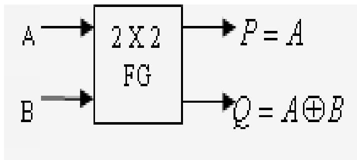


Fig.3: Feynman Gate

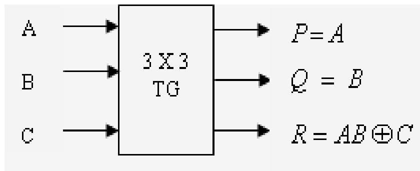


Fig.4: Toffoli Gate

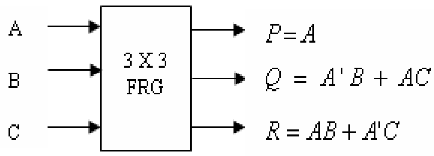


Fig.5: Fredkin Gate

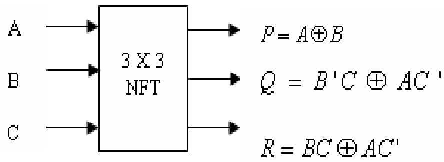


Fig.6: NFT gate

5 Proposed URLG

In the Low power design era Reversible Logic design is proved to be ideal design strategy for logic synthesis, as it is reported in [18] that reversible logic gate will not dissipate the KTln2 heat energy. Here, we explore a Reversible Logic gate which is shown to be a Universal gate for molecular QCA. Here we made attention to implement all classical reversible logic gate namely AND, OR, NOR, NOT and XOR using our proposed URLG. We also focus on the important metric 'garbage count' as reported in [20].

5.1. Characterization of URLG

In this subsection we describe the 3 × 3 URLG. Reversible Logic gate is defined as input vector and output vector must be with one to one correspondences i.e. bijective. The mapping function is given by f: M→M. The Input vector Iv(A,B,C) and corresponding Output vector is given by Ov=(P=AB+BC+CA,Q=A'B+BC'+C'A',R=AB'+B 'C'+C'A) and the truth vector is [3,0,2,6,1,5,7,4]. The symbol diagram is shown in fig 7. The URLG is implemented with 3 MV's as shown in fig 11, the QCA layout; simulated with the simulator QCA Designer v2.0.3 [29]. In our design we required only two No's of Clocking zones. The Clocking zones are namely D0, D1. It is found to be best ever report in terms of clocking zones. In our proposal, we found total no of QCA cells are 135 being used to implement the URLG and the area is given by simulator is 0.16µm2. The cell size and different parameter are being used, are discussed in the section 6.

5.2. Implementing Basic Logic

In this subsection we analyze the URLG as universal gate. It is found to represent as building block of all basic gates. Fig 8 shows that one URLG implement AND and NOR gate while the constant input A=-1.00 i.e. binary '0'. The output P act as AND gate and output R act as NOR gate. Q output is found to be one garbage output. Similarly the fig 9 shows implementation of OR and NAND gate while the constant input A=+1.00 i.e. binary '1'. The fig 10 represents the XOR Gate implementation using three URLG. There are three constant input as described in fig 11 and six garbage output for implementing XOR gate. The comparison made with recently made proposal by Haghparast and Navi [19] is shown Table 1. The result is concluded in terms of garbage minimization and no's of URLG used to better than the existing counterpart.

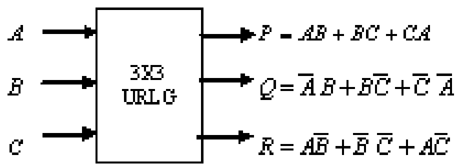


Fig.7: Symbol diagram of 3 × 3 URLG

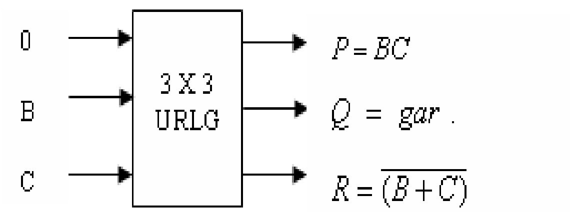


Fig. 8: AND-NOR gate implementation using URLG

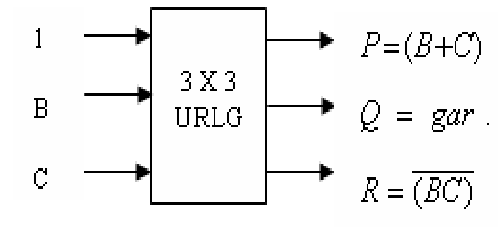


Fig.9: OR-NAND gate implementation using URLG

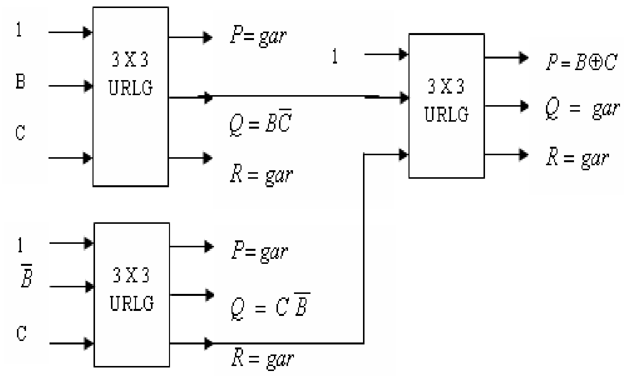


Fig.10: XOR gate implementation using URLG

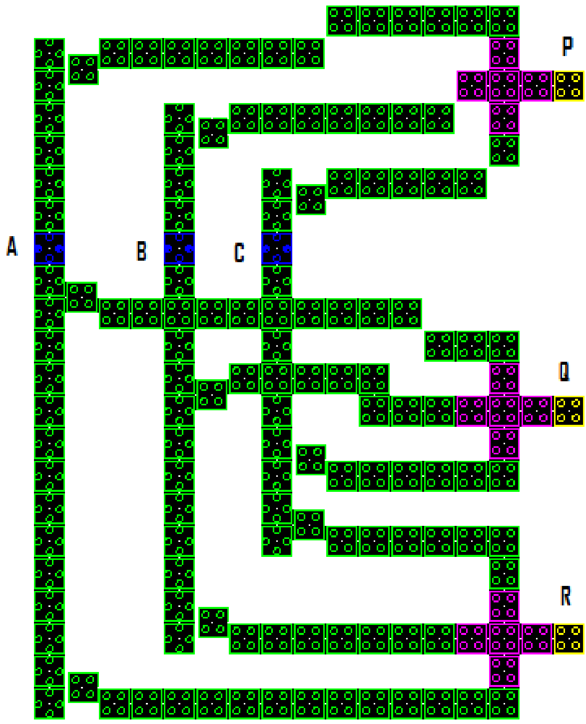


Fig.11: QCA layout of conventional URLG simulated with QCA Designer

6 Simulation Result

The design of URLG was verified with simulator QCA Designer V2.0.3 [29]. In the bi stable approximation we used following parameter: QCA cell size 18nm × 18 nm, Dot size= 5nm2, number of sample =42800, convergence tolerance=0.001000, radius of effect=41nm, relative permittivity =12.9, clock high=9.8e-22 and clock low=3.8e-23, layer separation =11.5000nm. In our QCA Layout we have the goal of remarkable design of URLG. Simulation result verified our proposed URLG. The simulated output is shown in fig 12.

7 Proposed URLG Using 3×3 Tiles

In this section, we are implementing our proposed URLG with 3×3 tile using QCA designer. For 3×3 tile designing we can avoid coplanar wire crossing. The Input vector are A,B,C and corresponding Output vector are P=AB+BC+CA, Q=A'B+BC'+C'A', R=AB'+B'C'+C'A and the truth vector is [3,0,2,6,1,5,7,4]. The QCA layout is shown in fig.13 and simulated output is shown in fig.14. Here we found total no. of QCA cells are 143 being used and the area of proposed URLG is 0.19µm2.

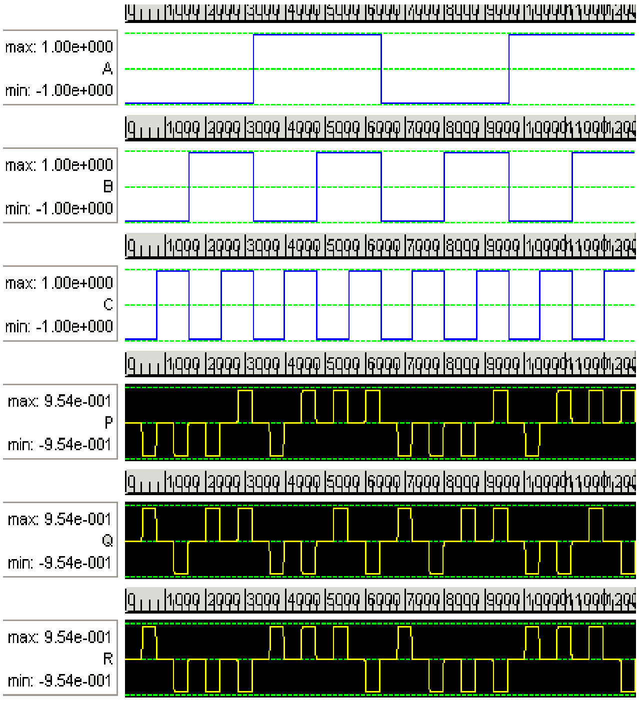


Fig. 12: Simulated output for given Conventional URLG

Table 1 Comparison with our Proposed URLG and NFT

Basic Logic Gate Design	Using NFT [8]			Using URLG			Improvement
	No's of NFT	No's of MV	No's of Garbage output	No's of URLG	No's of MV	No's of Garbage output	
AND +1 NOR	2	26	4	1	3	1	Over all 0 % improvement in using No's of RLG and 20% improvements in Garbage count.
NAND+OR	3	39	6	1	3	1	
EXOR	1	13	2	3	9	6	
NOT	-!	-	-	1	3	2	
Total	6	68	12	6	60	10	

[1]both gates are implemented with single gate. Hence no extra URLG required. ! NOT gate implemented with NAND gate

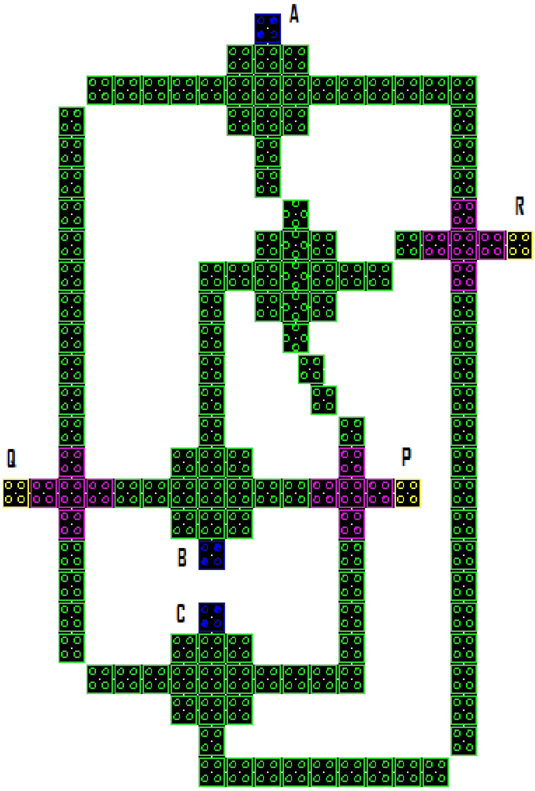


Fig.13: 3×3 grid QCA layout of conventional URLG simulated with QCA Designer

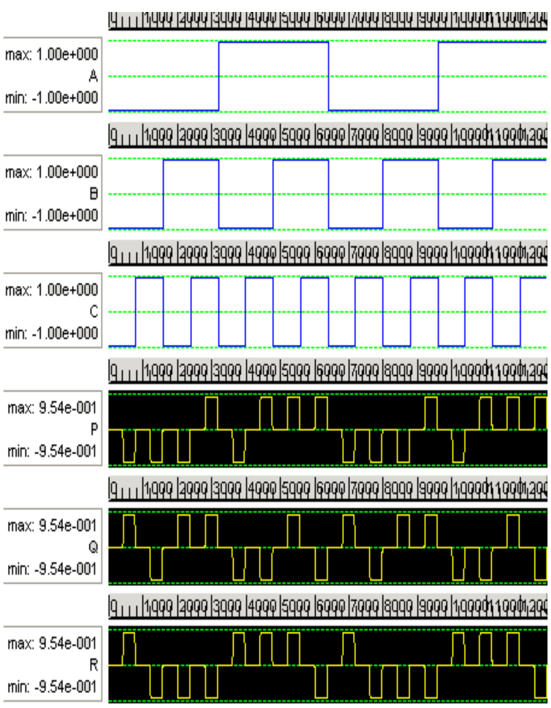


Fig. 14: Simulated output for proposed URLG

8 Implementation

Thirteen standard functions are shown in Table 2. No. of URLG required for every standard functions are shown in Table 2 and Fig.15 to Fig 27.

Table 2 Number of URLG used in standard function

	Functions	No. of URLG
1	F = AB'C	3
2	F = AB	1
3	F = A'BC+A'B'C'	4
4	F = A'BC+AB'C'	5
5	F = A'B+BC'	3
6	F = AB'+A'BC	4
7	F = A'BC+ABC'+A'B'C'	6
8	F = A	1
9	F = AB+AC+BC	1
10	F = A'B+B'C	4
11	F = A'B+BC+AB'C'	5
12	F = AB+A'B'	5
13	F = ABC'+A'B'C'+AB'C+A'BC	7

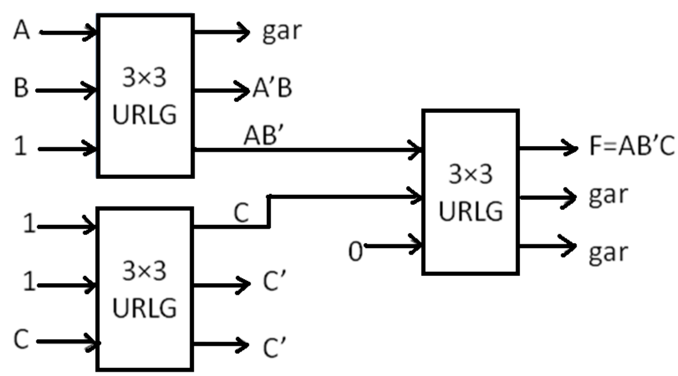


Fig.15. Logic Implementation using URLG for F = AB'C.

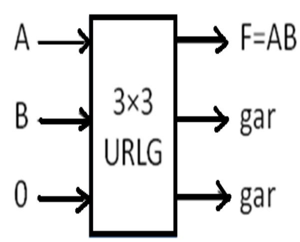


Fig.16. Logic Implementation using URLG for F=AB.

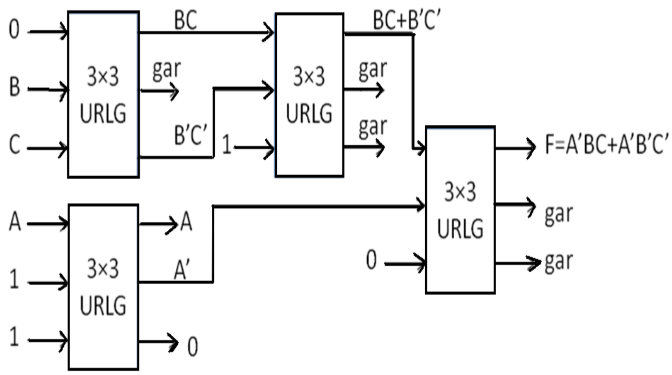


Fig.17. Logic Implementation using URLG for F = A'BC+A'B'C'

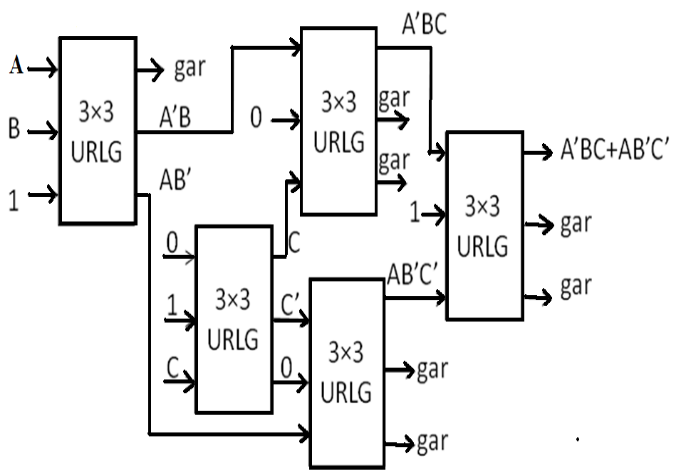


FIg.18. Logic Implementation using URLG for F = A'BC+AB'C'

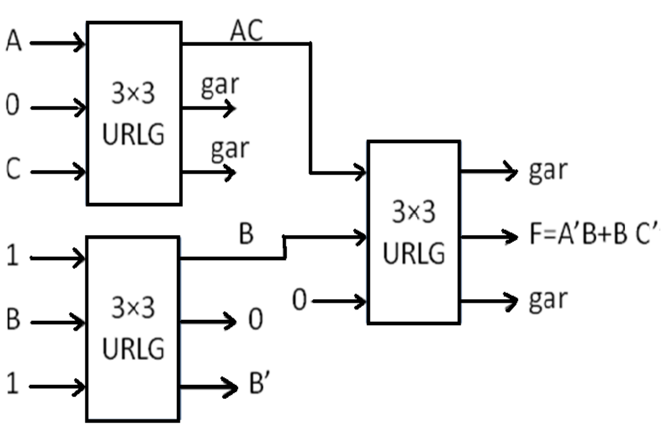


Fig.19. Logic Implementation using URLG for F = A'B+BC'

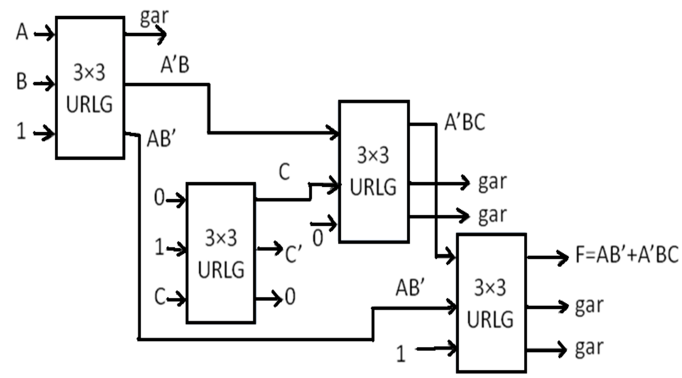


Fig.20. Logic Implementation using URLG for F = AB'+A'BC

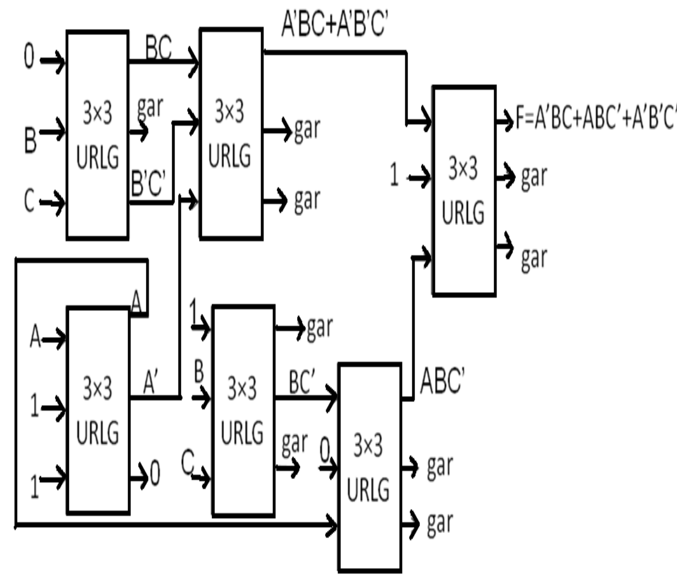


Fig.21. Logic Implementation using URLG for F = A'BC+ABC'+A'B'C'

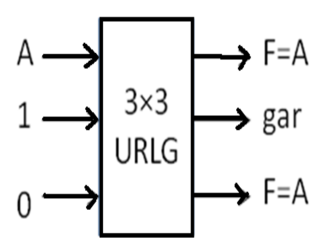


Fig.22. Logic Implementation using URLG for F = A

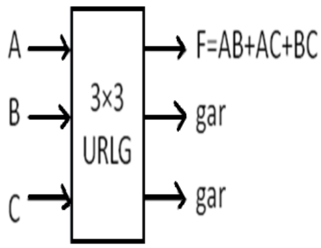


Fig.23. Logic Implementation using URLG for F = AB+AC+BC

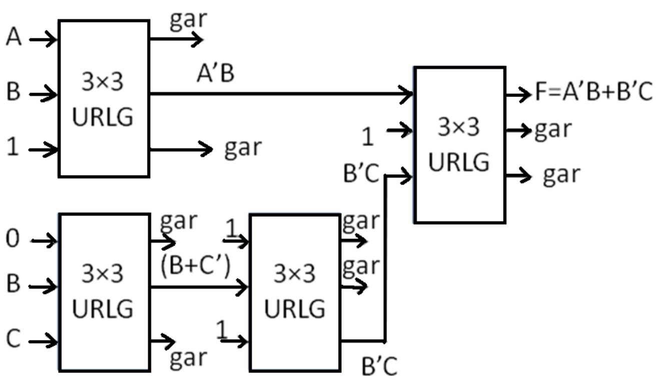


Fig.24. Logic Implementation using URLG for F = A'B+B'C

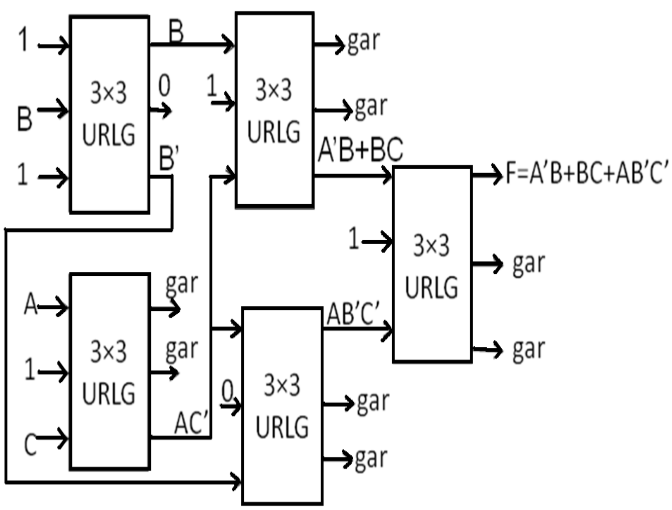


Fig.25. Logic Implementation using URLG for F = A'B+BC+AB'C'

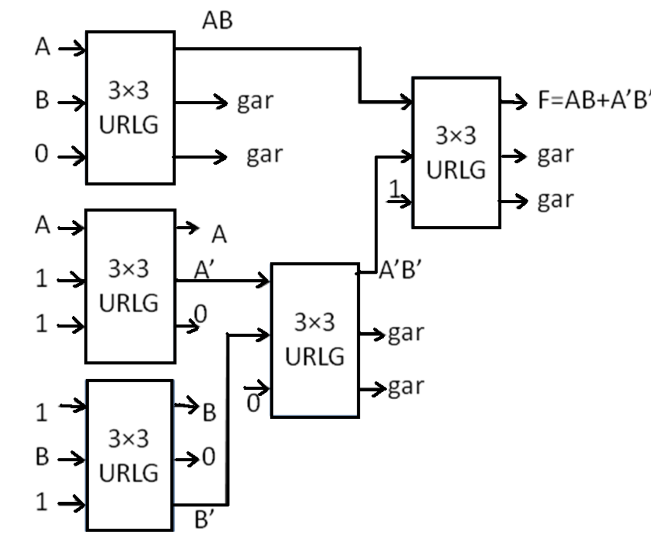


Fig.26. Logic Implementation using URLG for F = AB+A'B'

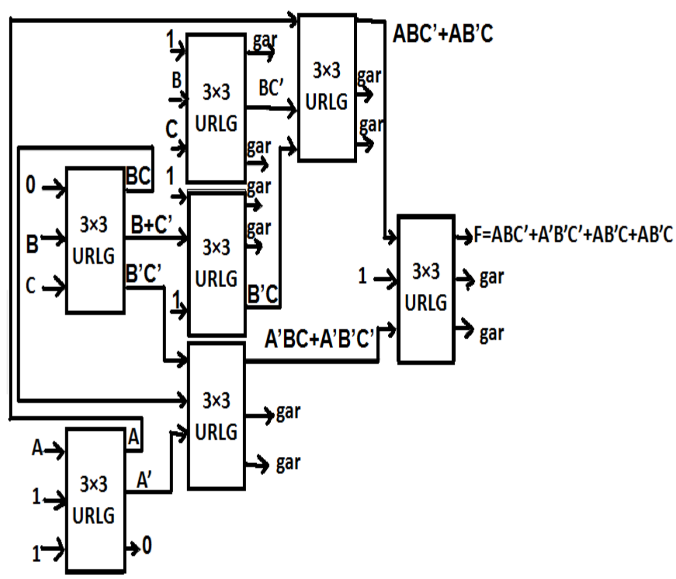


Fig.27. Logic Implementation using URLG for F = ABC'+A'B'C'+AB'C +A'BC.

9 Temperature Dependencies

The maximum achievable operating temperature is one of the main challenges in QCA technology. QCA works properly under very low temperature region, up to 1K. Fig. 28 shows the graph of output polarization Q in different temperature, where blue line is used for maximum value and red line is used for minimum value. After 50K, output becomes zero.

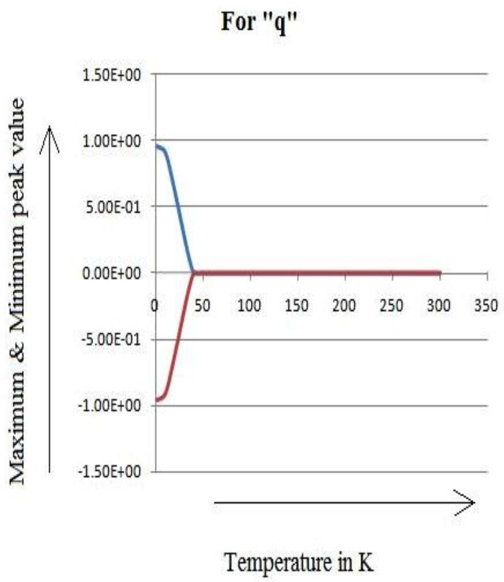


Fig.28. Temperature dependency curve of QCA cell Polarization

Table 3 Output Q in different temperature

Temperature in K	Out put Polarization Q	
	Maximum value of polarization	Minimum value of polarization
1	9.54e-001	-9.54e-001
10	9.04e-001	-9.04e-001
20	6.06e-001	-6.06e-001
30	2.64e-001	-2.64e-001
35	1.07e-001	-1.07e-001
40	3.05e-004	-3.05e-004
50	6.85e-008	-6.85e-008
100	3.41e-010	-3.41e-010
150	4.62e-012	-4.62e-012
200	1.50e-012	-1.49e-012
250	8.23e-013	-8.23e-013
300	3.89e-013	-3.89e-013

10 Conclusions

In this paper, we explore a Universal Reversible logic gate (URLG) which is proved to be Universal gate. We also compared with very recently proposed NFT and achieved 20% improvements in garbage minimization with respect to NFT as shown in Table 1. During design of URLG we made attention to reduce the No's of cells as well as to reduce the area which is found to be 0.16μm2 and compare to Fredkin gate it achieved 56% reduction in area (area of Fredkin gate is 0.37 μm2). The area which is found to be 0.19μm2 in 3×3 tiles and compare to Fredkin gate it achieved 48% reduction in area (area of Fredkin gate is 0.37 μm2).URLG is also must be applicable for design sequential circuit and is a basic building block for 4×4 RLG. Hence we conclude that our proposed URLG design must be promising step towards the goal of low power design in nanotechnology.

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, 277, 1997,pp.928-930..

[2] C S Lent , P D Tougaw , W Porod and G H Bernstein, Quantum cellular automata, Nanotechnology, 1,4, 49-57, **(1993).**

[3] C. S. Lent and P. D. Tougaw, A device architecture for computing with quantum dots, Proc. IEEE, 85, 4, 1997, pp. 541-557.

[4] W. Porod, Quantum-dot devices and quantum-dot cellular automata, Inter. J. Bifurcation and Chaos, 7, 10, 1997,pp. 2199-2218.

[5] I. Amlani, A. O. Orlov, G. Toth, C. S. Lent, G. H. Bernstein and G. L. Snider, Digital logic gate using quantum-dot cellular automata, Applied Physics Letters, 284,1999,pp.289-291.

[6] C. S. Lent and P. D. Taugaw, Dynamic behavior of quantum cellular automata J. Appl. Phys. 80, 8, 1996**.**

[7] Minsu Choi, Myungsu Choi, Zachary Patitz and Nohpill Park, Efficient and Robust Delay-Insensitive QCA (Quantum-Dot Cellular Automata) Design, Proceedings of the 21st IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT'06),2006**.**

[8] Kyosun Kim, Kaijie Wu and Ramesh Karri, The Robust QCA Adder Designs Using Composable QCA Building Blocks, IEEE Trans. on Computer-Aided design of Integrated Circuits and Systems, 26, 1,2007.

[9] G. Toth and C. S. Lent, Quasiadiabatic switching for metal island quantum dot cellular automata, J. Appl. Phys., 85(5), 1999, pp. 2977–2984.

[10] I. Amlani, A. O. Orlov, R. K. Kummamuru, G. H. Bernstein, C. S. Lent, and G. L. Snider, Experimental demonstration of a

leadless quantum-dot cellular automata cell, Appl. Phys. Lett., 77, 5, 200,pp.738-740.
[11] K. Hennessy and C. S. Lent, Clocking of molecular quantum dot cellular automata, J. Vac. Sci. Technol. B, 19(5), 2001, pp.1752– 1755.
[12] C. S. Lent and B. Isaksen, Clocked molecular quantum-dot cellular automata, IEEE Trans. on Electron Dev., 50(9),2003, pp. 1890– 1895.
[13] M. T. Niemier, M. J. Kontz and P. M. Kogge, A design of and design tools for a novel quantum dot based microprocessor, In 27th Ann. Design Automation Conference, 2000,pp.227–232.
[14] A. Vetteth, K. Walus, V. Dimitrov and G. Jullien, Quantum dot cellular automata carry-look-ahead adder and barrel shifter, IEEE Emerging Telecommunications TechnologiesConference, 2002.
[15] Landauer R., Irreversibility and heat generation in the computing process, IBM J. Research and Development, 5(3), 1961, pp.183-191.
[16] Bennett C. H., Logical reversibility of computation, IBM J. Research and Development, 17,1973,pp. 525-532.
[17] P.D. Tougaw and C.S. Lent, Logical devices implemented using quantum cellular automata, Journal of Applied Physics, 75, 1818**,** 1999.
[18] G. Snider, A. Orlov, C. Lent, G. Bernstein, M. Lieberman and T. Fehlner, Implementation of Quantum-dot Cellular Automata, ICONN,2006,pp. 544-547.
[19] Feynman R., Quantum mechanical computers. Optics News, 11, 1985, pp. 11-20.
[20] Toffoli T., Reversible computing, Tech Memo MIT/LCS/TM-151, MIT Lab for Computer Science, 1980.
[21] Fredkin E. and T. Toffoli, Conservative logic, Int'l J. Theoretical Physics, 21, 1982, pp. 219-253.
[22] M.Haghparast and K.Navi, A Novel Fault Tolerance Reversible Gate for Nanotechnology Based System, Am.J. Applied Sci., 5(5), 2008, pp.519-523.
[23] D. Maslov and G. W. Dueck, Reversible Cascades with Minimal Garbage, IEEE Trans. on CAD, 23(11),2004,pp.1497-1509.
[24] K.Das and D.De, A Novel approach of And-Or-Inverter (AOI) gate design for QCA, Proc. IEEE conf. CODEC-09, ISBN 978-81-8465-152-2, 2009, pp.1-4.
[25] M. Momenzadeh, Jing Huang, M. B. Tahoori and F. Lombardi, Characterization, test, and logic synthesis of and-or-inverter (AOI) gate design for QCA implementation, Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions,24, 12,2003,pp.1881–1893.
[26] B. Sen and B. K. Sikdar, Characterization of universal Nand-Nor-Inverter QCA gate", submitted to 11th IEEE VLSI Design and Test Symposium, Kolkata, 2007,pp.8-11.
[27] J. Huang, M. Momenzadeh and F. Lombardi, Defect tolerance of QCA tiles, in the proceedings of Design Automation and Test in Europe (DATE), 1, 2006,pp.1-6.
[28] J. Huang, M. Momenzadeh, L. Schiano and F. Lombardi, Simulation-based design of modular QCA circuits, in Proceedings of IEEE conference on nanotechnology, Nagoya, 2005.
[29] K.Walus, 'ATIPS laboratory QCA Designer' ATIPS laboratory, University of Calgary, Canada, 2002.homepage-http://www.atips.ca/projects/qcadesigner.