JOURNAL OF NANO- AND ELECTRONIC PHYSICS
Vol. **8** No 2, 02042(8pp) (2016)

ЖУРНАЛ НАНО- ТА ЕЛЕКТРОННОЇ ФІЗИКИ
Том **8** № 2, 02042(8cc) (2016)

Efficient Design of Reversible Code Converters Using Quantum Dot Cellular Automata

Javeed Iqbal Reshi*, M. Tariq Banday†

Department of Electronics & Inst. Technology, University of Kashmir, Srinagar, India

(Received 25 January 2016; revised manuscript received 14 June 2016; published online 21 June 2016)

Quantum dot Cellular Automata (QCA) is an attractive field of nano-technology which offers the various advantages over existing CMOS technology for the development of logic circuits. Contradictory to other technologies which use the voltage levels for logic representation, QCA utilizes the polarization of electrons for representing the binary states in the QCA Cell. Conventional logic circuits are not energy efficient as they are not reversible in nature and hence lead to energy dissipation. Thus there is a need of a serious effort that will provide an efficient paradigm for designing the circuits which does not dissipation the energy and hence will preserve the information. This paper offers the efficient design of various QCA reversible code converters which prove to be efficient in term of cell Area, cell count, total area, latency and complexity. All the proposed reversible code converter designs were simulated and their credibility was successfully verified with the QCADesigner tool

Keywords: Quantum dot Cellular Automata, CMOS, Reversible code converters, Latency, QCADesigner.

DOI: 10.21272/jnep.8(2).02042 PACS number: 85.35.Be

1. INTRODUCTION

The main motive of the semiconductor technology is to provide the large scale integration, high performance, high performance, micro-scale computation and energy efficient circuits. Over several decades the conventional technologies especially the Complementary Metal Oxide Semiconductor technology (CMOS) succeeded to provide these merits. But reducing the dimensions beyond the optimum limit several technological problems arise, like high leakage current, high lithography cost, and limitation of speed in GHz range. To resolve these problems researchers need to find out an alternative of CMOS technology. After serious efforts several efficient alternate technologies were put forward including Quantum dot Cellular Automata (QCA) were proposed. Among them QCA gained the attention because of unconventional way of storing the information and hence was opted as an alternative to conventional CMOS technology. Unconventionally in QCA technology the binary information is stored as the position of electrons in a QCA cell. This novel idea was first put forwarded by Dr. Craig Lent [1, 2].

In 1961 Landauer [3] proved that there is a direct relation between the information loss and power dissipation. He states that in traditional logic circuits a single bit of information loss leads to the power dissipation of the order of KT ln (2) joules of energy (where T is absolute temperature and K is the Boltzmann's constant). Bennett in 1973 showed that for an efficient digital design if the logic circuits are designed with reversible logic no power will be dissipated. Thus Bennett's theorem stressed technological necessity that future technology will render on the reversible gates so as to reduce power loss [4]. Thus reversible circuits are the circuits that do not lose information during the logic computations. A circuit can be called as reversible if it satisfies the following conditions:

1) The number of inputs must be equal to number of outputs.
2) There must be one to one mapping between the input and output vectors. i.e. for a particular input combination there must be a unique output. A single output will not exist for different input combinations.

The conventional logic circuits are designed with the traditional logic gates like AND, OR etc. and hence they are not energy efficient and as such there is a need to replace the conventional logic gates with the reversible gates. The simplest type of reversible gate is an inverter as it fulfills all the conditions required for a reversible gate. Several reversible gates are reported in the open literature among which the most commonly used reversible logic gates are the Toffoli [5], Feynman [6] and the Fredkin gate [7]. In addition to these gates, other reversible gates have also been proposed [8-15]. Reversible computing is emerging computing paradigm which finds the applications in low power nano computing optical computing, DNA computing and quantum computers.

2. QCA BASICS

The primary computational block of the QCA technology is the QCA cell. The cell is considered to have a square shape with dimensions in nanometer range. The cell contains four sites at the corners where the charge can reside known as quantum dots. The cell can store either a binary 1 or binary 0 depending on the orientation of electrons within a cell. The mobile electrons are pushed in the cell which can tunnel between the four quantum dots within the cell. The tunneling is confined within a cell and is not permitted between the cells. The two electrons within the cell align at the antipodal sites due to the coulombic repulsion. As a result the two polarizations are possible. The two polarizations are defined as binary "1" and "0", as shown in Fig. 1

* javeedreshi@gmail.com
† sgrmtb@yahoo.com

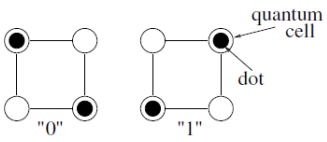


Fig. 1 – QCA cell polarizations and representations of binary 0 and binary

It is possible to implement all combinational and sequential logic functions by proper arrangement of cells so that the polarization of one cell sets the polarization of a nearby cell .According to previous studies, several logic gates and computing devices [16] are implemented with QCA.

2.1 Majority Gate

The majority logic gate is shown in Fig. 2a. It consists of three inputs labeled as A, B, C and single output. The centre cell is referred as driver cell and it switches to major polarization and maintains a constant output. The logic equation of the majority gate can be realized as:

$$M(A,B,C) = AB + BC + AC$$

The majority gate can be programmed to either function as an AND or an OR gate by fixing the polarization of any one of the input cells as $P = -1$ or $P = +1$. As shown in Fig. 2.

$$M(A,B,-1) = A.B$$

$$M(A,B,1) = A + B$$

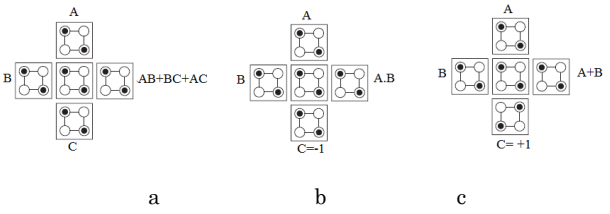


Fig. 2 – (a) Majority Gate. (b) Majority AND gate. (c) Majority OR gate

2.2 Inverter

The QCA cells can be used to form the primitive logic gates. The simplest structure is the inverter shown Fig. 3

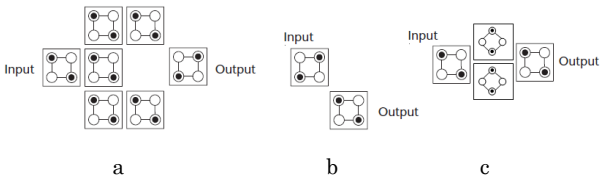


Fig. 3a – (a) Seven cell inverter (b) two cell inverter (c) four cell inverter

There three main types of inverters used in the QCA technology. Fig. 3a shows the seven cell inverter which operates properly in almost all the circuits because of strong polarization. The input polarization split into two polarizations and at the end the two wires join and make the reverse polarization. The second type of inverter is a two cell inverter which is designed with the two cells which are vertically displaced as shown in Fig 3b. The other type of inverter is a four cell inverter shown in Fig. 3c. The two vertical cells are rotated while as the two horizontal cells in which one represents the input and the other represents the output are normal cells.

2.3 QCA Wire

There are two types of wires in QCA, 90° QCA wire and 45° wire as shown in Fig. 4.

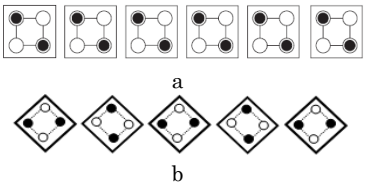


Fig. 4 – (a) 90° QCA wire (b) 45° QCA wire

The 90° QCA wire is simply an array of cells arranged in a cascaded fashion. The polarization of each cell is affected by the electrostatic force due to coulombic repulsion generated by the neighboring cells. Thus information propagates down from one cell to another through the QCA wire [4]. The 45° wire comprise of rotated cells and the signal alternates between the input value and its logic complement as it traverses the chain towards the output.

2.4 QCA Wire Crossing

One of the unique property of the QCA is the capability to create the different signal wire crossings. There are two types of wire crossings available in QCA: Coplanar crossing and multi layer crossing. A coplanar crossing is implemented by one layer only. A coplanar crossing uses both regular and rotated cells as shown in Fig. 5a. These cells do not interact with each other if properly aligned. On the other hand the multi layer crossing is implemented using more than one layer. Multi layer crossings are expected to achieve more reliable results, but they are difficult to fabricate. The multi layer crossing is shown in Fig. 5b.

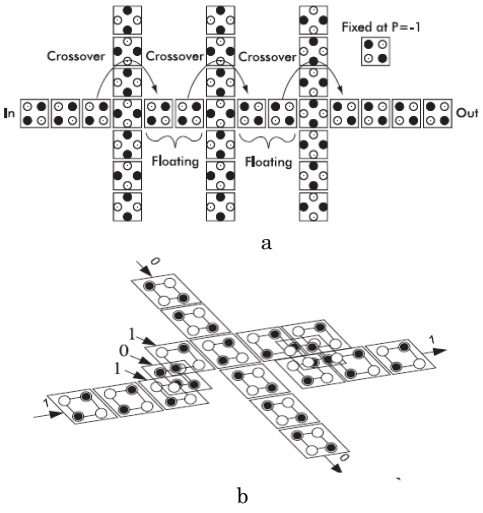


Fig. 5 – (a) Coplanar wire crossing (b) Multi layer wire crossing

2.5 QCA Clock

QCA uses the Quasi-adiabatic four phase clocking which ensures the system is always in its instantaneous ground state. QCA cells are timed in four successive clocking zones. Each signal is 90° phase shifted and each clock consists of four phases switch, hold, release and relax. These clock phases are used to change the state of a cell either by raising or lowering the tunneling barrier. During the switch state the inter-dot barrier is changed from low to high and hence the cell attains a fixed polarization. During the Hold state the inter-dot barrier is high and the cell holds the state of polarization. During the Release state the inter-dot barrier is lowered and is allowed to relax in to unpolarized state. In the Relax state the inter-dot barrier is kept low and the cell remains in NULL state. The four phase clocking is shown in Fig. 6.

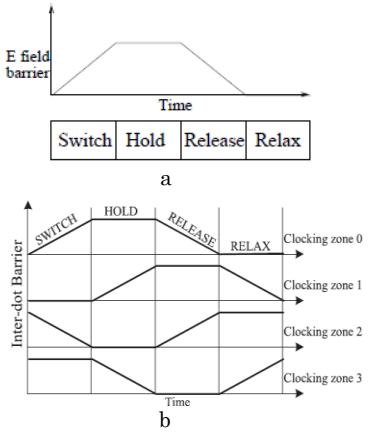


Fig. 6 – (a) 4 phase clocking. (b) Switching in a binary wire

3. FEYNMAN GATE

Feynman gate which is also known as controlled NOT gate (CNOT) is a 2×2 reversible gate. The input vector is $I(A,B)$ is mapped in to output vector $O(P,Q)$. The relationship between the inputs and the outputs is given as:

$$P = A \tag{1}$$

$$Q = A \oplus B \tag{2}$$

The majority logic format for the equations can be given as:

$$P = A \tag{3}$$

$$Q = (M(M(\overline{A}, B, -1), M(A, \overline{B}, -1), 1) \tag{4}$$

Table 1 gives the relationship between the input and output of Feynman gate.

Table 1 – Truth table of Feynman gate

Inputs		Outputs	
A	**B**	**P**	**Q**
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

From the truth table it is clear that when the $A = 0$ the output $Q = B$, but when $A = 1$, the output Q is compliment of B. The Schematics and the Majority Voter representation clearly explain the functioning of Feynman gate and is shown in Fig. 7.

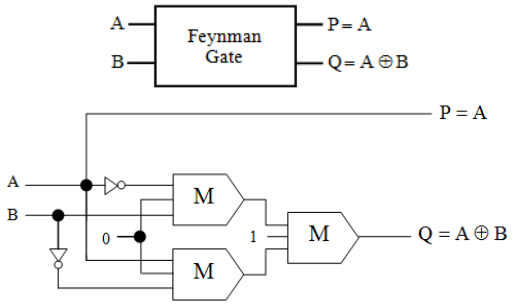


Fig. 7 – Schematics and the MV representation of Feynman gate

The Feynman gate is designed with the help of three majority gates in which the two performs the AND operation and one performs the OR operation. The QCA circuit implementation and the simulation result of Feynman gate is shown in Fig. 8.

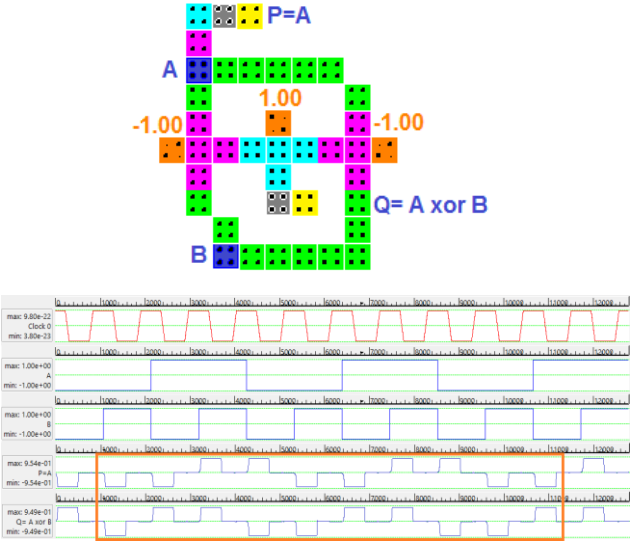


Fig. 8 – QCA implementation & simulation result of Feynman gate

4. 3-BIT REVERSIBLE BINARY TO GRAY CODE CONVERTER

A code is basically a symbolic representation of discrete information and in digital electronics there exists various types of code like binary code, binary coded decimal code (BCD), Gray code, etc. A Code converter is a combinational circuit which converts one type of code into other. One of the most commonly code is a gray code which is a non weighted code. Most of code especially in binary code there is a rapid switching of bits from one transition to another (e.g. 011 to 100) but in a gray code there is a single bit change from LSB to MSB while traversing between the logical sequences. This feature is useful to reduce the rapid switching activity. Consider a 3 bit reversible code converter in which binary input vector is $I(A,B,C)$ and the gray output vec-

tor be $O(P,Q,R)$. The truth table of 3 bit reversible binary to gray code is depicted in Table 2.

Table 2 – Truth table of 3 bit reversible binary to gray code converter

Input (Binary Code)			Output (Gray Code)		
A	**B**	**C**	**P**	**Q**	**R**
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

From the Table 2 it is clear that the outputs follow the following equations

$$P = A \tag{5}$$

$$Q = A \oplus B \tag{6}$$

$$R = B \oplus C \tag{7}$$

From the above equations it is clear that we only require an EX-OR operation for implementing the 3 bit converter. The Ex-OR operation can be achieved with the Feynman gate, hence the equation 4 & 5 can be implemented by a single Feynman gate. Thus the required 3 bit converter (binary to gray) can be implemented by employing two Feynman gates only as it produces least number of garbage outputs. In QCA every expression needs to decompose in to majority gate and then it can be easily implemented. The majority logic format of the equations 4, 5 & 6 can be given as

$$P = A \tag{8}$$

$$Q = (M(M(\overline{A},B,-1),M(A,\overline{B},-1),1) \tag{9}$$

$$R = (M(M(\overline{B},C,-1),M(B,\overline{C},-1),1) \tag{10}$$

The Block diagram and the majority logic representation of 3 bit reversible binary to gray code converter are shown in Fig. 9.

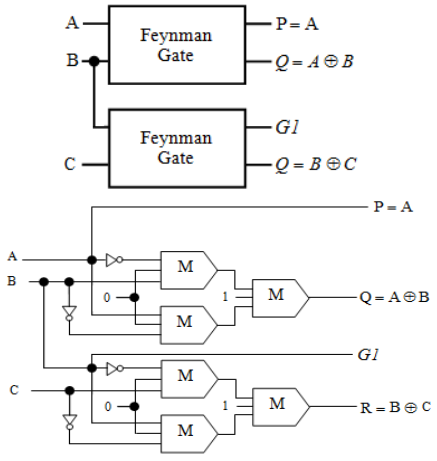


Fig. 9 – Schematics & MV representation of 3 bit reversible binary to gray code converter

The QCA implementation and simulation result of 3 bit reversible binary to gray code converter is shown in Fig. 10 respectively.

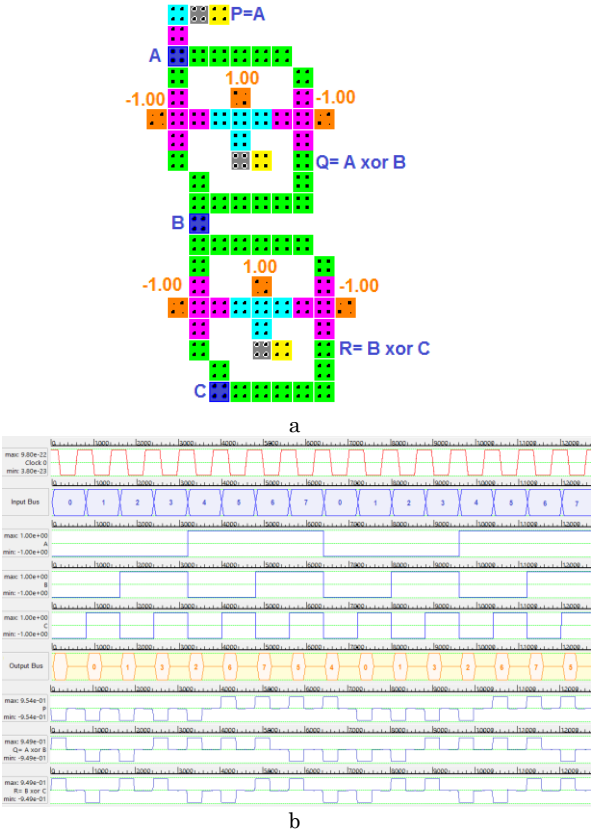


Fig. 10 – (a) QCA implementation & (b) simulation result of 3 bit reversible binary to gray code converter

4.1 4-bit Reversible Binary to Gray code converter

Using the Feynman gate as the basic building block the 4 bit reversible binary to gray code can be designed. The input vector $I(A,B,C,D)$ can be mapped into the output vector $O(P,Q,R,S)$. The input output relationship for the 4 bit reversible binary to gray code converter holds the following equations.

$$P = A \tag{11}$$

$$Q = A \oplus B \tag{12}$$

$$R = B \oplus C \tag{13}$$

$$S = C \oplus D \tag{14}$$

As the above equations requires only ex-or gates for implementation of the required circuit. These ex-or gates are designed with the majority gate. The majority logic format of the equations (11-14) is shown as:

$$P = A \tag{15}$$

$$Q = (M(M(\overline{A},B,-1),M(A,\overline{B},-1),1) \tag{16}$$

$$R = (M(M(\overline{B},C,-1),M(B,\overline{C},-1),1) \tag{17}$$

$$S = (M(M(\overline{C}, D, -1), M(C, \overline{D}, -1), 1) \quad (18)$$

The truth table of the 4 bit reversible binary to gray code converter is given in Table 3.

Table 3 – Truth table of 4 bit reversible binary to gray code converter

Input (Binary Code)				Output (Gray Code)			
A	B	C	D	P	Q	R	S
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

Thus the required circuit can be designed with three Feynman gates only. The schematics and the MV representation of the 4 bit converter is shown in fig. 11.

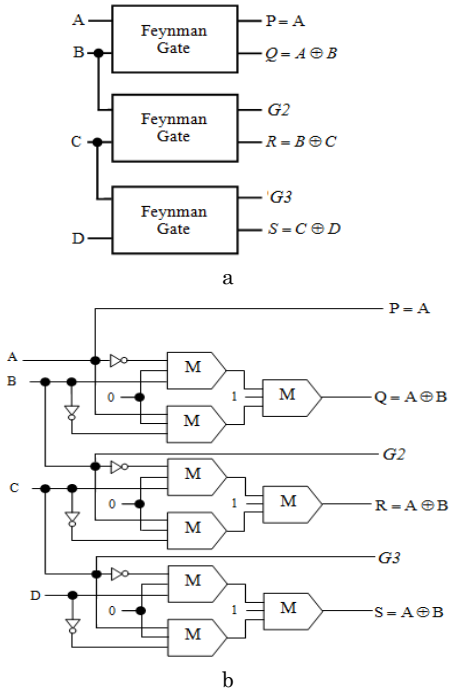


Fig. 11 – (a) Schematics & (b) MV representation of 4 bit binary to gray code converter

The QCA implementation and the simulation result of 4-bit reversible binary to gray code converter are given in Fig. 12.

The QCA implementation of the 4 bit binary to gray code converter shows that the design attains a modular approach thereby reducing the complexity of the circuit. The circuit generates only to garbage outputs which have

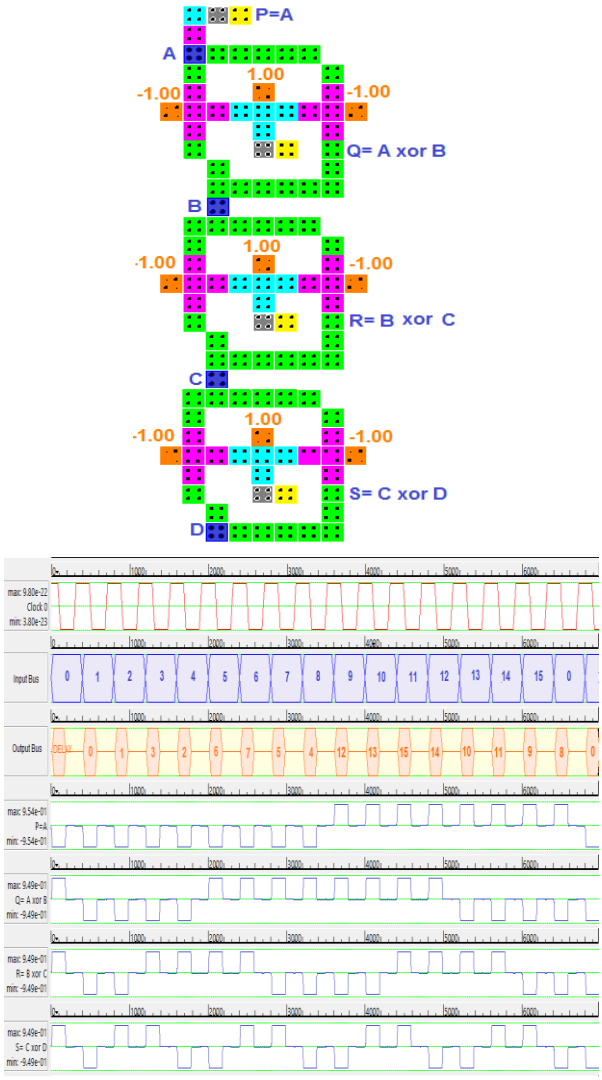


Fig. 12 – QCA implementation & Simulation result of 4 bit binary to gray code converter

omitted from the QCA implementations as they are not required. The performance of the proposed reversible binary to gray code converters and the other circuits given in the open literature are given in the Table 4. From the Table 4 it is evident that the proposed circuits are better in terms of cell count, cell area, total area latency and complexity. Moreover the said circuit has been designed with Feynman gates only which results Quantum cost of only 2.

5. 3 BIT GRAY TO BINARY CODE CONVERTER

Consider a three bit gray to binary converter having the input vector $I(A,B,C)$ and the output vector is $O(P,Q,R)$ .The input output relationship can be represented with the equations given as

$$P = A \quad (19)$$

$$Q = A \oplus B \quad (20)$$

$$R = A \oplus B \oplus C \quad (21)$$

Table 4 – Comparison of various reversible Binary to gray code converters

Circuit	Cell Count (μm²)	Cell Area (μm²)	Total Area (μm²)	Latency (in clock cycles)
Feynman gate[17]	54	0.0174	0.0388	4
Feynman gate[18]	75	0.0243	0.0605	4
Feynman gate[19]	65	0.0207	0.0631	4
Proposed Feynman gate	39	0.0129	0.0291	4
3 bit Reversible Binary to gray code converter [17]	118	0.0382	0.0926	4
3 bit Reversible Binary to gray code converter [20]	117	0.0379	0.0953	10
Proposed 3 bit Reversible Binary to gray code converter	75	0.0243	0.0554	4
Novel 4 bit Reversible Binary to gray code converter	111	0.0359	0.0816	4

The majority logic format of the equations (19-21) is shown as:

$$P = A \tag{22}$$

$$Q = (M(M(\overline{A}, B, -1), M(A, \overline{B}, -1), 1) \tag{23}$$

$$R = (M(M(\overline{Q}, C, -1), M(Q, \overline{C}, -1), 1) \tag{24}$$

mapping of the 3 bit converter is shown in Table 5.

Table 5 – Truth table of 3 bit reversible Gray to Binary code converter

Input (Gray Code)			Output(Binary Code)		
A	**B**	**C**	**X**	**Y**	**Z**
0	0	0	0	0	0
0	0	1	0	0	1
0	1	1	0	1	0
0	1	0	0	1	1
1	1	0	1	0	0
1	1	1	1	0	1
1	0	1	1	1	0
1	0	0	1	1	1

The 3 bit reversible gray to binary code converter is shown in fig. The required 3 bit reversible gray to binary converter can be designed by cascading the two Feynman gates. The 2nd output of first Feynman gate (i.e. Q) is applied to the C input of the 2nd Feynman gate. This arrangement will generate only a single garbage output labeled as G1. The QCA implementation and the simulation result of 3 bit gray to binary converter is shown in Fig. 13.

The QCA implementation and the simulation result of 3-bit reversible gray to binary code converter are given in Fig. 14.

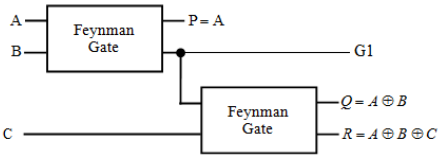
5.1 4-Bit Reversible Gray to Binary Code Converter

For a 4 bit reversible gray to binary code converter consider the input vector $I(A,B,C,D)$ and an output vector $O(P,Q,R,S)$. The input and output vectors are related with the following relations.

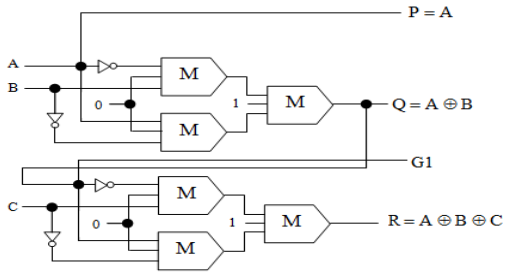
$$P = A \tag{25}$$

$$Q = A \oplus B \tag{26}$$

$$R = A \oplus B \oplus C \tag{27}$$

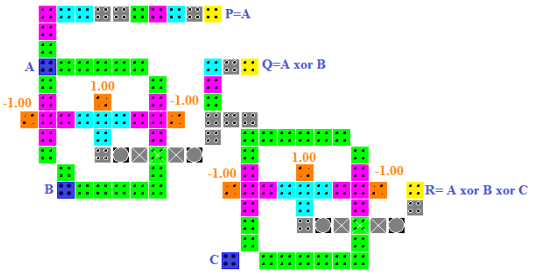


a

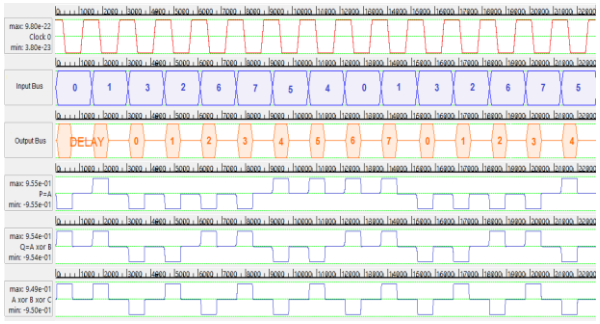


b

Fig. 13 – Schematics & MV representation of 3 bit reversible gray to binary code converter



a



b

Fig. 14 – QCA implementation and simulation result of 3 bit reversible 3-bit reversible Gray to Binary code converter

$$S = A \oplus B \oplus C \oplus D \tag{28}$$

The majority logic format of the equations (25-28) is shown as:

$$P = A \tag{29}$$

$$Q = (M(M(\overline{A}, B, -1), M(A, \overline{B}, -1), 1) \tag{30}$$

$$R = (M(M(\overline{Q}, C, -1), M(Q, \overline{C}, -1), 1) \tag{31}$$

$$S = (M(M(\overline{R}, C, -1), M(R, \overline{C}, -1), 1) \tag{32}$$

The truth table of the 4-bit reversible gray to binary code converter derived from the above relations is given in Table 6.

Table 6 – Truth table of 3 bit reversible Gray to Binary code converter

Input(Gray Code)				Output(Binary Code)			
A	**B**	**C**	**D**	**P**	**Q**	**R**	**S**
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1

The Schematics and MV representation of 4 bit reversible gray to binary code converter is given in Fig. 15.

From the fig. 16 it is clear that the required 4 bit reversible converter can be obtained by cascading the three stages of Feynman gates. The design enjoys an advantage of generating the two garbage outputs only. The QCA implementation and the simulation result of 4 bit reversible gray to binary converter is shown in Fig. 16.

A little work has been done on the design of reversible code converter circuits using the quantum dot cellular automata, only the two references are available in the open literature. The Feynman gate is used as the basic block for designing the reversible binary to gray code converters. From the Table 4 it is clear that the proposed Feynman gate and the other converters using the proposed Feynman gates are efficient in term of cell count, cell area, total area, latency and the complexity. As all the circuits are designed with Feynman gates only the quantum cost of all the code converter circuits are 2. Moreover the 4 bit reversible binary to gray converter is a novel as it is not reported in the literature.

The performance parameters of the various reversible gray to binary code converters with are available in the open literature and the proposed ones are given in the Table 7.

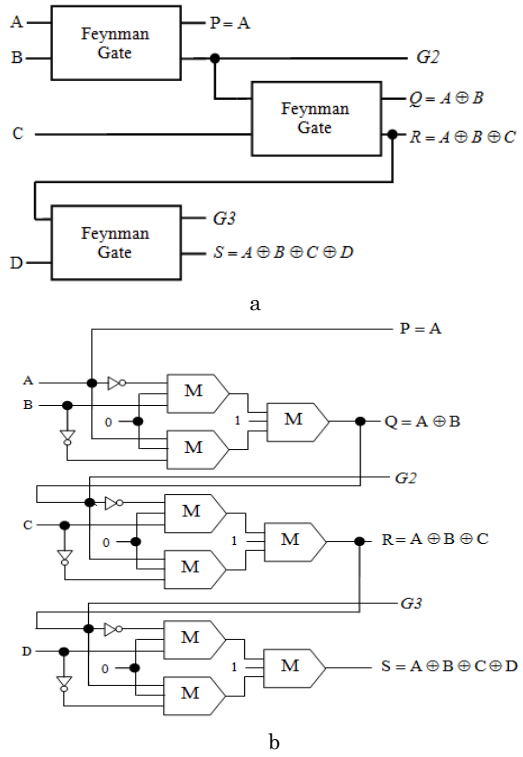
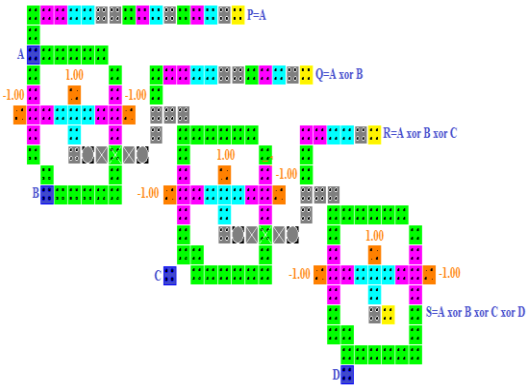


Fig. 15 – Schematics & MV representation of 3 bit reversible gray to binary code converter



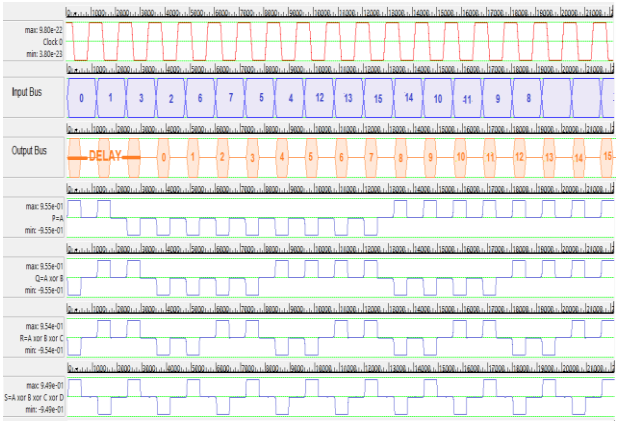


Fig. 16 – QCA implementation and simulation result of 4 bit reversible Gray to Binary code converter

Table 7 – Comparison of various reversible Binary to gray code converters

Circuit	Cell Count (μm²)	Cell Area (μm²)	Total Area (μm²)	Latency (in clock cycles)
3 bit Reversible gray to Binary code converter [17]	112	0.0362	0.1399	3
3 bit Reversible gray to Binary code converter [20]	194	0.0628	0.1944	N.A
Proposed 3 bit Reversible gray Binary to code converter	114	0.0369	0.1069	2
Novel 4 bit Reversible gray to Binary code converter	175	0.0567	0.1969	3

The same Feynman gate is used to design the reversible gray to binary code converters. The performances of the various gray to binary converters are shown in Table 7. From the Table 7 it can be seen that the converter circuit in [17] employs less number of cells as compared to the proposed one. After studying and analyzing the reversible gray to binary code converter presented in [17], it was found that the outputs (B_0, B_1, B_2) has different delays. The B_0 has three delays, B_1 has two delays and the B_2 provides a single delay thereby producing the incorrect output. If the design would have used the equal amount of delay at the output terminals the cell count would have been high. In the proposed 3 bit reversible gray to binary circuits all the output terminals (i.e. P, Q, and R) provides the equal amount of delays. Thus the proposed circuits have an edge over the other designs in term of cell count, area, latency, complexity and the garbage outputs. Furthermore using the same methodology the novel 4 bit reversible gray to binary code converter has been designed.

6. CONCLUSION

In this paper various QCA based reversible code converters were discussed and few novel circuits were also introduced. It was observed that the proposed circuits are more efficient in term of cell Area, total area, latency, complexity and the quantum cost.

REFERENCES

1. C.S. Lent, P.D. Tougaw, W. Porod, G.H. Bernstein, *Nanotechnol.* **4** No 1, 49 (1993).
2. C.S. Lent, P.D. Tougaw, *Proc. IEEE* **85** No 4, 541 (1997).
3. D. Maslov, *Reversible logic synthesis* (Doctoral dissertation: University of New Brunswick: 2003).
4. C.H. Bennett, *IBM J. Res. Develop.* **17** No 6, 525 (1973).
5. T. Toffoli, *Reversible computing Tech memo MIT.* M IT/LCS/TM-151, M IT Lab for Comp. Sci. (1980).
6. R. Feynman, *Found. Phys.* **16** No 6, 507 (1986).
7. G.J. Milburn, *Phys. Rev. Lett.* **62** No 18, 2124 (1989).
8. A. Peres, *Phys. Rev. A* **32** No 6, 3266 (1985).
9. M.M.A. Khan, *International Conference on Computer and Information Technology, Dhaka, Bangladesh*, 515 (2002).
10. M.S. Islam, M.M. Rahman, Z. Begum, M.Z. Hafiz, *Inform. Technol. J.* **8** No 2, 208 (2009).
11. A.K. Biswas, M.M. Hasan, A.R. Chowdhury, H.M.H. Babu, *Microelectronics J.* **39** No 12 1693 (2008).
12. M. Haghparast, K. Navi, *Am. J. Appl. Sci.* **5** No 3, 282 (2008).
13. B. Hertz Shargel, M.R.D'Orsogna, T. Chou, *J. Phys. A: Math. Theor.* 43, 305003 (2010).
14. N.M. Nayeem, L. Jamal, H.M. Babu, *J. Comput. Sci.* **5** No 1, 49 (2009).
15. H.R. Bhagyalakshmi, M.K. Venkatesha, *Int. J. Comput. Appl.* **32** No 3, 36 (2011).
16. P.D. Tougaw, C.S. Lent, *J. Appl. Phys.* **75** No 3, 1818 (1994).
17. Jadav Chandra Das, Debashis De, *IETE J. Res.* 1 (2015).
18. P. Biswas, N. Gupta, N. Patidar, *Int. J. Eng. Res. Appl.* **4** No 6, 12 (2014).
19. Rahman, Md Anisur, *Int. J. Computer Sci.* **10** No 4, (2013).
20. Neeraj Kumar Misra, Subodh Wairya, V.K. Singh, *Proceedings of the 4th International Conference on Frontiers in Intelligent Computing: Theory and Applications (FICTA) 2015*. (2016).