Vol 05, Article 10467, October 2014 International Journal of VLSI and Embedded Systems-IJVES
http://ijves.com ISSN: 2249 – 6556

DESIGN OF CARRY-LOOK-AHEAD ADDER USING REVERSIBLE LOGIC IMPLEMENTATION IN QCA

SUBHASHEE BASU[1], ADITI BAL[2], SUPRIYA SENGUPTA[3]
[1, 2, 3]St Thomas College of Engineering and Technology,4 Diamond Harbour Road,Kolkata 700023,India
[1]subhashreebasu1984@yahoo.co.in, [2]amiaditi@gmail.com, [3]supriyo210@yahoo.com

ABSTRACT

Various representations of reversible logics are available. Still the quest for more is there. As the reversible logic gates is an important area in terms of fault-tolerant system. Scientists are trying to implement the concept of reversibility in different domains of nanotechnology. On the other hand QCA is taken as the substitute of CMOS. The main advantage that QCA have over quantum circuits is that QCA make considerably fewer demands on the underlying hardware. In this paper, QCA is investigated for the implementation of reversible logics. Six reversible gates can be implemented using QCA basic gates which are unique from the already existing reversible gates (like Toffoli and Fredkin). In the proposed Carry-look-ahead adder circuit both QCA as well as quantum reversible gates are implemented using QCA basic gates and are compared in terms of number of gates and delay. It is seen that the QCA reversible gates require much lesser gates than the conventional quantum gates which are realized using QCA basic gates. Hence the area and delay is also much less.

Keywords: *QCA , Reversible logic gates, Carry-look-ahead adder.*

[1] INTRODUCTION

Extensive research in the field of nanotechnology is being done in order to overcome the limitations of CMOS. In fact, the prime motivation for studying reversible computing is to reduce power dissipation in computing machinery, and thus achieve higher density and speed. Another most important characteristic of reversible circuit[3][21] is that any operation that in conventional logic lead to the destruction of macroscopic information, and thus lead to energy dissipation, can be replaced by an information lossless invariant. As a result, it becomes possible to design circuits whose initial power dissipation, under ideal physical circumstances, is zero. The power dissipation that would arise at the interface between such circuit and the outside world would be at most proportional to the number of inputs/output lines rather than the number of logic gates.

QCA is an emerging technology. Different designs of logic circuits have been proposed for QCA implementation[8][12].The basic logic devices in QCA are the majority voter(MV) and the inverter(INV).The majority voting function is logically irreversible, because the information in the minority input is lost in the computation. In this paper, new reversible logic gates are defined in QCA.As QCA logic is based on majority voting, the definition of a reversible gate, must take into consideration, the unique features of majority voter gate.

1.1 PRELIMINARIES

1.1.1 *Reversible Circuits Basics*

Traditional logic functions(such as AND and OR) are not reversible, because more than one input state is mapped to a common output state. Given a output state, it is not possible to determine the input state. Thus information is lost during computation process.INV is a simple example of a reversible logic. The basic principle of reversible computing [3][21] is that no information is lost during the computation process and the entire computing process is invertible i.e. for each final state of the system, a unique initial state can be determined. A reversible logic function is a one-to- one mapping between inputs and outputs i.e. each input pattern has a unique output pattern mapped to it and vice versa[14].Using reversible logic, it is ideally possible to design circuits with no power dissipation. Even if reversible circuit is interfaced with irreversible circuit,it is seen that heat dissipation in the worst case is proportional to the number of inputs and outputs. Toffoli and Fredkin gates[17][13] are the most common reversible gates. Both these gates have three inputs and three outputs and are universal as any combinational logic can be realized using these gates. Fredkin gate is also conservative i.e. the number of 1s in the output is equal to the number of 1s in the input. The truth tables of both the gates are shown below.

1.1.2 QCA Basics

The quantum dot cellular automata[11][10] use a binary representation of in-formation, by replacing the current switch with a cell having a bistable charge configuration. One configuration of charge represents a binary 1, the other a 0, but no current flows into or out of the cell. The field from the charge con-figuration of one cell alters the charge configuration of the next cell. The basic unit of QCA is a QCA cell[16][12] which contains four quantum-dots at four corners of a square and two mobile electrons. These electrons can move to any of the quantum dots by electron tunneling. Due to Coulombic interaction, these electrons usually tend to reside at the diagonal corners of the square, giving rise to two configurations that are stable. Assigning polarization +1 and -1

Vol 05, Article 10467, October 2014 International Journal of VLSI and Embedded Systems-IJVES
http://ijves.com ISSN: 2249 – 6556

to these configurations lead to the two binary logic state 1 and 0.

Table 1 Truth table of Toffoli Gate

	Input		Output		
C1	C2	A	C1	C2	A (C1C2)
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Table 2 Truth table of Fredkin Gate

Input			Output		
C1	C2	A	C1	C2	A (C1C2)
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Input			Output		
C	A	B	C	C'A (CB)	CA (C'B)
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

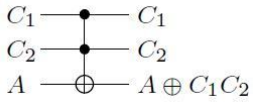


Fig. 1 Toffoli gate

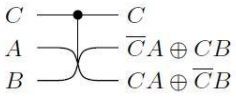


Fig. 2 Fredkin Gate

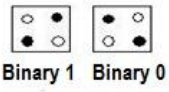


Fig. 3 QCA Cell

The QCA cells can be arranged serially to form QCA wire which acts as the propagating medium for binary information. Majority voter gate and inverter[15] acts as the basic gates in QCA[8][20][24].

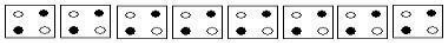


Fig. 4 QCA Wire

The governing equation of majority voter gate is

$$M(A; B; C) = AB + BC + AC \quad (1)$$

Two input AND and OR gate can be implemented from three input majority voter gate by making one input constant.

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

$$M(A; B; 0) = AB \quad (2)$$

Fig 5 and Fig 6 shows the gate symbols and their layouts.

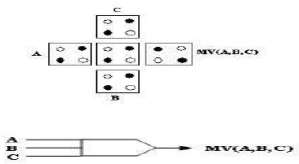


Fig. 5 QCA Majority Voter Gate

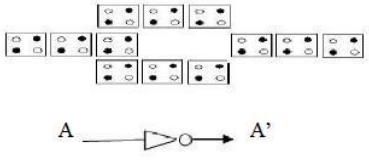


Fig. 6 QCA Inverter Gate

2. PROPOSED METHOD

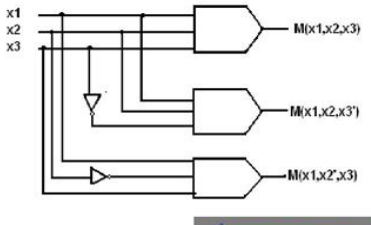
2.1 QCA reversible gates

The majority circuit logic is in general irreversible[1][5][6][22][19][2][7][9][4][23][18] as the minority input is lost. But by passing certain combinations of the inputs via the majority gate, we can build a QCA reversible circuit. For example, let there be three inputs x1,x2,x3.Passing these inputs into three majority gates such that the outputs are y1=M(x1,x2,x3),y2=M(x1,x2,x3'),y3=M(x1,x2',x3) will make the circuit behave like a reversible one.It is seen that six such combinations exist for three input lines producing six reversible circuits. They can be accordingly named as REV1,REV2,REV3,REV4,REV5 and REV6.Their majority expression

Vol 05, Article 10467, October 2014
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556

along with their respective circuit diagram, truth table and the layout design is shown below.

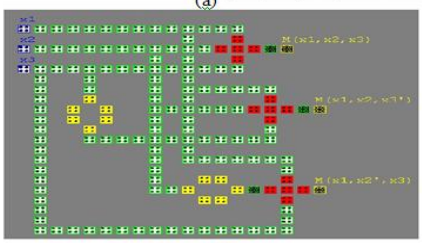
$y1 = M(x1; x2; x3); y2 = M(x1; x2; x3^{'}); y3 = M(x1; x2^{'}; x3)$ (3)



(a)

INPUTS			OUTPUTS		
x1	x2	x3	M(x1,x2,x3)	M(x1,x2,x3')	M(x1,x2',x3)
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	0	1	1
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	1	1	1

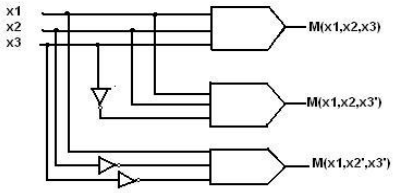
(b)



(c)

Fig. 7 REV1:(a)circuit diagram (b)truth table (c)layout

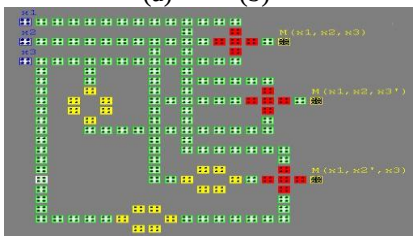
$y1 = M(x1; x2; x3); y2 = M(x1; x2; x3^{'}); y3 = M(x1; x2^{'}; x3^{'})$ (4)



(a)

INPUTS			OUTPUTS		
x1	x2	x3	M(x1,x2,x3)	M(x1,x2,x3')	M(x1,x2',x3')
0	0	0	0	0	1
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	0	1	1
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

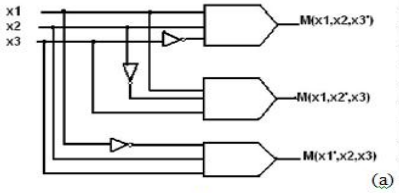
(b)



(c)

Fig. 8 REV2:(a)circuit diagram (b)truth table (c)layout

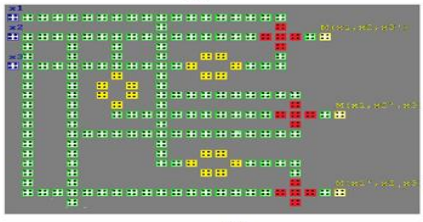
$y1 = M(x1; x2; x3^{'}); y2 = M(x1; x2^{'}; x3); y3 = M(x1^{'}; x2; x3)$ (5)



(a)

INPUTS			OUTPUTS		
x1	x2	x3	M(x1,x2,x3')	M(x1,x2',x3)	M(x1',x2,x3)
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	1	0	1
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	0	1	0
1	1	0	1	0	0
1	1	1	1	1	1

(b)

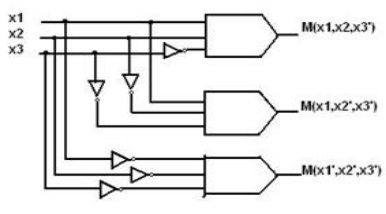


(c)

Fig. 9 REV3:(a)circuit diagram (b)truth table (c)layout

$y1 = M(x1; x2; x3); y2 = M(x1; x2^{'}; x3^{'}); y3 = M(x1^{'}; x2^{'}; x3^{'})$ (6)

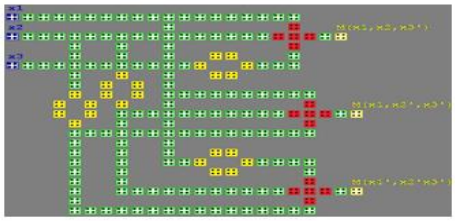
Vol 05, Article 10467, October 2014
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556



(a)

INPUTS			OUTPUTS		
x1	x2	x3	M(x1,x2,x3')	M(x1,x2',x3')	M(x1',x2',x3)
0	0	0	0	1	1
0	0	1	0	0	1
0	1	0	1	0	1
0	1	1	0	0	0
1	0	0	1	1	1
1	0	1	0	1	0
1	1	0	1	1	0
1	1	1	1	0	0

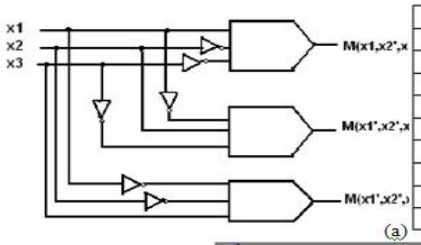
(b)



(c)

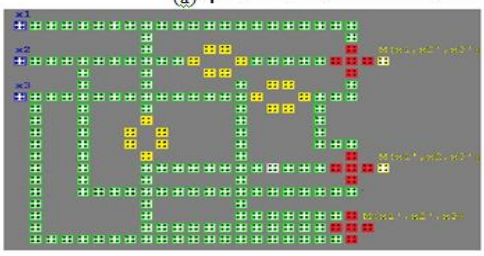
Fig. 10 EV4:(a)circuit diagram (b)truth table (c)layout

$y1 = M(x1; x2^{'}; x3^{'}); y2 = M(x1^{'}; x2; x3^{'}); y3 = M(x1^{'}; x2^{'}; x3^{'})$ (7)



INPUTS			OUTPUTS		
x1	x2	x3	M(x1,x2',x3')	M(x1',x2,x3')	M(x1',x2',x3)
0	0	0	1	1	1
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	0	0	0

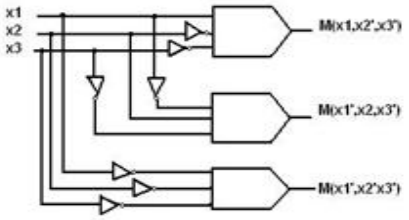
(a) (b)



(c)

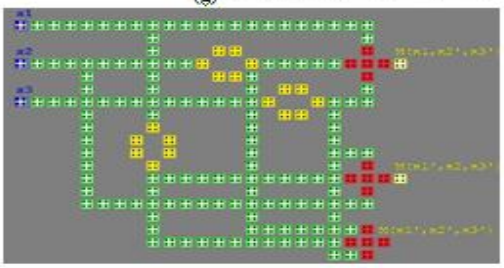
Fig. 11 REV5:(a)circuit diagram (b)truth table (c)layout

$y1 = M(x1; x2^{'}; x3^{'}); y2 = M(x1^{'}; x2^{'}; x3); y3 = M(x1^{'}; x2^{'}; x3^{'})$ (8)



INPUTS			OUTPUTS		
x1	x2	x3	M(x1,x2',x3')	M(x1',x2,x3')	M(x1',x2',x3)
0	0	0	1	1	1
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	0	1
1	0	1	1	0	0
1	1	0	1	1	0
1	1	1	0	0	0

(a) (b)



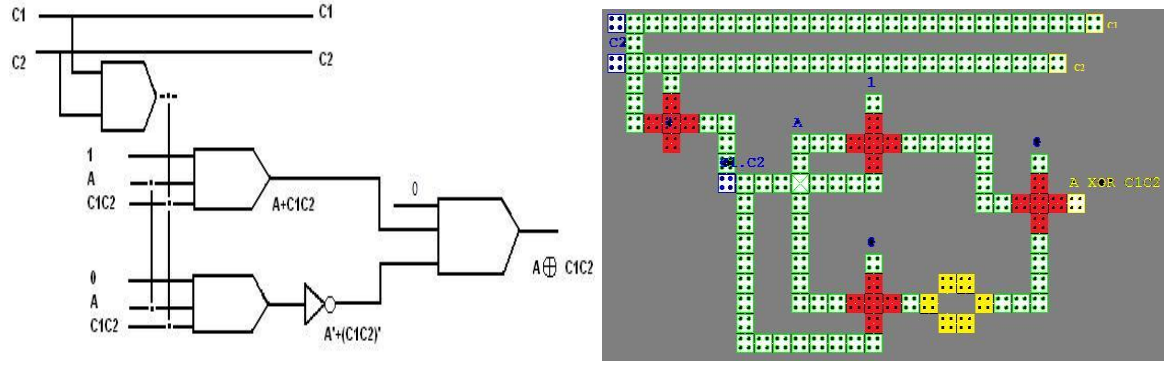
(c)

Fig. 12 REV6:(a)circuit diagram (b)truth table (c)layout

The conventional three input reversible gates i.e Toffoli and Fredkin gates are also implemented using QCA. Their circuit diagram formed by exploiting the unique characteristics of majority voter gate as well as the layout

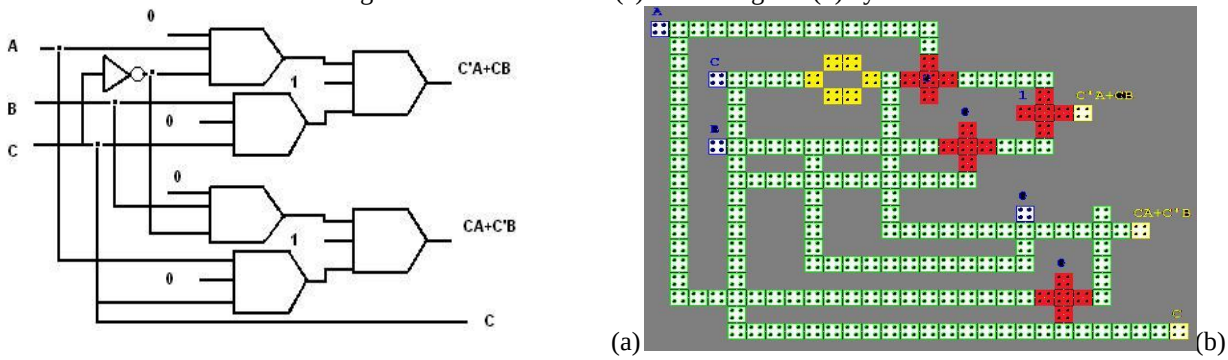
Vol 05, Article 10467, October 2014 International Journal of VLSI and Embedded Systems-IJVES
http://ijves.com ISSN: 2249 – 6556

designed using QCADesigner is shown below



(a) (b)

Fig. 13 TOFFOLI GATE:(a)circuit diagram (b)layout



(a) (b)

Fig. 14 FREDKIN GATE:(a)circuit diagram (b)layout

3. CARRY -LOOK-AHEAD ADDER

Carry propagation time is a limiting factor on the speed with which two numbers are added in parallel[1][5][6].Since all other arithmetic operations are implemented by successive additions, the time consumed during the addition process is very critical. An obvious solution for reducing the carry propagation delay time is to employ faster gates with reduced delays. But physical circuits have a limit to their capability. There are several techniques for reducing the carry propagation time in parallel adder. The most widely used technique employs the principle of look-ahead carry.

$$S_i = P_i \quad C_i \oplus$$

$$C_(i + 1) = G_i + P_i{\cdot}C_i \qquad (9)$$

Where G_i is called the carry generate and it produces an output carry when both A_i and B_i are 1,regardless of the input carry. P_i is called the carry propagate because it is the term associated with the propagation of carry from C_i to $C_(i + 1)$.The boolean function for the carry output of each stage is substituted by the carry expression of the previous equations.

$$C_1 = G_0 + P_0 \,.\, C_0$$
$$C_2 = G_1 + P_1 \,.\, C_1$$
$$C_3 = G_2 + P_2 \,.\, C_2$$
$$C_4 = G_3 + P_3 \,.\, C_3 \qquad (10)$$

Substituting C_1 into C_2, then C_2 into C_3, then C_3 into C_4 yields the expanded equations:

$C_1 = G_0 + P_0 \,.\, C_0$
$C_2 = G_1 + G_0 \,.\, P_1 + C_0 \,.\, P_0 \,.\, P_1$
$C_3 = G_2 + G_1 \,.\, P_2 + G_0 \,.\, P_1 \,.\, P_2 + C_0 \,.\, P_0 \,.\, P_1 \,.\, P_2$
$C_4 = G_3 + G_2 \,.\, P_3 + G_1 \,.\, P_2 \,.\, P_3 + G_0 \,.\, P_1 \,.\, P_2 \,.\, P_3 + C_0 \,.\, P_0 \,.\, P_1 \,.\, P_2 \,.\, P_3 \qquad (11)$

In the above expressions consist of C_0, C_4 does not have to wait for C_2 and C_3 to propagate. In fact, C_4 is propagated at the same time as C_2 and C_3.

But, instead of such a complex expression, we can reduce the expression of the carry and sum of each full adder as follows.

$$Cout = a'b'Cin + a'b'C_in + a'b'C_in + a'b'C_in$$

Vol 05, Article 10467, October 2014 International Journal of VLSI and Embedded Systems-IJVES
http://ijves.com ISSN: 2249 – 6556

$$Cout = a'b'C_in + a'b'C_in + a'b'C_in + a'b'C_in + a'b'C_in + a'b'C_in$$

$$Cout = a'b'(C_in + C_in) + b'C_in'(a + a) + a'C_in'(b + b)$$

$$Cout = a'b + b'C_in + a'C_in$$

$$Cout = m(a; b; C_in) \quad (12)$$

$$Sum = a'b'C_in + a'b'C_in + a'b'C_in + a'b'C_in$$

$$Sum = (a'b'C_in + a'b'C_in) + (a'b'C_in + a'b'C_in) + (a'b'C_in + a'b'C_in)$$

$$Sum = (a'b + a'Cin + b'Cin)(a'b + a'Cin + b'Cin) + (a'b + b'Cin + a'Cin)$$

$$(a'b + b'Cin + a'Cin) + (a'b + a'C_in + b'C_in)(a'b + a'C_in + b'C_in) \quad (13)$$

$$Sum = m(a; b; C_in)!m(a; b; C_in) + m(a; b; C_in)'m(a; b; C_in) + m(a; b; C_in)'m(a; b; C_in)$$

$$Sum = m(m(a'; b; C_in); m(a; b'; C_in); m(a; b; C_in')) \quad (14)$$

If we observe carefully, it will be seen that the expression of Sum is similar to that of reversible gate REV3.So,we can replace each full adder by the reversible circuit REV3.The circuit diagram and QCA layout of carry-look-ahead adder is shown below.

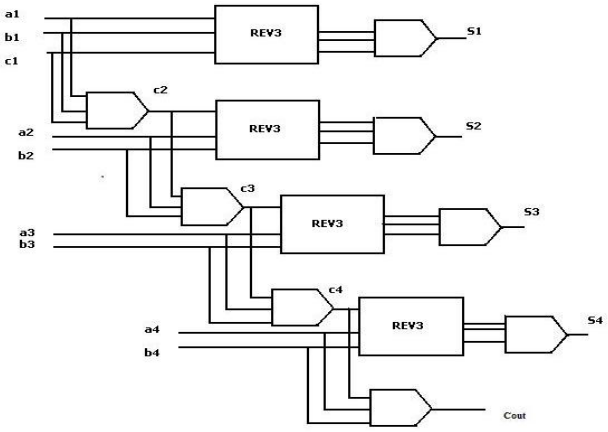


Fig. 15 Carry-look-ahead Adder using QCA majority gates

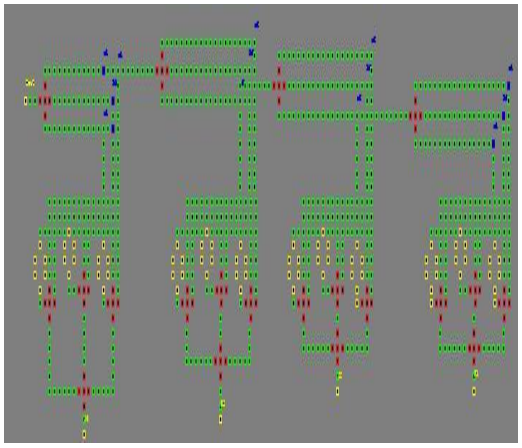


Fig. 16 Layout of Carry-look-ahead Adder

4. RESULT ANALYSIS

As shown in table, different figures of merit are used in the comparison of the synthesis method.

Properties	REV1	REV2	REV3	REV4	REV5	REV6	TOFFOLI	FREDKIN
Area	0.21 m^2	0.22 m^2	0.25 m^2	0.25 m^2	0.24 m^2	0.24 m^2	0.22 m^2	0.22 m^2
No:ofgates	5	6	6	6	6	6	5	7
No:ofcells	160	162	208	174	205	174	207	195
Delay	1/2	1/2	1/2	1/2	1/2	1/2	0	0

As seen in the above table, the improvement of QCA reversible gates com-pared to that of Toffoli and Fredkin Gates are not very significant. But still, QCA reversible gates can be advantageous over Toffoli and Fredkin gates because all the majority voter outputs can be utilized as disjoint devices unlike Tof-foli/Fredkin gates where some of the outputs are just the extension of the in-puts. This advantage is mainly realized while implementing some benchmark circuits where all the MVs can be used to get the outputs unlike Toffoli/Fredkin gates where some of the outputs would be redundant. Moreover, implementing carry-look-ahead adder using QCA reversible gates reduces the heat loss significantly as no signals are lost. It may seem that there will be an increase in the area but simulating the above layout gives an area of 1.17 m^2 which is a very small measurement .Apparently, it may appear that there will be a delay in the circuit. But, since the delay in QCA is itself negligible, we can infer that all the five outputs will be generated concurrently.

5. CONCLUSION

This paper has presented a comprehensive synthesis of reversible circuit implementation in quantum dot cellular automata. Initially, six reversible circuits (denoted by REV1, REV2, REV3, REV4, REV5, REV6) are proposed. These gates take into account, the primitive characteristics of QCA majority gate to retain the one-to-one mapping of inputs and outputs. Conventional reversible gates(like Toffoli and Fredkin) are also implemented in QCA for comparison. However, this is not a strictly reversible logic synthesis method as fanouts and fixed polarization is allowed in QCA circuits.

Vol 05, Article 10467, October 2014 International Journal of VLSI and Embedded Systems-IJVES
http://ijves.com ISSN: 2249 – 6556

REFERENCES

[1] Mostafa Rahimi Azghadi*, O. Kavehei, and K. Navi. A novel design for quantum-dot cellular automata cells and full adders. *Journal of Applied Sciences*, 7:3460, 2007.
[2] S. Basu and S. Bhattacharjee. Implementation of symmetric functions using quantum dot cellular automata. In Lakhmi C. Jain Robert J. Howlett, editor, *Smart Innovation, Systems and Technologies*, number 28 in Second International Conference on Advanced Computing, Networking and Informatics (ICACNI-2014), page 451=460, Switzerland, June 2014. Springer.
[3] C. Bennett. Logic reversibilty of computation. *IBM Journal of Research and Development*, 17:525532,1973.
[4] Pijush Kanti Bhattacharjee. Use of symmetric functions designed by qca gates for next generation ic. *International Journal of Computer Theory and Engineering*, 2:1793– 8201, April 2010.
[5] Heumpil Cho. Adder designs and analyses for quantum-dot cellular automata. *IEEE Transactions on Nanotechnology*, 6:374 – 383, May 2007.
[6] Heumpil Cho and Earl E. Swartzlander. Adder and multiplier design in quantum-dot cellular automata. *IEEE Transactions on Computers*, 58:721 – 727, June2009.
[7]C.Rovetta and M.Mouffron. De bruijan sequences and complexity of symmetric func-tions. *Cryptography and Communications journal*, 3:207–225, December 2011.
[8]A. O. Orlov I. Amlani, G. Toth, C. S. Lent, G. H. Bernstein, and G. L.Sinder. Digital logic gate using quantum dot cellular automata. *Science*, 284:289–291, April 1999.
[9]Hema Sandhya Jagarlamudi, Mousumi Saha, and Pavan Kumar Jagarlamudi. Quantum dot cellular automata based effective design of combinational and sequential logical structures. *World Academy of Science, Engineering and Technology*, 5:12–25, 2011.
[10]C. S. Lent and P. D. Taugaw. A device architecture for computing with quantum dots. volume 85, pages 541–557. IEEE, 1997.
[11]C. S. Lent, P. D. Taugaw, W. Porod, and G. H. Berstein. Quantum cellular automata. *Nanotechnology*, 4:49–57, 1993.
[12]M. Lieberman, S. Chellamma, B. Varughese, Y.Wang, C. S. Lent, G.H. Bernstein, G.L.Snider, and F. Peiris. Quantum dot cellular automata at a molecular scale. *Annals of the New York Academy of Sciences*, 960:225–239, 2002.
[13]D. Maslov, G. W. Dueck, , and M. D. M. Synthesis of fredkin-toffoli reversible networks. *IEEE Transcation on VLSI*, 13:765 – 769, 2004.
[14]Zahra Mohammadi, Majid Mohammadi, and Mahdi Hasani. Designing of testable re-versible qca circuits using a new reversible mux 2:1. *Journal of Advances in Computer Research*, 3:51–64, February 2012.
[15]M. Momenzadeh, M. B. Tahoori, J. Huang, and F. Lombardi. Characterization, test and logic synthesis of and-rinverter (aoi) gate design for qca implementation. *IEEE Trans on Computer Aided Design of Integrated Circuits and Systems*, 24:1881–1893, December 2005.
[16]A. O. Orlov, I. Amlani, G. H. Bernstein, C. S. Lent, and G. L.Sinder. Realization of a functional cell for quantum dot cellular automata. *ScienceScience*, 277:928–930, August1997.
[17]P.Picton. Modified fredkin gates in logic design. *Microelectronics Journal*, 25:437–441, 1994.
[18]H. Rahaman, B. K. Sikdar, and D.K. Das. Synthesis of symmetric boolean functions using quantum cellular automata. 10:119–124, 2006.
[19]Geza Toth* and Craig S. Lent. Quantum computing with quantum-dot cellular au-tomata. *Physical Review A*, 63:052315, 2001.
[20]Whitney J. Townsend and Jacob A. Abraham. Complex gate implementations for quantum dot cellular automata. pages 625–627, August2004.
[21]T.Toffoli. Reversible computing. TWI Report 151, MIT Laboratory for Computer Science, 1980.
[22]K. Walus, G. Schulhof, G. A. Jullien, R. Zhang, and W. Wang. Circuit design based on majority gates for application with quantum dot cellular automata. *IEEE Trans Signals, Systems and Computers*, 2:1354–1357, Nov 2004.
[23]Z. Y. Xu, M. Fenga, and W. M. Zhang. Universal quantum computation with quantum-dot cellular automata in decoherence-free subspace. *Quantum Information and Com-putation, Rinton Press*, 8:977–986, 2008.
[24]R. Zhang, K.Walus, W.Wang, and G.A.Jullien. A method of majority logic reduction for quantum cellular automata. *IEEE Trans on Nanotechnology*, 3:443– 450, Dec2004.