Design and Application of Universal Logic Gate based on Quantum-Dot Cellular Automata

Yinshui Xia, Keming Qiu
Institute of Circuits and Systems
Ningbo University, Ningbo, China
xiayinshui@nbu.edu.cn, qiukemingfeng@163.com

***Abstract*—Wire-crossing is an issue in quantum-dot cellular automata (QCA) design. In this paper, QCA based universal logic gate (ULG) is proposed to reduce the number of wire-crossings. Using the ULGs, full adder/subtraction, full comparator and 4-to-1 multiplexer are designed. QCADesigner simulation results show that the proposed circuits have correct logic function. Compared with traditional design based on majority gates and inverters (MIs), the ULG based design can reduce the number of wire-crossings.**

Keywords-quantum-dot cellular automata; universal logic gate; majority gate; wire-crossing.

I. INTRODUCTION

CMOS faces some challenging problems, such as high power consumption and difficulties in further feature size scaling. Research predicts that CMOS technology may hit physical scaling limits in 2012 and will be superseded by some emerging technologies [1]. Quantum-dot cellular automata (QCA) introduced by Lent in [2], is one of these technologies. Its unique feature is that logic states are represented by the position of electrons.

In current QCA based design, the fundamental logic element is the majority gate. However, majority gate itself is not logic competed. Hence, majority gate and inverter (MI), which consist of logic complete set, are used to design QCA circuits. Adder [3,4], comparator [5], multiplier [6] and multiplexer designs [7] are some typical examples. Currently, QCA layout is restricted to a single layer with very limited number of wire-crossings permitted. Therefore, minimization of wire-crossings is one of the major issues for QCA based design. However, MI based design may not be optimized in terms of wire-crossings. In this paper, universal logic gate (ULG) is proposed. Some common combinational circuits are designed based on ULGs. Compared with MI based design, ULG based design can reduce the number of wire-crossings.

II. QCA BASICS

A. QCA Cell

A QCA cell is a nano-scale device which can store logic states and transmit information by Coulomb interaction. As shown in Fig. 1, each cell is composed of four quantum dots arranged in a square. The cell is charged with two extra electrons, which tend to occupy diagonally opposed dots as a result of their Coulomb repulsion. Thus, there are two possible arrangements denoted as cell polarization P=+1 and P=-1, respectively. By using cell polarization P=+1 to represent logic "1" and P=-1 to represent logic "0", binary information can be encoded.

978-1-4244-2251-7/08/$25.00 ©2008 IEEE

B. QCA Clock

Clock in QCA based circuits is not only to synchronize and control information flow but also to provide the power to run the circuits. Normally, a QCA based circuit is divided into four sections and is controlled by four phase clock signals. As shown in Fig. 2, each clock signal lags in phase by 90 degrees with respect to the one before it. It is known that when clock is high, the cell is unlatched and no net polarization exists, i.e. P=0; when clock is low, the cell is latched and cells take on the polarization.

C. QCA Logic Gates

The logic function of majority gate is

$$M(A,B,C) = AB + BC + AC \tag{1}$$

Both logic AND and OR functions can be implemented by setting one input into 0 or 1. Since the majority logic gate only is not logic completed, QCA inverter is also used to form a completed logic set. Fig. 3 shows the gate symbols and their layouts.

D. QCA Wire

There are two kinds of QCA wires. One is binary wire implemented with the cells of 90 degree orientation. The other is inverter chain implemented with the cells of 45 degree orientation. Fig. 4 shows the QCA wires and wire-crossing structure.

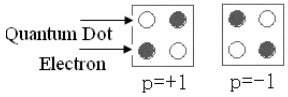


Figure 1. QCA cell

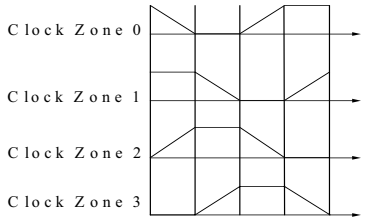


Figure 2. QCA clock

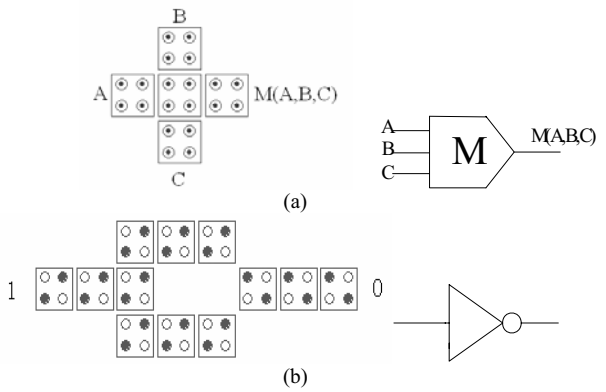


Figure 3. Logic gates (a) Majority gate; (b) Inverter

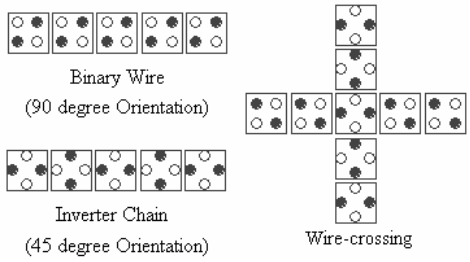


Figure 4. QCA wire and wire-crossing structure

In coplanar wire-crossing, cells oriented at 90 degrees and 45 degrees are used. However, manufacturing nano-scale cells with two different orientations is a challenging task [1]. Furthermore, excessive wire-crossing structures may affect the function of the circuit [8]. Thus, it is important to minimize the number of wire-crossings.

III. UNIVERSAL LOGIC GATE

A. Definition

A function with m-inputs can achieve any function with a given number of variables, known as universal logic function, the corresponding gate is known as universal logic gate (ULG). The gate which can implement any n-variable function is known as n variables universal logic gate (ULG.n).

In [9], six possible candidates to implement the ULG were proposed. Among them, the function of ULG as in (2) is chosen in this work.

$$f(y_1, y_2, y_3) = y_1 y_2 + \bar{y}_1 \bar{y}_3 \tag{2}$$

From (2), the XNOR and multiplexer function can be implemented easily.

B. QCA Based ULG Implementation

By using (1) and (2), we can obtain

$$f(y_1, y_2, y_3) = M(M(y_1, y_2, 0), \overline{M(y_1, y_3, 1)}, 1) \tag{3}$$

Fig. 5 shows its logic symbol, four clocked QCA cells with different colors and the QCA implementation. In the implementation, four different colors are used to tell from the corresponding clock signals to drive the circuits. It can be seen that there are 3 wire-crossings in the design.

The QCA implementation of the ULG is simulated by the QCADesigner tool [10]. The parameters used in [4] are employed: the diameter of quantum dot is 5nm and the cell size is 18nm×18nm; the cell distance is 2nm and the grid space is 20nm. During simulation, the simulation engine, bistable approximation, is chosen, that is: number of samples=128000, convergence tolerance=0.001, radius of effect=65nm, relative permittivity=12.9, clock high=9.8e-22, clock low=3.8e-23, clock amplitude factor=2, layer separation=11.5nm, maximum iterations per sample=100 [11].

The input and output waveforms are shown in Fig. 6. The first three waveforms represent the input signals. The fourth waveform represents the output signal. The first meaningful output appears in second clock tick after 1 clock delay. It can be seen that the ULG has correct logic function.

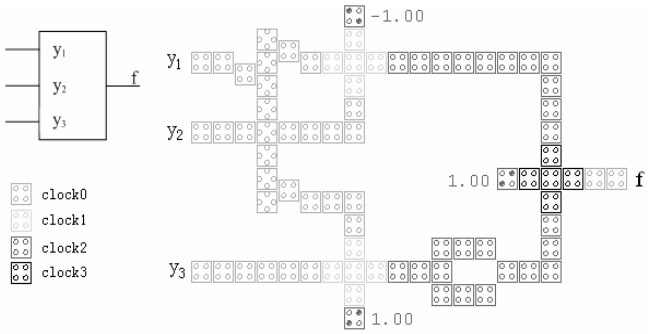


Figure 5. Logic symbol and QCA implementation of the ULG

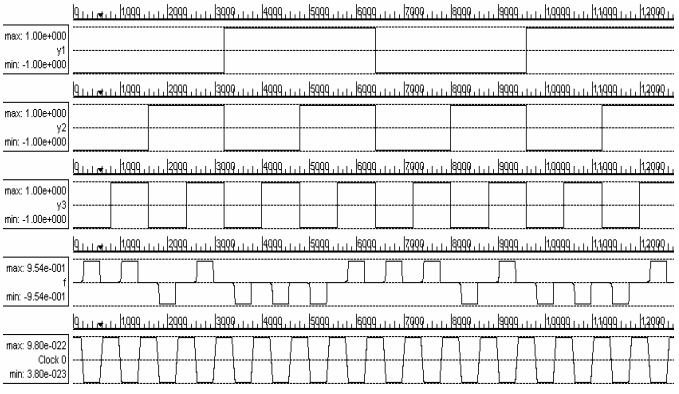


Figure 6. Simulation waveforms of the ULG

C. *Example of QCA Circuits*

Three kinds of QCA circuits, full adder/subtraction, full comparator and 4-to-1 multiplexer are designed by using the ULGs. Three parameters are used to measure circuit performance: the number of wire-crossings, the number of QCA cells and time delay.

Take a full comparator for example. The function of the full comparator can be expressed as in (4)

$$\begin{cases} Gn = (AB + \overline{A}\overline{B})G + A\overline{B} \\ Ln = (AB + \overline{A}\overline{B})L + \overline{A}B \end{cases} \tag{4}$$

By using (2) and (4), (5) can be obtained.

$$\begin{cases} Gn = f(f(B,A,A),G,B) \\ Ln = f(f(B,A,A),L,A) \end{cases} \tag{5}$$

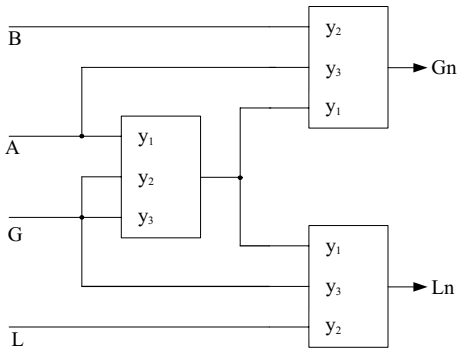
From (5), its logic schematic diagram and QCA implementation can be obtained as shown in Fig. 7. Fig. 8 shows the logic schematic diagram and QCA implementation of the full comparator based on the MI design. Both of circuits are simulated by QCADesigner tool. The results show that the circuits have correct logic function.

From Fig. 7(b), it can be seen that for the ULG based design, 353 cells are required, there are 9 wire-crossings in the circuit and the time delay is 2.25 clocks while for the MI based design as in Fig. 8(b), it needs 222 cells, the number of wire-crossings is 11, and the time delay is 2 clocks. Compared to the MI based design, the ULG based implementation achieves 22.2% reduction in terms of the number of wire-crossings.

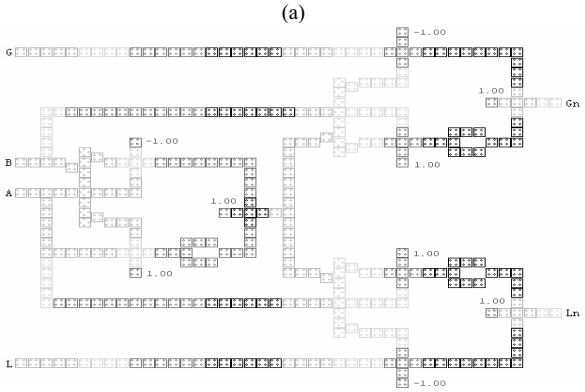
Similar analysis is pursued for full adder/subtraction and 4-to-1 multiplexer. To save the space, the detail is not given here. TABLE I summarizes the result implementing these circuits by using ULG and MI based design, respectively. From TABLE I, it can be seen that ULG based design has advantage in term of the reduction of wire-crossings. It should be pointed out that in the design of 4-to-1 multiplexer, the ULG based design requires fewer number of QCA cells than MI based design apart from the reduction of wire-crossings.

IV. CONCLUSIONS

Minimization of wire-crossings is crucial in QCA design. ULG is proposed to reduce the number of wire-crossings. Three classic circuits are designed to test the efficiency of the proposed approach. The QCADesigner tool is used to simulate the circuits. The results show that ULG based design can reduce the number of wire-crossings compared with MI based design.

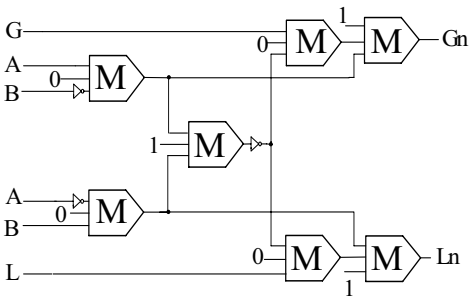


(a)

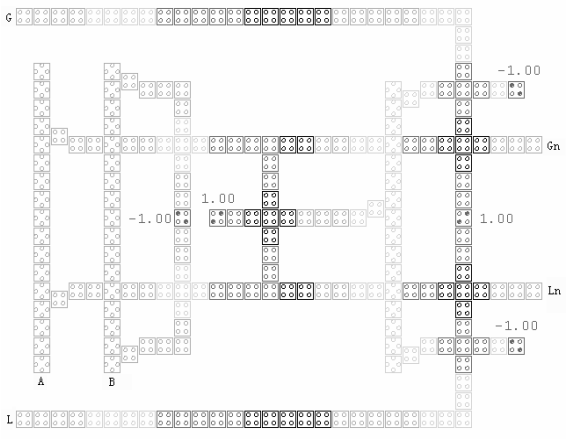


(b)

Figure 7. (a) Logic schematic diagram of the full comparator of ULG based design; (b) QCA implementation of the full comparator of ULG based design



(a)



(b)

Figure 8. (a) Logic schematic diagram of the full comparator of MI based design; (b) QCA implementation of the full comparator of MI based design

TABLE I. CIRCUITS USING ULG AND MI BASED DESIGN

Circuit		Full adder/ subtraction	Full comparator	4-to-1 multiplexer
ULG based design	QCA cells	432	353	273
	Time delay (clocks)	2.25	2.25	2.25
	Number of wire crossings	15	9	12
MI based design	QCA cells	288	222	287
	Time delay (clocks)	1.5	2	2
	Number of wire crossings	16	11	16

ACKNOWLEDGMENT

This work is supported by National Natural Science Fund (60676017), Zhejiang Natural Science Fund (R105614, Y106818), Science & Technology Program from Department of Zhejiang Sci. & Tech. (Grant No. 2007C24017) and Return Fund Program from National Education Ministry.

REFERENCES

[1] D. A. Antonelli, D. Z. Chen, T. J. Dysart, X. S. Hu, A. B. Kahng, P. M. Kogge et al. "Quantum-dot cellular automata (QCA) circuit partitioning: Problem modeling and solutions," Proceedings of the 41st ACM/IEEE Design Automata Conference (DAC-04), San Diego, CA, June, pp. 363-368, 2004.

[2] C. S. Lent, P. D. Tougaw and W. Porod, "Bistable saturation in coupled quantum dots for quantum cellular automata," Applied Physics Letters, 62(7), pp. 714-716, 1993.

[3] Shen Wang, Li Cai, lv Guo, "A novel full adder implementation using quantum cellular automata," Research and Progress of SEE Solid State Electronics, 25(2), pp. 148-151, 2005.

[4] H. Cho, E. E. Swartzlander, "Adder designs and analyses for quantum-dot cellular automata," IEEE Transactions on Nanotechnology, 6(3), pp. 374-383, 2007.

[5] K. M. Qiu, Y. S. Xia, "Quantum-dots cellular automata comparator," 2007 7th International Conference on ASIC Proceedings, Gui Lin, China, pp. 1297-1300, Oct.26-29, 2007.

[6] H. Cho, E. E. Swartzlander, "Serial parallel multiplier design in quantum-dot cellular automata," IEEE Symposium on Computer Arithmetic, Montpellier, France, pp. 7-15, Jun. 25-27, 2007.

[7] A. Gin, S. Williams, H. Meng, P. D. Tougaw, "Hierarchical design of quantum-dot cellular automata devices," Journal of Applied Physics, 85(7), pp. 3713-3720, 1999.

[8] K. Kim, K. Wu, R. Karri, "Quantum-dot cellular automata design guideline," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E89-A(6), pp. 1607-16, 2006.

[9] X. X. Chen, X. W. Wu, Moderm digital theory, Zhejiang University Press, 2001.

[10] K. Walus, T. J. Dysart, G. A. Jullien, A. R. Budiman, "QCADesigner: A rapid design and simulation tool for quantum-dot cellular automata," IEEE Transactions on Nanotechnology, 3(1), pp. 26-31, 2004.

[11] QCADesigner Documentation [Online]. http://www.qcadesigner.ca.