International Journal of Control and Automation
Vol. 9, No. 4 (2016), pp. 347-358
http://dx.doi.org/10.14257/ijca.2016.9.4.34

Low Hardware Complexity QCA Decoding Architecture Using Inverter Chain

Jun-Cheol Jeon

Department of Computer Engineering, Kumoh National Institute of Technology
61, Daehak-ro, Gumi, Gyeongbuk 730-701, Korea
jcjeon@kumoh.ac.kr

Abstract

This paper presents a quantum-dot cellular automata (QCA) 3-8 decoder with low hardware complexity. The proposed architecture is based on inverter chains to get inverse value and cross over a wire. We also use 5-input majority voting gates for 3-input AND gates. In this paper, we focus on the hardware complexity to reduce the number of cells and minimize wasting an area. Our architecture has not only an excellent regularity and scalability but a strong signal strength so that it is easy to extend a structure and connect with other circuit.

***Keywords**: Quantum-dot cellular automata, Decoder, Hardware complexity, Inverter chain*

1. Introduction

It was as early as 1965 when Gordon Moore predicted that the number of transistors that can be integrated on to a single chip will double every 18 months [1]. This law put forth by Moore has been a benchmark for semiconductor scaling for more than four decades. The IC industry which has been primarily driven by CMOS technology scaling is now forced to look into other alternatives as the scaling is fast approaching its fundamental limits. The International Technology Roadmap for Semiconductors (ITRS) has predicted that size limit of CMOS technology will be limited to about 5nm to 10nm and believes this limit will be reached as early as 2017 [2]

A trend towards nanotechnology is an inevitable path due to the scaling down of CMOS technology [3]. This scaling down has resulted in an increase in computing powers but at the same time as a number of problems have been discovered which have prompted the need to find an alternative to CMOS [4]. Quantum-dot cellular automata (QCA) is one such nanotechnology that is proposed to take over CMOS circuit designs. This technology offers low power consumption, very high density circuits and very high computation speeds than current CMOS technologies [5].

As device feature sizes decrease steadily with the shrinking of semiconductor transistor size, power dissipation has become clearly identified as a key limiter of continued CMOS (complementary metal-oxide silicon) scaling. Beyond the particular components of the problem in CMOS (*e.g.,* static power versus dynamic power, drain-induced barrier lowering, *etc*) lurk fundamental questions of heat dissipation and device operation. Just how small can a computational device be? How much heat must it generate [6]?

Continuous scaling down of CMOS technology is reaching its limiting levels. Further reduction in size has resulted in a number of problems for microelectronics although computing powers have increased at the same time. Some of the problems include interconnection of the circuit at such small scales and also high current leakage or inefficient power dissipation. All these problems require that an alternative technology be found that should replace CMOS technology. One such technology is quantum-dot

ISSN: 2005-4297 IJCA

cellular automata (QCA) [7-9].

In order to construct a random access memory (RAM), a decoder is essentially demanded. Various decoders have been proposed based on QCA. In [10] they proposed a 2:4 decoder with enable lines. This decoder is similar to decoders that are used in typical CMOS memory. In this design they implemented row and column decoders with enable lines. This approach results in large unused area. In [11], they also proposed 3-8 decoder with two modular 2-4 decoder, and they have presented that their structure has been designed with the minimum number of cells. However we reduced many hardware complexities compared to their architecture. In [12], they have proposed two 2-4 decoders that reduced the number of gates and clock phases but they use little more cells compared to the previous architectures

This paper proposes a 3:8 decoder implemented using QCA, this scheme reduces the number of cells in the circuit than the circuit we compared with in [11]. Our architecture uses 5-input majority voting gates for AND operations and inverter chains for taking the reverse value and wire crossing.

The paper is organized as follows; Section 2 gives the related works including QCA basics, an explanation of a decoder and the architecture proposed by Kianpour *et. al.* Section 3 discusses the proposed 3-8 decoder for minimization of hardware complexity, Section 4 gives the simulation and analysis of the results and finally Section 5 gives the conclusions.

2. Related Works

In this section we look at the QCA basics, a brief description of decoders and a previous work by Kianpour *et. al.*

2.1. Quantum-dot Cellular Automata Basics

The basic building block of QCA is a cell. The QCA cell is made-up of four quantum dots. The cell has two electrons that occupy antipodal sites due to Coulombic interaction. The electrons can tunnel around the cell but they cannot tunnel outside the cell. These electrons repel each other due to Coulombic interaction and occupy two opposite sites. This gives the cells stable states. The cells can tunnel within the cell but not outside the cell [13].

In QCA wires can also be formed by arranging cells one after the other in line. In the wire signals propagate from the input to the output and the cells will assume polarization of the input signal. An inverter chain is formed by aligning the quantum dots to be at 45 degrees. The value in an inverter chain changes from ‘1’ to ‘0’ and vice versa as it moves along the wire. An inverter gate inverts an input (*i.e.* ‘1’ to ‘0’ and vice versa) as it moves in the gate. Lastly, a majority voting gate outputs the majority of the three or five input values [14]. Using the majority gate an AND gate and an OR can be implemented.

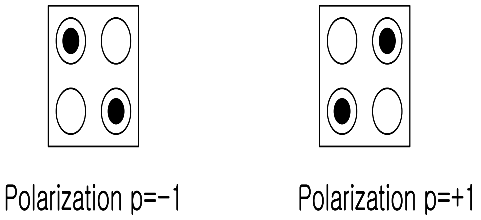


Figure 1. QCA Cells Having Two Ground State Electrons and their Polarization, Where Polarization P=-1 Represents Logical “0” and P=+1 Is Logical ‘1’

QCA structures are formed by placing cells together. A wire can be formed by placing cells one after another as in Figure 2(a). In the wire, information propagation takes the form of the input signal. If the input is a 0, then the information transferred from input to output will be zero or if the input is a one, then one will be transferred to the output [15].

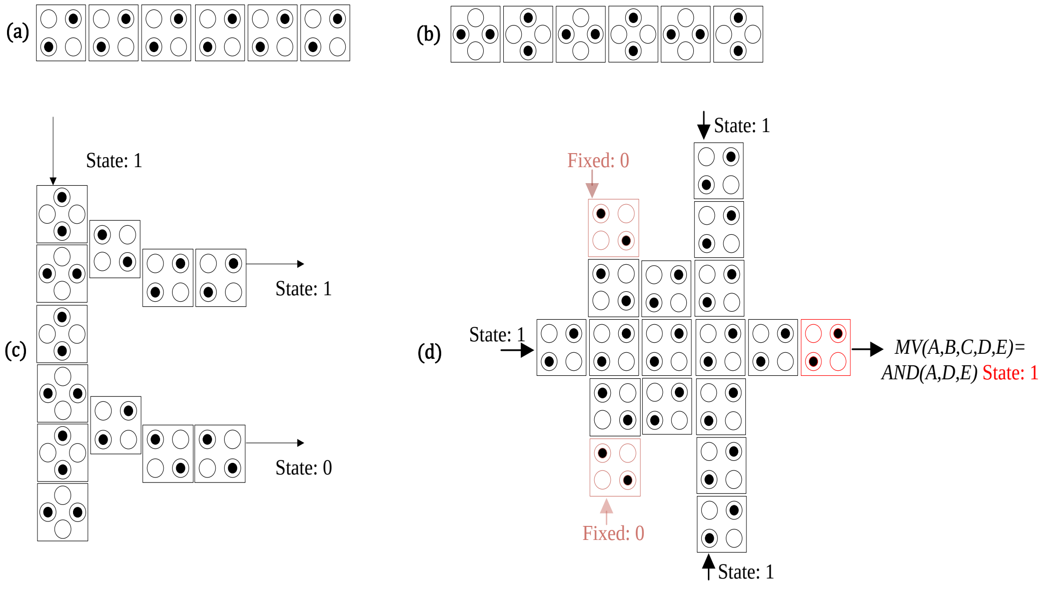
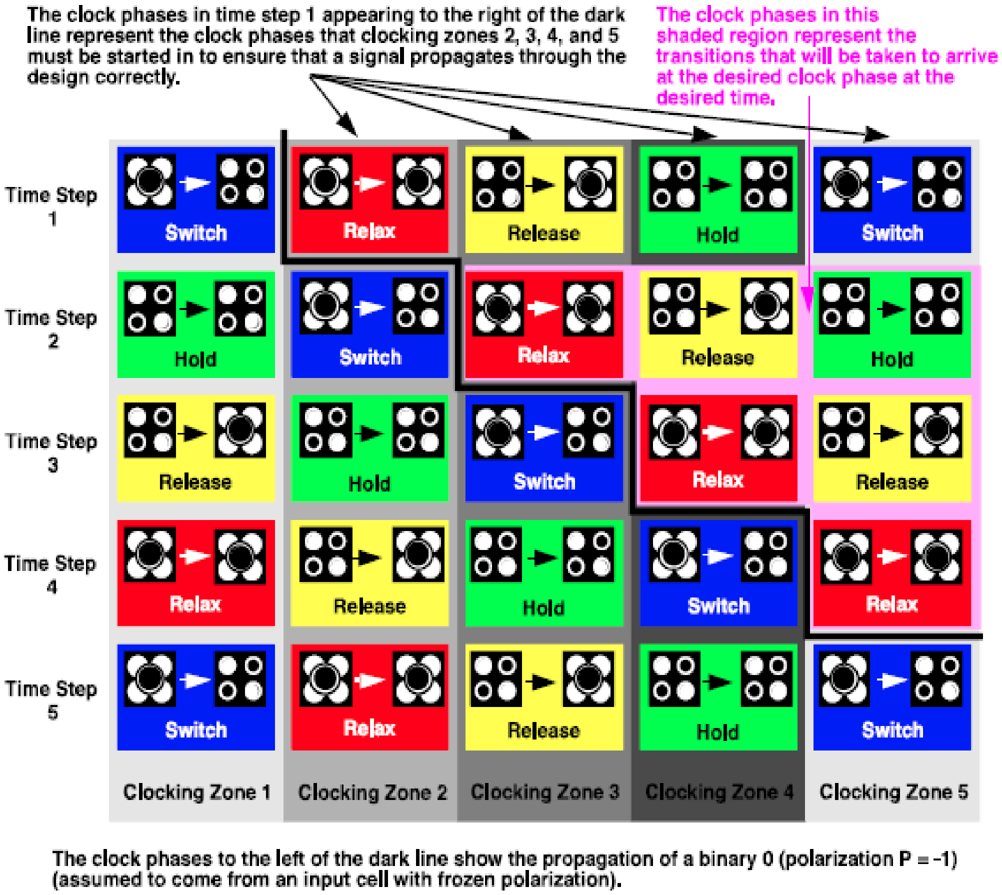


Figure 2. QCA Basic Structures. (A) QCA Wire, (B) Inverter Chain, (C) Inverter Gate, (D) 5-Input Majority Voting Gate Using AND Gate

Figure 2(b) shows an inverter chain. In the inverter chain, the information alternates between 0 and 1 as it is being propagated in the chain and it depends on the initial input signal. The other QCA logic gate is the inverter gate (although there are several versions of inverters which we will not be shown in the paper) shown in Figure 2(c). This inverter is a little different figure compare to others. This is very efficient when you get the inverted value from inverter chain. If you use this structure you can get any value from the inverter chain by transmitting the cell two times. The other important logical gate is the 3-input majority gate and 5-input majority voting gate. Figure 2(d) shows a 5-input majority voting gate fixing two cells as 0 for making an AND gate. The voting gate can be used as AND gate by fixing two input values, *B* and *C* as '0' then the majority voting gate performs the following equation, *i.e.*, $M(A,B,C,D,E) = ADE$.

In QCA, the cells belong to one of the four clock zones and each clock zone has a different clock signal. The clock labeling convention is from time step 1 to 4. Figure 3 shows the four clock zones [16]. The four clock phases are switch, hold, release and relax are utilized in the four clock signals. In the switch phase, the cells begin to compute the value; the cells are unpolarized and have low potential barriers which begin to raise. In the hold phase, the value is held and the barriers are held high. In the release phase, the barriers are lowered and lastly, in the relax phase, the barriers remain lowered where the cells remain unpolarized. The phases allow for a reliable signal transmission and also functional gain which is of paramount importance in computational systems [17-19].



The clock phases in time step 1 appearing to the right of the dark line represent the clock phases that clocking zones 2, 3, 4, and 5 must be started in to ensure that a signal propagates through the design correctly.

The clock phases in this shaded region represent the transitions that will be taken to arrive at the desired clock phase at the desired time.

	Clocking Zone 1	Clocking Zone 2	Clocking Zone 3	Clocking Zone 4	Clocking Zone 5
Time Step 1	Switch	Relax	Release	Hold	Switch
Time Step 2	Hold	Switch	Relax	Release	Hold
Time Step 3	Release	Hold	Switch	Relax	Release
Time Step 4	Relax	Release	Hold	Switch	Relax
Time Step 5	Switch	Relax	Release	Hold	Switch

The clock phases to the left of the dark line show the propagation of a binary 0 (polarization P = -1) (assumed to come from an input cell with frozen polarization).

Figure 3. QCA Clock Phases of Switch, Hold, Release and Relax [16]

There exist three representative wire-crossing techniques: a coplanar-based, a multilayer-based and a crossbar network. Tougaw and Lent proposed a coplanar-based wire-crossing technique as shown in Figure 4 [20]. In this example, vertical and horizontal wires are transmitting the value ‘1’ and ‘0’, respectively. In order to implement this wire-crossing, cells of horizontal wire are rotated by 45′. When the number of cells after an intersection in the vertical wire is sufficient (that is, the number of cells is greater than or equal to 3), a transmitting value is not affected by the other wire.

The number of cells in the horizontal wire should be composed of odd because of the characteristic of rotated cells. 45′ rotated cells induce the additional space between cells. It significantly decreases the energy separation between the ground state and the first excited state, which degrades the performance of such a device in terms of maximum operating temperature, resistance to entropy, and minimum switching time [21]. Moreover, this is much easier to construct and has less noise than multilayered. The horizontal and vertical inputs never get any interruption and get the straight output.

The multilayer-based wire-crossing technique uses a crossover bridge method. This technique is similar to coplanar-based technique in the perspective of the floor plan because it looks like appearance of two wires crossing. The structure of this technique can be more miniaturized and generalized than coplanar-based technique because it does not need to rotate cells. Although this technique has some advantages, there exists the noise problem between intersection cells in crossover area [22].

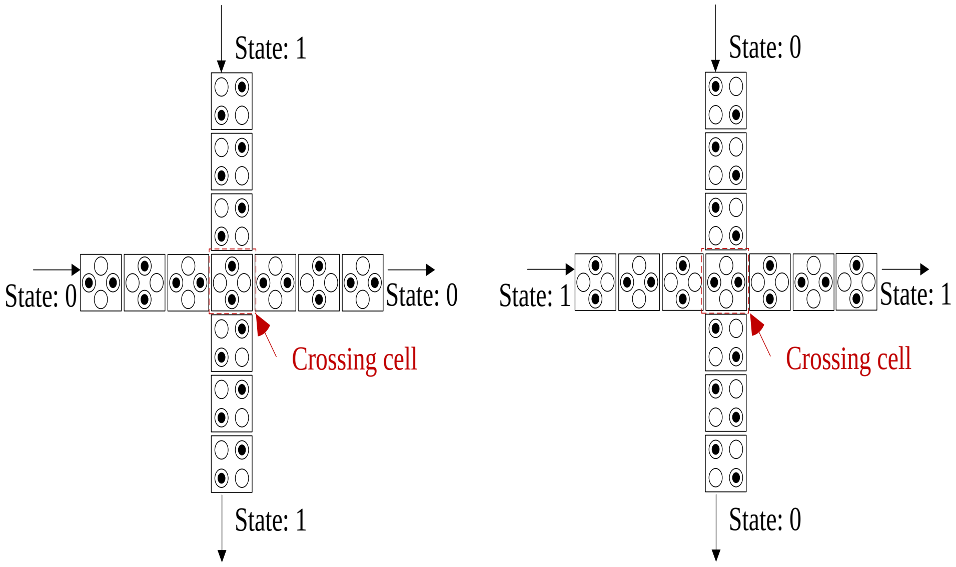


Figure 4. QCA Wire-Crossing Based on One Dimensional Coplanar of 45′ Rotated Cells

2.2. Decoding Architecture

In digital computers, discrete quantities of information are represented by binary codes. An *n*-bit binary code is capable of representing up to 2^n distinct elements of coded information. Decoding is the conversion of an *n*-bit input code to an *m*-bit output code with $n \leq m \leq 2^n$, such that each valid input code word produces a unique output code. Decoding is performed by a decoder, a combinational circuit with an *n*-bit binary code applied to its inputs and an *m*-bit binary code appearing at the outputs [23].

The decoder selects one out of several outputs lines when it has been activated for output. For the 3-input and 8-output decoder, one of the inputs is used to select the output [24]. Many studies have been done on decoders in QCA. They proposed 2-4 or 3-8 decoders using various techniques including a wire-crossing, inverting value and majority voting gate. In [25], they proposed a 5-input majority gate based 2 to 4 decoder, and they reduced the number of gates, delay and both time and hardware complexities. In [11], they proposed a decoder which uses 4 majority gates and also uses 7 clocks. The decoder can be extended to a 3-8 decoder by using two 2-4 decoders.

The truth table of a 3-8 decoding architecture is given in Table 1. There are three variables as an input and each output raw has one output value equal to 1 and seven output values equal to 0.

Table 1. Truth Table of 3-8 Decoder

Input			Output							
A	B	C	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

Figure 5 shows the diagram of the 3-8 decoder based on the truth table shown in Table 1. There are three inverters and eight 3-input AND gates.

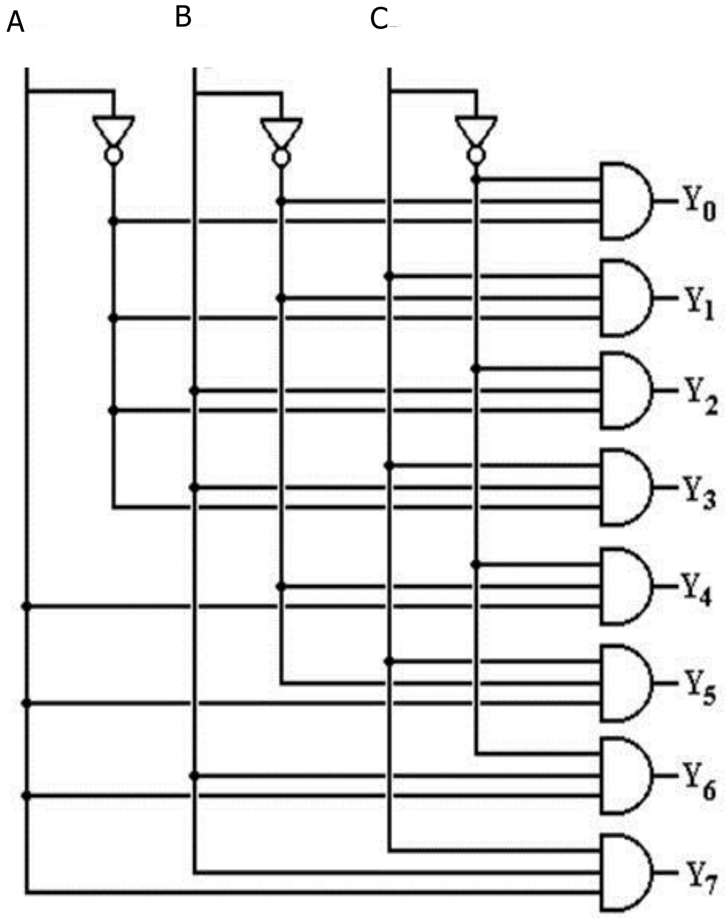


Figure 5. Diagram of 3-8 Decoder

2.3. Previous Works

Kianpuour *et. al.* has proposed a modular decoder using two 2-4 decoder [11]. The 2-4 decoder is implemented by the 5-input majority voting gates and inverter chain as shown in Figure 2. The architecture of a 2-4 decoder required four majority gates and 7 clocks, and they constructed a 3-8 decoder using two small decoders and linked with wiring based on the inverter chain, inverter and wire-crossing technique. Their implementation has a quite good modularity so that easy to extend the architecture to a 4-16 decoder. However, the architecture has many problems including signal strength and propagation, many wasting area and cells.

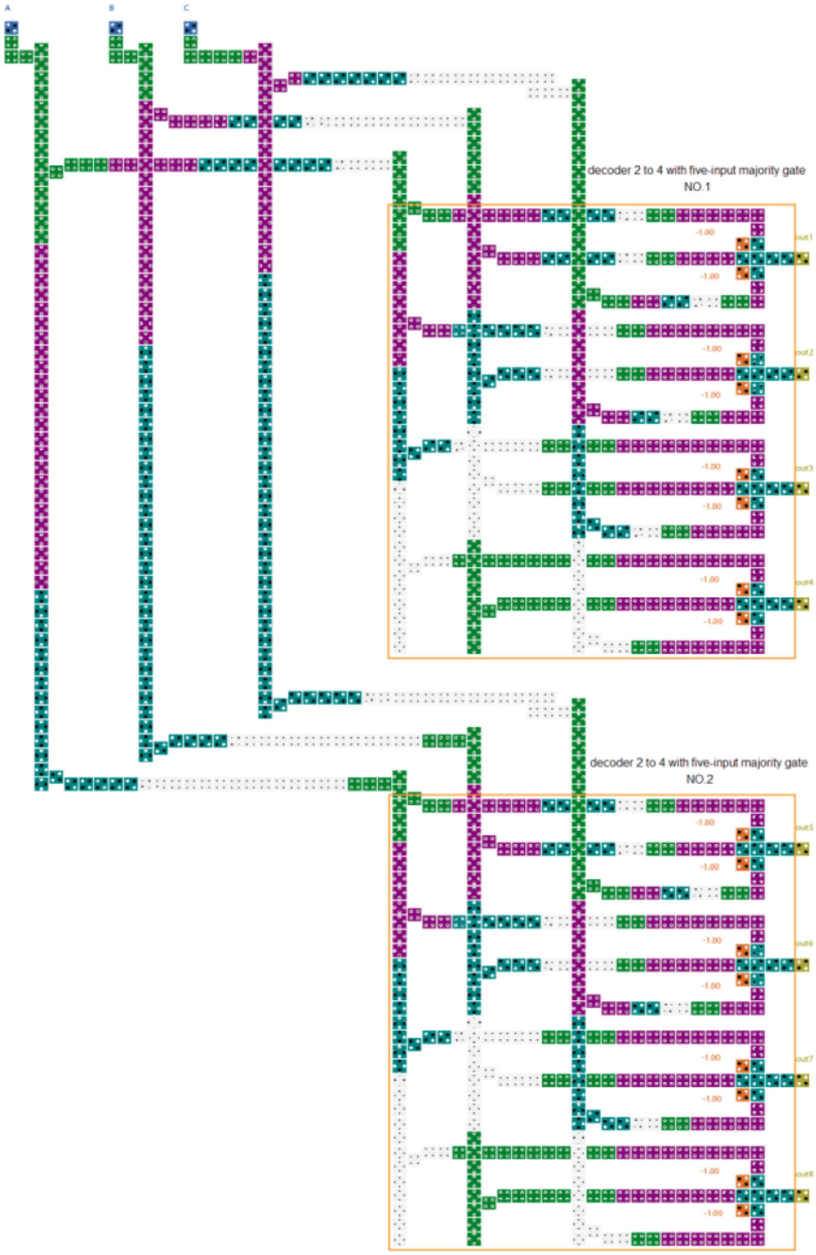

A
B
C
decoder 2 to 4 with five-input majority gate
NO.1
-1.00
out1
out2
out3
out4
decoder 2 to 4 with five-input majority gate
NO.2
out5
out6
out7
out8

Figure 6. 3-8 Decoder by Two Modules 2-4 Decoder by Kianpour

3. Proposed Decoder

We propose a QCA decoder shown in Figure 7. This decoder is made up of eight 5-input majority gates. The gates were implemented as AND gates by fixing two inputs as a '0'. We also adopt a strong signal driven 5-input majority gate proposed in [26] which helps to have a good polarization. We also minimize the wasting area and interval among the chains and wiring, and optimize the interruption among cells.

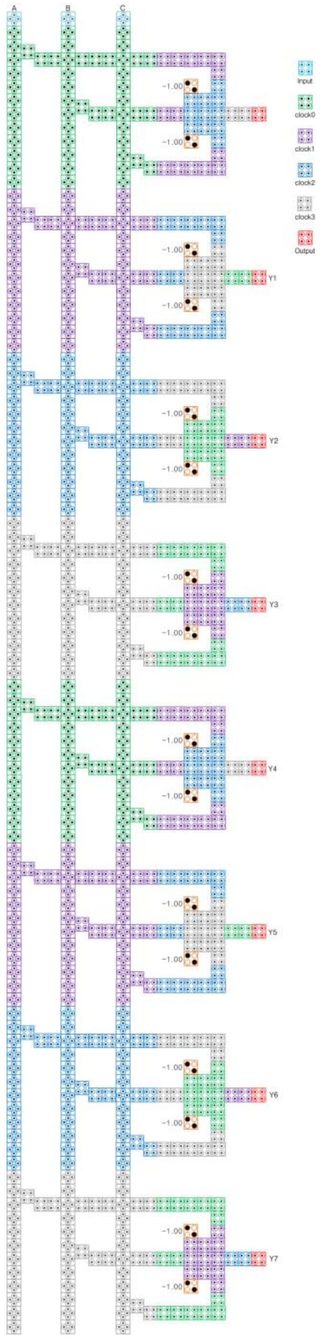


Figure 7. Proposed Structure of the QCA Decoder

It is hard to reduce the delay time since the circuit needs a minimum clock phase and proper wire construction to design but we can minimize the wasting area and cells to get the legitimate results. In this circuit, we only used one layer (not multilayer) hence the use of the inverter chains with rotated cells to help in the signal to propagate properly. Moreover, we do not use inverters but inverter chains with transmitted cells to get inverted values instead of inverters. We also adopt the 5-input majority gate with a strong signal proposed in [26]. Most of all, we designed and displayed cells on very compact and useful so that there is no useless area.

4. Simulation and Analysis

The circuits in this paper were designed and simulated using QCADesigner [27]. The QCADesigner is to perform a design and simulation for QCA, and it is the product of an ongoing research effort by the Walus Group at University of British Columbia [28]. The objective of this tool is to create an easy to use simulation and layout tool available freely to the research community via the Internet. One of the most important design specifications is that other developers should be able to easily integrate their own utilities into this tool. It is accomplished by providing a standardized method of representing information within the software. As well, simulation engines can easily be integrated into QCADesigner using a standardized calling scheme and data types. It is written in C/C++ and employs a wide range of open source software such as GNU image manipulation program toolkit graphics library, and is maintained under the GNU's not Unix public license (GPL) for open source software [29].

The comparison has been done by the number of cells and area covered between the previous circuit proposed in [11]. The comparisons are summarized is Table 2. The outputs are produced after 11clock phases on both architectures. Our circuit demand only 651 cells and 0.77μm^2 for construction of circuit while the previous one demands 1074 cells and 2.24μm^2 area. Thereby we reduced about 40% of cells and 65% of area covered on hardware complexity.

Table 2. Comparison Focused on Hardware Complexity

Comparison parameter	Proposed in [11]	Ours
Number of cells	1074	651
Area covered(μm^2)	2.24	0.77
Clock phase	11	11

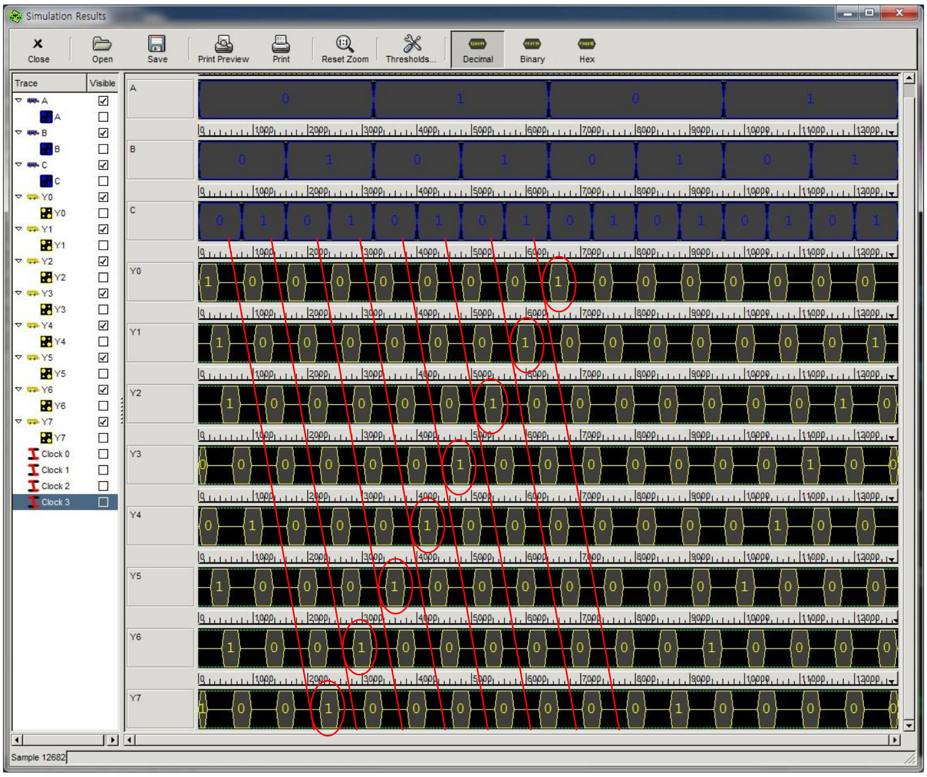


Figure 8. Simulation Result of the Proposed Decoder

The simulation result of our decoder is shown in Figure 8 which has three inputs, *A*, *B* and *C* and eight outputs, *Y*0 to *Y*7 that the outputs are very clear and have a strong signal strength.

5. Conclusions

In this paper, we have proposed a 3-8 decoder with low hardware complexity. We reduced the required cells to design our circuit compared to the previous architecture by minimizing the wasted area and efficient wire-crossing. Moreover, our circuit had strong signal strength and propagation based on the 5-input majority gate which has a strong signal intensity. Our circuit also has an excellent regularity and expandability so that it can easily connect to other circuits and extended for bigger decoders.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NO. NRF-2015R1A2A1A15055749).

References

[1] G. E. Moore, “Cramming more components onto integrated circuits”. Electronics, vol. 38, no. 2 **(1965)**, pp. 114-117.

[2] International technology road map for semiconductors. http://www.itrs.net/, **(2011)**.

[3] K. Makanda and J. C. Jeon, “Combinational Circuit Design Based on Quantum-Dot Cellular Automata”, International Journal of Control and Automation, vol. 7, no. 6, **(2014)**, pp. 369-378.

[4] N. Kumar and V. Kumar, “Gate Engineering on the Analog Performance of DM-DG MOSFETs with High K Dielectrics”, IJAST, vol. 25, **(2010)**, pp. 1-6.

[5] M. Ottavi, S. Pontarelli, V. Vankamamidi, A. Salsano and F. Lombardi, “QCA memory with parallel read/serial write: design and analysis”, IEE Proc.-Circuits Devices Syst, vol. 153, no. 3, **(2006)**, pp. 199-206.

[6] C. S. Lent, M. Liu and Y. Lu, “Bennett clocking of quantum-dot cellular automata and the limits to binary logic scaling”, Nanotechnology, vol. 17, **(2006)**, pp. 4240-4251.

[7] S. Bibhash, D. Manojit and K. S. Biplab, “Efficient design of parity preserving logic in quantum-dot cellular automata targeting enhanced scalability in testing”, Microelectronics Journal, vol. 45, **(2014)**, pp. 239-248.

[8] International Technology Roadmap for Semiconductors, http://www.itrs.net/Links/2011ITRS/ 2011Chapters/ 2011ERD.pdf, **(2011)**.

[9] C. S. Lent, P. D. Tougaw and W. Porod, “Bistable saturation in coupled quantum dots for quantum cellular automata”, Appl. Phys. Lett. vol. 62, **(1993)**, pp. 714-716.

[10] C.C. Tung, R. B. Rungta and E. R. Peskin, “Simulation of a QCA-based CLB and a Multi-CLB Application”, Proceedings of the Field Programmable Technology, Sydney, Australia, **(2009)** December 9-11.

[11] M. Kianpour and R. S. Nadooshan, “A Novel Modular Decoder Implementation In Quantum-dot Cellular Automata (QCA)”, Proceedings of International Conference on Nanoscience, Technology and Societal Implications (NSTSI), **(2011)**, pp. 1-5.

[12] K. Makanda and J. C. Jeon, “Combinational Circuit Design Based on Quantum-Dot Cellular Automata”, International Journal of Control and Automation, vol. 7, no. 6, **(2014)**, pp. 369-378.

[13] J. C. Jeon, “Five-Input Majority Gate Based QCA Decoder”, Proceedings of IRTT2016, **(2016)**, pp. 95-99.

[14] P. D. Tougaw and C. S. Lent, “Logical devices implemented using quantum cellular automata”, Journal of Appl. Phys. vol. 75, no. 3, **(1994)**, pp. 1818-1825.

[15] J. C. Jeon, “Extendable Quantum-Dot Cellular Automata Decoding Architecture Using 5-Input Majority Gate”, International Journal of Control and Automation, vol. 8, no. 12, **(2015)**, pp. 107-118.

[16] M. T. Niemier, “Designing digital systems in Quantum Cellular Automata”, Ph.D. thesis, University of Notre Dame, **(2004)**.

[17] S. W. Kim and E. E. Swartzlander, “Restoring Divider Design for Quantum-Dot Cellular Automata”, Proceedings of IEEE Conference on Nanotechnology, **(2011)**, pp. 1295-1300.

[18] L. R. Hook and S. C. Lee, “Design and Simulation of 2-D 2-Dot Quantum-Dot Cellular Automata Logic”, IEEE Transaction on Nanotechnology, vol. 10, no. 5, **(2011)**, pp. 996-1003.

[19] V. Pudi and K. Sridharan, “Efficient Design of a Hybrid Adder in Quantum-Dot Cellular Automata”, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 19, **(2011)**, pp. 1535-1548.

[20] P. D. Tougaw and C. S. Lent, “Logical devices implemented using quantum cellular automata”, Journal Appl. Phys. vol. 75 **(1994)**, pp. 1818-1825.

[21] C. R. Graunke, D. I. Wheeler, D. Tougaw and J. D. Will, “Implementation of a crossbar network using quantum-dot cellular automata”, IEEE Transactions on Nanotechnology, vol. 4, no. 4, **(2005)**, pp. 435-440.

[22] M. Janez, P. Pecar and M. Mraz, “Layout design of manufacturable quantum-dot cellular automata”, Microelectronics Journal, vol. 43, **(2012)**, pp. 501–513.

[23] M. Mano, C. R. Kime and T. Martin, “Logic and Computer Design Fundamentals 5th edition”, Pearson, England **(2016)**.

[24] A. B. Marcovitz, “Introduction to Logic Design”, McGraw-Hill, New York **(2010)**.

[25] M. Kianpour and R. Sabbaghi-Nadooshan, “A Conventional Design for CLB Implementation of a FPGA in Quantum-dot Cellular Automata (QCA)” proceedings of NANOARCH, **(2012)** pp. 36-42.

[26] R. Akeela and M. D. Wagh, “A Five-input Majority Gate in Quantum-dot Cellular Automata”, proceedings of NSTI-Nanotech, vol. 2, **(2011)**, pp. 13-16.

[27] The Walus Group at the University of British Columbia, QCADesigner **(2007)** http://www.mina.ubc.ca/qcadesigner.

[28] T. G. Team, The gtk+ project, http://www.gtk.org/.

[29] H. Cho, “Adder Designs and Analyses for Quantum-Dot Cellular Automata”, IEEE Transaction on Nanotechnology, vol. 6, no. 3 **(2007)**, pp. 374-383.

Author

Jun-Cheol Jeon is currently a professor in Department of Computer Engineering at Kumoh National Institute of Technology. He received B.S. degree from Kumoh National Institute of Technology in 2000, the M.S. and Ph.D. degrees from Kyungpook National University in 2003 and 2007 respectively. His current research interests are cryptography, cellular automata, quantum-dot cellular automata and quantum computation.