LOGIC-BASED QCA IMPLEMENTATION OF A 4×4 S-BOX

[1]Mohammad Amin Amiri, [1]Sattar Mirzakuchaki, [2]Mojdeh Mahdavi

[1]E. E. Department, Iran University of Science and Technology, Tehran, Iran
[2]Department of Engineering, Science and Research Branch,
Islamic Azad University, Tehran, Iran

Key words: Quantum Cellular Automata; Substitution Box

Abstract: Quantum Cellular Automata (QCA) represents an emerging technology at the nanotechnology level. Nowadays, many applications of QCA technology are introduced and cryptography can be an interesting application of QCA technology. Substitution boxes are important components in many modern day block and stream ciphers. Here, we have implemented a specific 4×4 S-Box using QCA technology. Simulation results are obtained from QCADesigner software.

Izvedba 4x4 S-Box vezja s QCA tehnologijo

Kjučne besede: QCA, S-Box

Izvleček: Tehnologija QCA predstavlja rastočo tehnologijo na nivoju nanotehnologije. Dandanes se poraja veliko aplikacij s QCA tehnologijo kot npr v kriptografiji. V prispevku opišemo izvedbo posebne 4x4 S-Box vezja z uporabo QCA tehnologije. Rezultate simulacij smo pridobili s programsko opremo QCADesigner.

1. Introduction

The microelectronics industry has improved the integration, the power consumption, and the speed of integrated circuits during past several decades by means of reducing the feature size of transistors. But it seems that even by decreasing the transistor sizes, some problems such as power consumption cannot be ignored. Utilizing the QCA technology for implementing logic circuits is one of the approaches which in addition to decreasing the size of logic circuits and increasing the clock frequency of these circuits, reduces the power consumption of these circuits. QCA which was first introduced by Lent et.al /1/ represents an emerging technology at the nanotechnology level. QCA cells have quantum dots, in which the position of electrons will determine the binary levels of 0 and 1.

Substitution provides a significant role in modern cryptography. For some applications, the substitutions are formed by simple Boolean functions (which take several Boolean inputs and give a single output as a result). The design of suitable functions has received significant attention from cryptographers for decades. Substitution is typically implemented by substitution boxes (S-Boxes). These functions have multiple inputs and multiple outputs. Perhaps the most famous S-Boxes are those of the Data Encryption Standard (DES) /2/. Within each round of DES, the most significant contribution to security is made by eight 6-input, 4- output functions. These are specified via lookup tables. The DES algorithm has been subject to a great deal of controversy. Much of this has revolved around the particular substitutions implemented by the eight S-Boxes. The S-Box idea has a firm hold in modern day cryptography. The new international symmetric key cryptography standard, the Advanced Encryption Standard (AES), also uses S-Boxes to perform substitutions /3/. As an application of QCA technology, we have implemented a specific 4×4 S-Box. The method which is used to implement the S-Box is the Logic-Based method. In this method, the S-Box is implemented by logic gates. In the next Section, we will briefly explain the Quantum dot Cellular Automata. It includes the cell introduction, cell-cell coupling, QCA logic, and QCA clocking. In Section III, our work is explained and the simulation results are illustrated.

Simulation results of this implementation are obtained from QCADesigner v2.0.3 software (QCADesigner is developed by the ATIPS lab at the University of Calgary in Canada). QCADesigner v2.0.3 features different simulation engines. Throughout this paper, the coherence vector engine is used due to its accurate and detailed evaluation of QCA.

2. QCA Review

In Quantum Cellular Automata (QCA), a cell contains four quantum dots, as schematically shown in Fig. 1. The quantum dots are shown as the open circles which represent the confining electronic potential. Each cell is occupied by two electrons which are schematically shown as the solid dots.

In a cell, the electrons are allowed to jump between the individual quantum dots by the mechanism of quantum mechanical tunneling but they are not allowed to tunnel between cells. The barriers between cells are assumed sufficient to completely suppress intercellular tunneling.

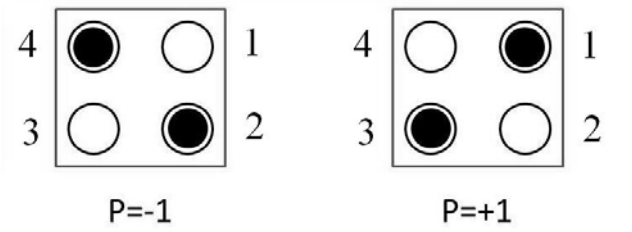

Fig. 1. QCA cell and its ground states

If they are left alone, they will meet the configuration corresponding to the physical ground state of the cell. It is in an obvious manner that the two electrons will tend to occupy different dots because of the Coulombic force associated with bringing them together in close proximity on the same dot.

By these concepts, it's concluded that the ground state of the system will be an equal superposition of the two basic configurations with electrons at opposite corners, as shown in Fig. 1. The positions of the electrons are also shown in this figure.

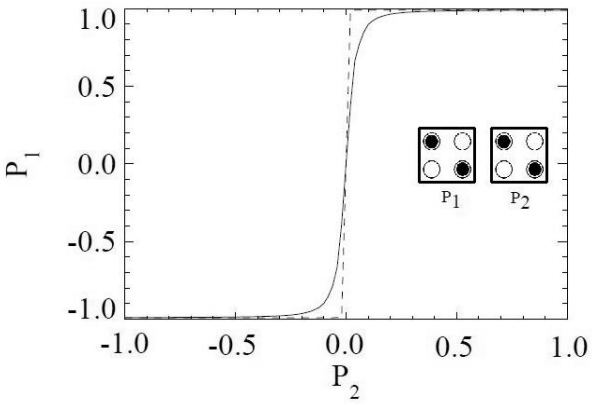

Fig. 2. Coupling of QCA cells

Coupling between the two cells is provided by the Coulomb interaction between electrons in different cells. Fig. 2 shows how one cell is affected by the state of its neighbor /4/. This figure shows the two cells where the polarization of cell 1 (P1) is determined by the polarization of its neighbor (P2). P2 is assumed to be fixed at a given value, corresponding to a specific arrangement of charges in cell 2 and this charge distribution exerts its influence on cell 1, thus determining its polarization. The result which can be drawn here is the strongly non-linear nature of the cell-cell coupling. Cell 1 is almost completely polarized even though cell 2 might only be partially and not completely polarized /3,4/.

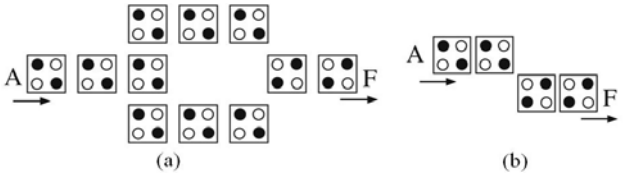

Fig. 3. (a) Redundant inverter gate, (b) Inverter gate

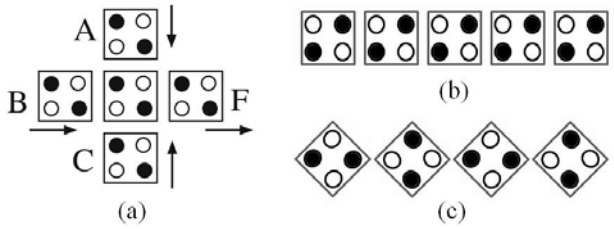

Fig. 4. (a) Majority logic gate, (b) Binary wire, (c) Inverter chain

The physical interactions between cells may be used to realize elementary Boolean logic functions. The basic logic gates in QCA are the Majority logic function and the Inverter which are illustrated in Fig. 4(a) and Fig. 3, respectively. The Majority logic function can be realized by only 5 QCA cells /5/.

The logic AND function can be implemented by a Majority logic function by setting one of its inputs permanently to 0 and the logic OR function can be implemented by a Majority logic function by setting one of its inputs permanently to 1.

QCA clocking provides a mechanism for synchronizing information flow through the circuit. It should be considered that the clock also controls the direction of information flow in a QCA circuit. The QCA clock also provides the power required for circuit operation. More precisely, the QCA clock is used to control the tunneling barrier height in cells. When the clock is low, the electrons are trapped in their associated positions and can't tunnel to other dots, therefore latching the cell (Hold phase). This is caused by the intracellular barriers which are held at their maximum height. When the clock signal is high, the cell goes to the null polarization state (Relax phase). This is caused by the intracellular barriers which are held at their minimum height. Between these two cases, the cells are either releasing or switching.

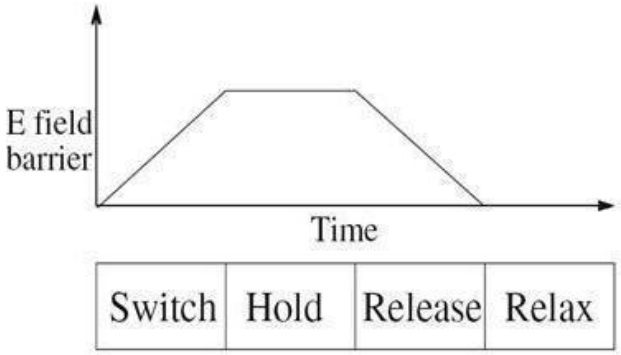

Fig. 5. Barrier height in four phases of clock

Fig. 5 shows the barrier height in four phases of clock. Each cell in a particular clocking zone is connected to one of the four available phases of the QCA clock shown in Fig. 6. Each cell in the zone is latched and unlatched in synchronization with the changing clock signal and therefore the information is propagated through cells /6-9/.

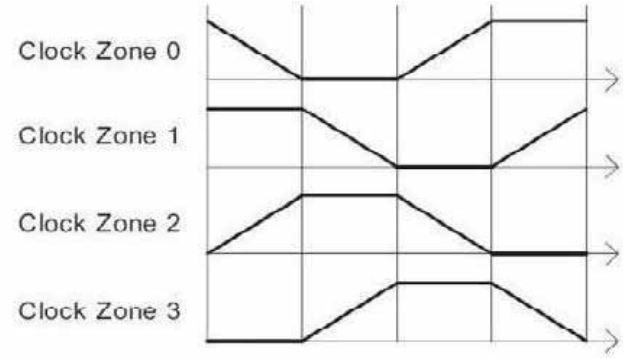

Fig. 6. QCA clock zones

3. Logic-Based Implementation of S-Box

At first, we have chosen a specific substitution function to implement. This 4×4 substitution function is illustrated in TABLE I. Input and output values are shown in hexadecimal format.

Table 1 The Input and Output of S-Box

S-Box Input				S-Box Output			
S3	S2	S1	S0	O3	O2	O1	O0
0	0	0	0	0	0	1	1
0	0	0	1	1	0	0	0
0	0	1	0	1	1	1	1
0	0	1	1	0	0	0	1
0	1	0	0	1	0	1	0
0	1	0	1	0	1	1	0
0	1	1	0	0	1	0	1
0	1	1	1	1	0	1	1
1	0	0	0	1	1	1	0
1	0	0	1	1	1	0	1
1	0	1	0	0	1	0	0
1	0	1	1	0	0	1	0
1	1	0	0	0	1	1	1
1	1	0	1	0	0	0	0
1	1	1	0	1	0	0	1
1	1	1	1	1	1	0	0

The name of S-Box refers to the length of its input and output. Here, the 4×4 S-Box means that the input and output of this S-Box have the length of 4 bits. With naming the input bits as A, B, C, D, and output as O3, O2, O1 and O0, the following logic functions are extracted:

$$O3 = \overline{B}\overline{C}D + \overline{A}\overline{B}C\overline{D} + \overline{A}B\overline{C}\overline{D} + BCD + ABC + A\overline{B}\overline{C}$$
$$O2 = \overline{A}C\overline{D} + \overline{A}B\overline{C}D + ABCD + A\overline{C}\overline{D} + A\overline{B}\overline{C} + A\overline{B}D$$

$$O1 = \overline{C}\overline{D} + \overline{A}BD + \overline{A}\overline{B}\overline{D} + A\overline{B}CD$$
$$O0 = \overline{A}\overline{B}\overline{D} + AB\overline{D} + \overline{A}C + A\overline{B}\overline{C}D$$

Considering the O3 output, it is implemented using QCA cells and is illustrated in Fig. 7. The inputs are applied to the circuit through binary wires. Each term of logic func-

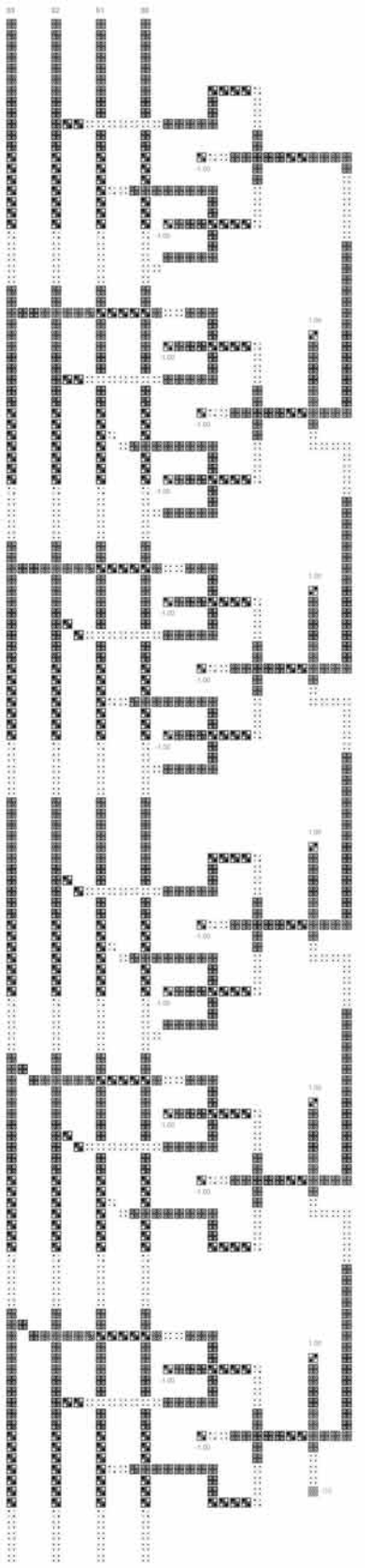

Fig. 7. O3 output implementation

tion is composed of two or three majority gates which are used as logic AND functions. Two majority gates are for the term which contains only three inputs and three majority gates are for terms which contain four inputs. The outputs of AND functions are then logically ORed to result the O3 output. An exhaustive simulation is accomplished for O3 output and simulation result of the O3 output is shown in Fig. 8. As illustrated, the O3 output is valid after eight clock cycles. The "0110100111000011" pattern in O3 which corresponds to the values of inputs from 0 to F, can be seen in Fig. 8.

Fig. 8. O3 output simulation result

Considering the O2 output, it is also implemented using QCA cells and is illustrated in Fig. 9. The inputs are applied to the circuit through binary wires. Each term of logic function is composed of two or three majority gates which are used as logic AND functions. Two majority gate is for the term which contains only three inputs and three majority gates are for terms which contain four inputs. The outputs of AND functions are then logically ORed to result the O2 output.

An exhaustive simulation is also accomplished for O2 output and simulation result of the O2 output is shown in Fig. 10. As illustrated, the O2 output is valid after eight clock cycles. The "0010011011101001" pattern in O2 which corresponds to the values of inputs from 0 to F, can be seen in Fig. 10.

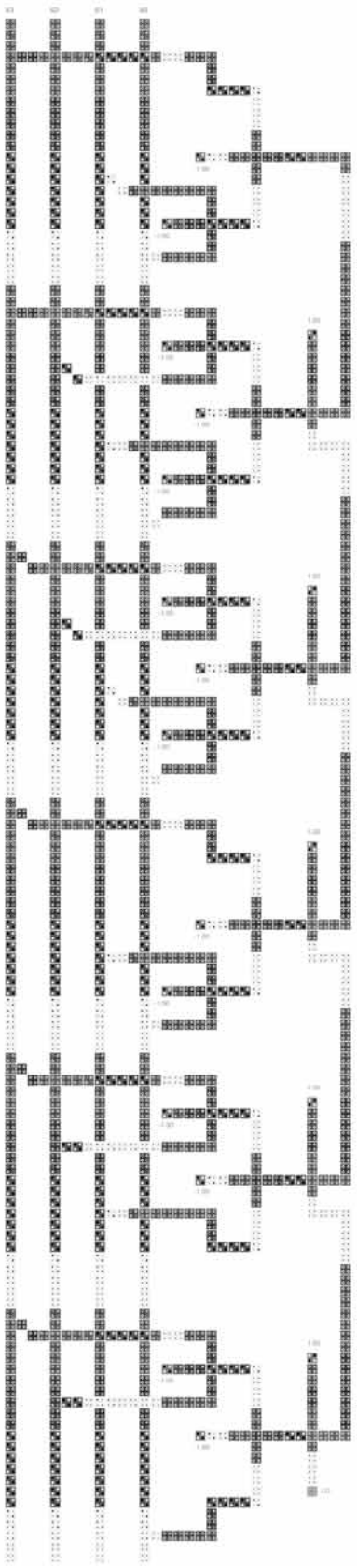

Fig. 9. O2 output implementation

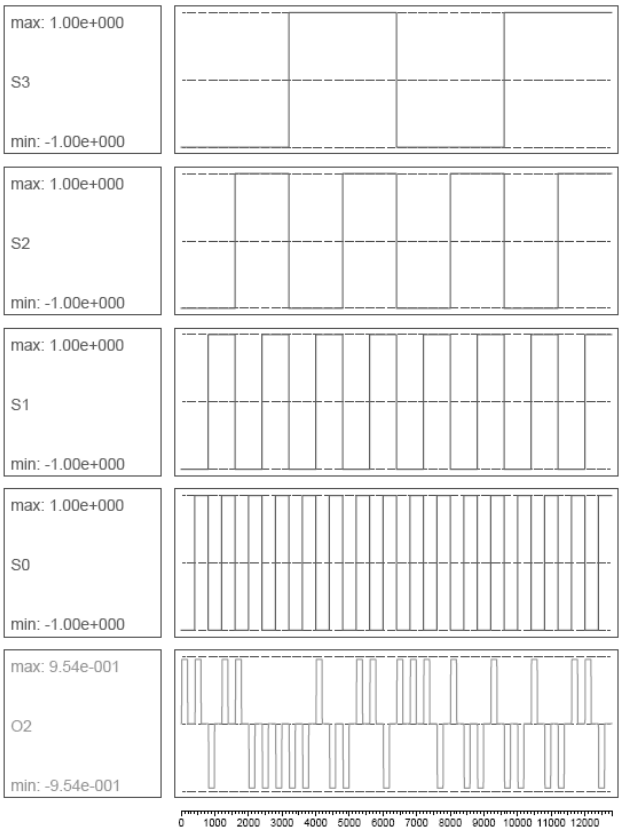

Fig. 10. O2 output simulation result

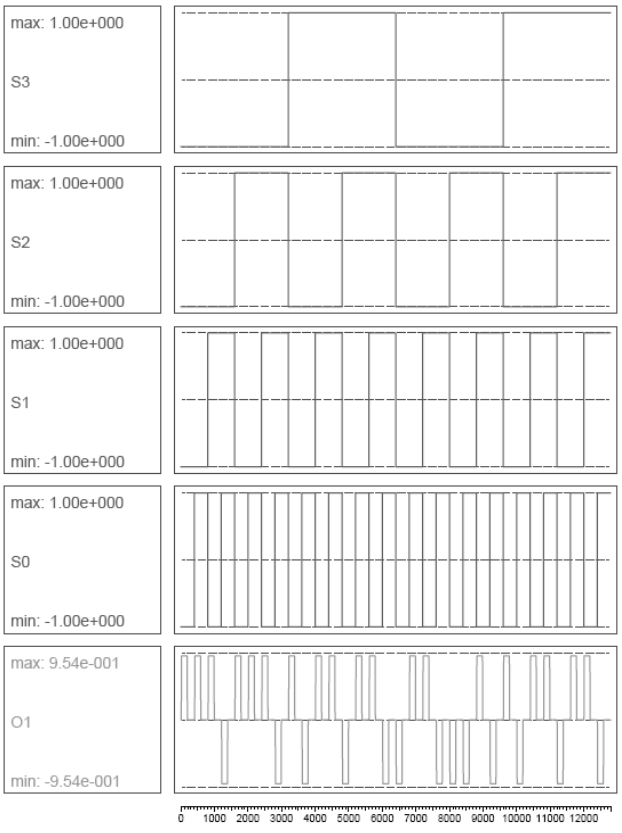

Fig. 12. O1 output simulation result

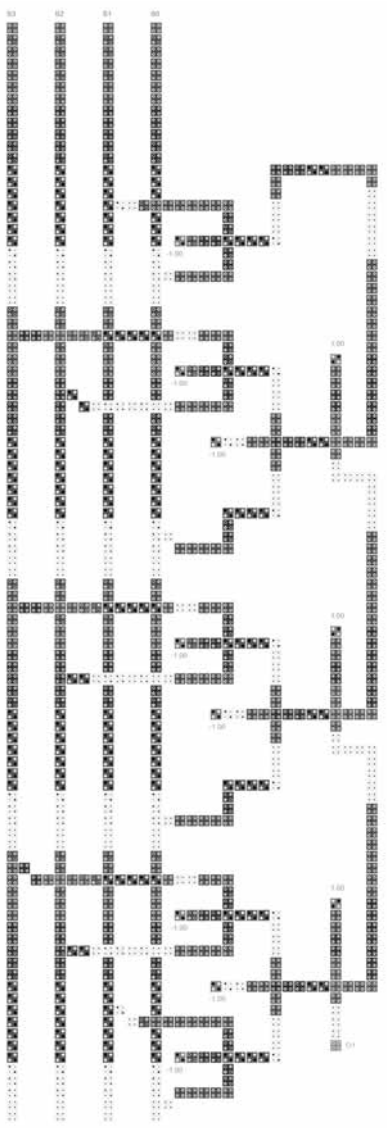

Fig. 11. O1 output implementation

Considering the O1 output, it is also implemented using QCA cells and is illustrated in Fig. 11. The inputs are applied to the circuit through binary wires. Each term of logic function is composed of one or two or three majority gates which are used as logic AND functions. One majority gate is for the term which contains only two inputs and two majority gates are for terms which contain three inputs and three majority gates are for terms which contain four inputs. The outputs of AND functions are then logically ORed to result the O1 output.

An exhaustive simulation is also accomplished for O1 output and simulation result of the O1 output is shown in fig. 12. As illustrated, the O1 output is valid after six clock cycles. The "1010110110011000" pattern in O1 which corresponds to the values of inputs from 0 to F, can be seen in Fig. 12.

Considering the O0 output, it is also implemented using QCA cells and is illustrated in Fig. 13. The inputs are applied to the circuit through binary wires. Each term of logic

function is composed of one or two or three majority gates which are used as logic AND functions. One majority gate is for the term which contains only two inputs and two majority gates are for terms which contain three inputs and three majority gates are for terms which contain four inputs. The outputs of AND functions are then logically ORed to result the O0 output.

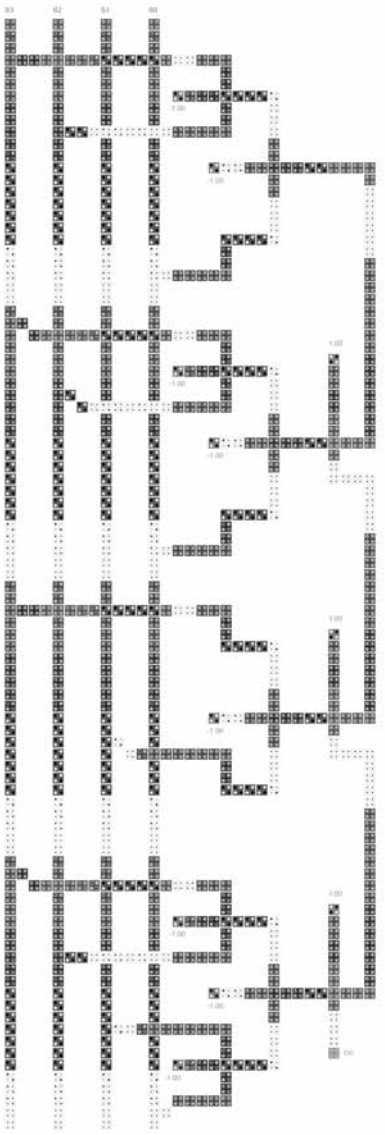

Fig. 13. O0 output implementation

An exhaustive simulation is also accomplished for O0 output and simulation result of the O0 output is shown in Fig. 14. As illustrated, the O0 output is valid after six clock cycles. The "1011001101001010" pattern in O0 which corresponds to the values of inputs from 0 to F, can be seen in Fig. 14.

The implementation results are illustrated in TABLE II. The Complexity, Area and Delay of this implementation are illustrated. The maximum Delay among four output bits is considered to be the Delay of this S-Box. Like previous works such as /10, 11/ each QCA cell is assumed to have

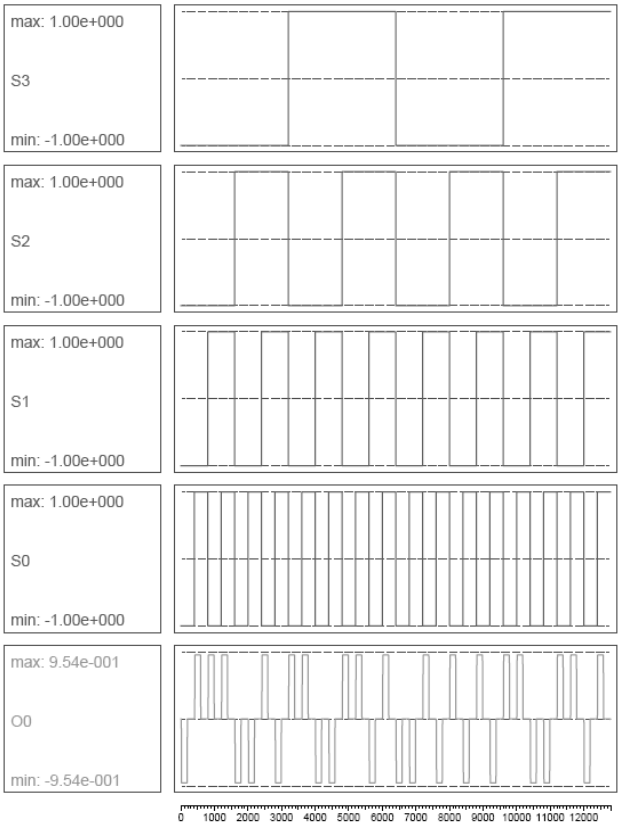

Fig. 14. O0 output simulation result

the width and length of 18 nm. The neighbor cells have a center to center distance of 20 nm.

Table 2 Implementation Results of S-Box

	O3	O2	O1	O0	S0 S-Box
Complexity(Cells)	1204	1205	758	798	3965
Area(μm^2)	1.7236	1.7236	1.1532	1.1532	5.7536
Delay(Clocks)	8	8	6	6	8

4. Conclusion

We have implemented a specific 4×4 S-Box using QCA technology. This S-Box has four bits of input and four bits of output. Every output is implemented and exhaustively simulated. Simulation results of all outputs are illustrated in Fig.8, Fig.10, Fig.12 and Fig.14.

Any type of S-Box with any length of input and output can be implemented and simulated in such a routine, even though there are some other methods for implementation as well.

References

/1/ C. S. Lent, P. D. Tougaw, W. Porod, G. H. Bernstein, "Quantum Cellular Automata," Nanotechnology, vol. 4, no. 1, 1993, pp. 49–57.

/2/ National Bureau of Standards, "Data Encryption Standard", NBS FIPS PUB 46, 1976.

/3/ John A. Clark, Jeremy L. Jacob, Susan Stepney, "The Design of S-Boxes by Simulated Annealing," International Conf. on Evolutionary Computation, Portland OR, USA, pages 1533-1537, IEEE 2004.

/4/ P. D. Tougaw, C. S. Lent, "Dynamic Behavior of Quantum Cellular Automata," J. Appl. Phys.,vol. 80, no. 8, October 1996, pp. 4722–4735.

/5/ P. D. Tougaw, C. S. Lent, and W. Porod, "Bistable Saturation in Coupled Quantum-dot Cells," J. Appl. Phys., vol. 74, no. 5, Sep. 1993, pp. 3558–3565.

/6/ P.D. Tougaw and C.S. Lent, "Logical Devices Implemented Using Quantum Cellular Automata," J. Appl. Phys., vol. 75(3) , 1994, pp. 1818-1825.

/7/ K. Hennessy and C. S. Lent, "Clocking of Molecular Quantum-dot Cellular Automata," J. Vac. Sci. Technol., vol. 19, no. 5, Sep. 2001, pp. 1752–1755.

/8/ C. S. Lent and Beth Isaksen, "Clocked Molecular Quantum-dot Cellular Automata," IEEE Trans. on Electron Devices, vol. 50, no. 9, Sep. 2003.

/9/ M. A. Amiri, M. Mahdavi, S. Mirzakuchaki, "QCA Implementation of a Mux-Based FPGA CLB," Proc. of International Conf. On Nanoscience and Nanotechnology, Australia, Feb. 2008, pp. 141-144.

/10/ Heumpil Cho, Earl E. Swartzlander, Adder Designs and Analyses for Quantum-Dot Cellular Automata, IEEE Trans. on Nanotechnology, vol. 6, n. 3, May 2007, pp. 374–383.

/11/ Heumpil Cho, Earl E. Swartzlander, Adder and Multiplier Design in Quantum-Dot Cellular Automata, IEEE Trans. on Computers, vol. 58, n. 6, June 2009, pp. 721–727.

Mohammad Amin Amiri
E. E. Department, Iran University of Science and Technology, Tehran, Iran
amiri@ee.iust.ac.ir

Sattar Mirzakuchaki
E. E. Department, Iran University of Science and Technology, Tehran, Iran
m_kuchaki@iust.ac.ir

Mojdeh Mahdavi
Department of Engineering, Science and Research Branch, Islamic Azad University Tehran, Iran
m.mahdavi@ieee.org

Prispelo (Arrived): 24.11.2009 *Sprejeto (Accepted): 09.09.2010*