Int. J. Simul. Multisci. Des. Optim. 2014, **5**, A04
© F. Ahmad et al., Published by EDP Sciences, 2014
DOI: 10.1051/smdo/2013012

IJSMDO

Available online at:
www.ijsmdo.org

ARTICLE OPEN ACCESS

A novel idea of pseudo-code generator in quantum-dot cellular automata (QCA)

Firdous Ahmad[1], M. Mustafa[1], Nisar Ahmad Wani[1], and Feroz A Mir[2,*]

[1] Department of Electronics & Instrumentation Technology, University of Kashmir, Srinagar 190006, India
[2] University Science Instrumentation Centre (USIC), University of Kashmir, Srinagar 190006, India

Received 23 June 2013 / Accepted 23 October 2013 / Published online 4 February 2014

Abstract – In present work, we have implemented the pseudo-code (PN-code) generator using quantum-dot cellular (QCA) technology. Simulation results are obtained from QCA designer software. The PN-code generation is of paramount importance for any secure communication system. The complex code generated is used to scramble incoming plain text. At the receiving end, the same code is generated and successfully used to decrypt the transmitted data. The algorithm for generating noise signal is quite simple. The simplicity of the circuit along with the complexity of the code generated makes the circuit attractive for secure message communication.

Key words: Generator, Communication, Noise, Technology, Circuit, Algorithm.

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 various semiconducting components. 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 approach 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. As an application of QCA technology, we have implemented the pseudo-code generator. In the next Section, we have explained the quantum dot cellular automata. It includes the cell introduction, cell-cell coupling, QCA logic, and QCA clocking. Section 3 describes the pseudo-code generator. In Section 4, our work is explained and the simulation results are illustrated. Section 5 concludes the paper. 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 Bistable approximation engine is used due to its accurate and detailed evaluation of QCA.

2 Quantum-dot cellular automata

Fundamental unit of QCA device is QCA cell. QCA cell essential consists of four quantum dots arranged in square pattern coupled by tunneling barriers as shown in Figure 1a. Quantum dots are nano meter sized structures that is capable of trapping electrons in three dimensions. Due to Coulombic repulsion, the two electrons reside in opposite corners representing two polarizations [2]. Here we associate $P = -1$ represents binary "0", $P = +1$ represents binary "1". Figures 1a and 1b shows how one cell is affected by the state of its neighbor. It shows how the polarization of cell 2 (P2) is determined by the polarization of its neighbor (P1). P1 is assumed to be fixed at a given value, corresponding to a specific arrangement of charges in cell 1 and this charge distribution exerts its influence on cell 2, and hence determining its polarization. The results which can be drawn here is the strongly non-linear nature of the cell-cell coupling. Cell 2 is almost completely polarized even though cell 1 might be partially and not completely polarized [5, 6].

2.1 QCA logic

Some basic elements for QCA logic implementation are wire, inverter, and majority voter [3] shown in Figures 2a–2d.

*e-mail: famirnit@gmail.com

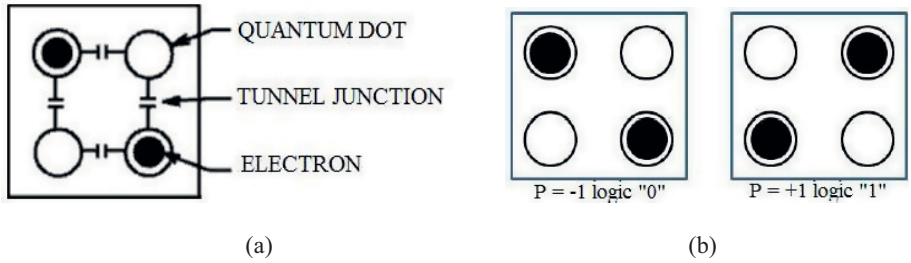


Figure 1. (a) Basic QCA cell; (b) Cell polarization.

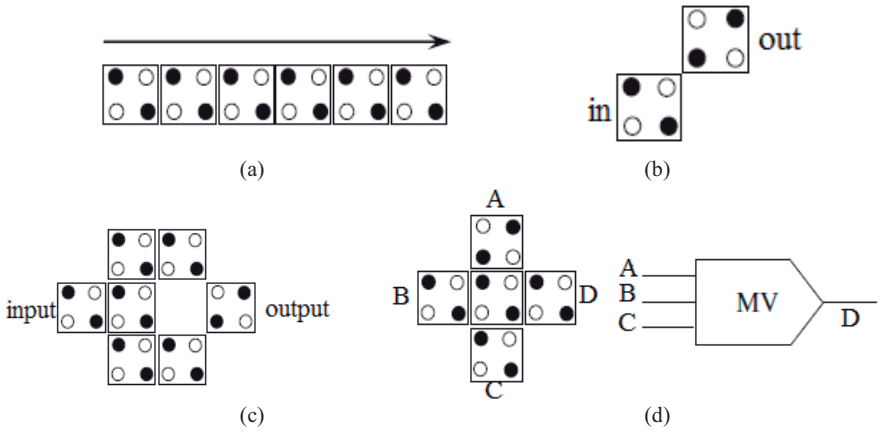


Figure 2. Basic QCA logic devices: (a) wire; (b) and (c) inverter (d) majority vote.

The QCA wire is formed by an array of QCA cells shown in Figure 2a, which provides a medium for data propagation based on Coulomb interactions. The simplest inverter is built by placing QCA cells in a diagonal structure shown in Figure 2b; the common inverter is built by seven cells shown in Figure 2c. The polarization of the output QCA cell "out" or "output" is the opposite of the polarization of input QCA cell "in" or "input". QCA majority voter (MV) and its logic symbol are shown in Figure 2d. Here the MV, is equivalent to a logic function F (A, B, C) = AB + AC + BC and can be implemented by five QCA cells arranged in a cross. Cells A, B, and C are input cells, and cell D is the output cell that is polarized according to the polarization of majority of the input cells. Logical AND and OR functions can be implemented from majority voter by presetting one input immutably to binary values 0 and 1, respectively.

2.2 QCA clocking

A QCA cell has four clock zones and clock zone has four phases; switch, hold, release and relax [4]. Figure 3a shows its operation process. During the switch phase, QCA cells begin unpolarized and their interdot potential barriers are low. The barriers are then raised during this phase and the QCA cells become polarized according to the state of their driver (i.e., their input cell). It is in this clock phase that the actual computation (or switching) occurs. By the end of this clock phase, barriers are high enough to suppress any electron tunneling and cell states are fixed. During the hold phase, barriers are held high so the outputs of the sub array can be used as inputs to the next stage. In the release phase, barriers are lowered and cells are allowed to relax to an unpolarized state. Finally, during the fourth clock phase, the relax phase, cell barriers remain lowered and cells remain in an unpolarized state [2, 4]. In the meantime, the large scale QCA circuit is partitioned into four clock zones; Figure 3b shows each clock zone signal and demonstrates pipeline mechanism. All cells in a certain zone are controlled by the same QCA clock signal. Cells in each zone perform a specific calculation; the state of a zone is then fixed so that it can serve as input signals to the next zone. Information transfers in a pipelined fashion.

3 Description of pseudo-code generator

A pseudo-code generator is a periodic binary sequence that is usually generated by means of a linear feedback shift registers (LFSR). Which is used everywhere in secure message communication systems for encryption whether using active attacks or passive attacks (these are attacks on message signal include: jamming or unauthorized reception). A various techniques can be used to encrypt the message signals. Spread spectrum modulation technique and cryptography are used as a primary counter against jamming or unauthorized reception respectively. A key device in a spread spectrum system is a pseudo-code generator, which is used to generate a modulated signal. It is usually implemented by means of a linear feedback shift register (LFSR). Cryptography is the science by of making communication unintelligible to everyone except the intended receivers.

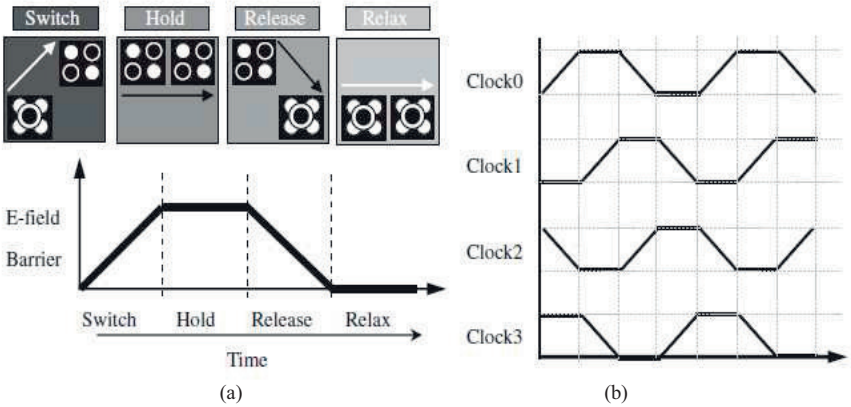


Figure 3. (a) Four phases of QCA clock; (b) Clock zones signal.

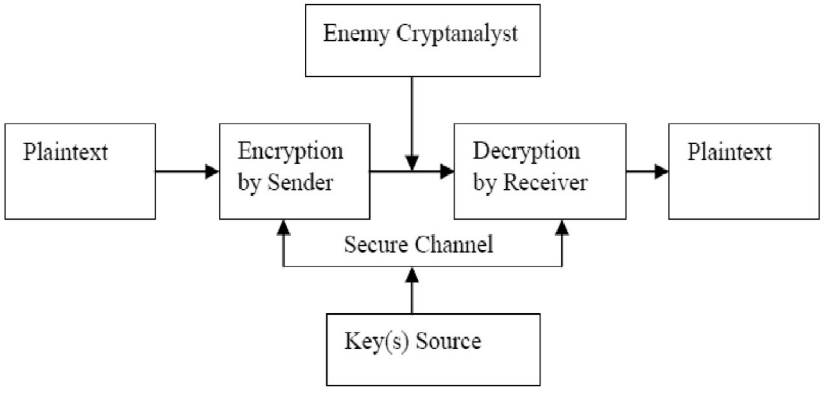


Figure 4. Shannon model of secret communication.

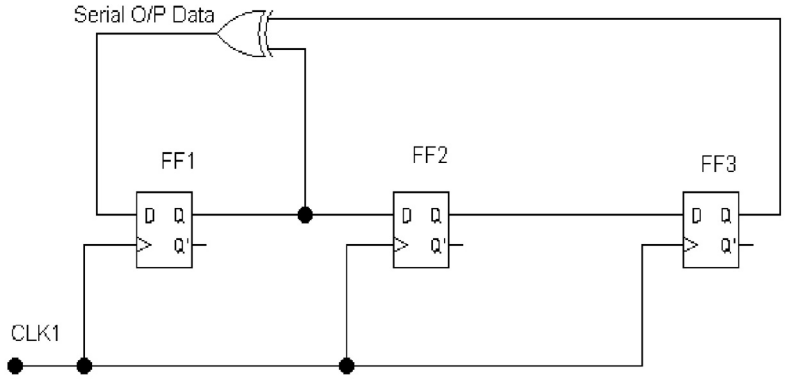


Figure 5. Linear feedback shift register.

Shannon shown in Figure 4 first proposed the model for the secret key.

LFSR counter can be a new trendsetter in cryptography, and is also beneficial as compared to GRAY & BINARY counter and variety of other applications. LFSR is a shift register whose input bit is a linear function unlike most everyday devices whose inputs and operations are effectively predefined, It is a shift register that, when clocked moves the signal through the register from one flip flop to next. Some of the outputs are combined in exclusive-OR configuration to form a feedback mechanism. A LFSR can be formed by performing exclusive-OR on the outputs of two or more of the flip-flops together and feeding those outputs back into the input of one of the flip flops as shown in Figure 5.

The length of periodic binary sequence is given by formula

$$L = \left(2^k - 1\right), \tag{1}$$

where k = the number of D-flip-flops used in LFSR. Here $k = 3$.

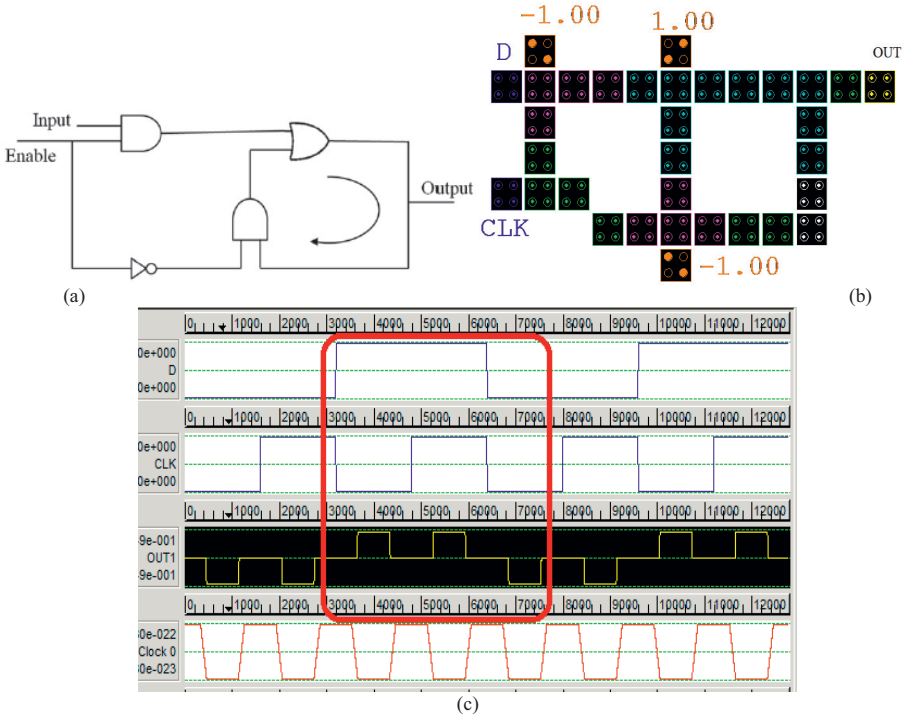


Figure 6. (a) D-latch circuit; (b) Layout design of D-latch; (c) Simulation result of D-latch.

Table 1. Periodic sequence of LFSR.

OUT 1	0	0	1	0	1	1	1
OUT 2	1	0	0	1	0	1	1
OUT 3	1	1	0	0	1	0	1

All the flip-flops are loaded with any bit pattern sequence except (0000), which will circulate "0" throughout the whole shift register because $0 + 0 = 0$ by performing modulo 2-addition.

The bit pattern to be loaded can be anyone of the following 001, 010, 011, etc. let us assume that initial bit pattern will be 001, *Period of sequence* $= 2^3 - 1 = 7$.

Sequence will repeat itself seven times as shown in Table 1.

4 Results and discussions: implementation of pseudo-code generator

In this section, the implementation of pseudo-code generator is investigated by means of implementation and simulation of its main blocks. One of its main blocks are D-latch and XOR gate. We have implemented serial-in serial-out shift register (SISO). One of the candidate structures that are likely to be used in a pseudo-code generator is D-latch.

4.1 D-Latch

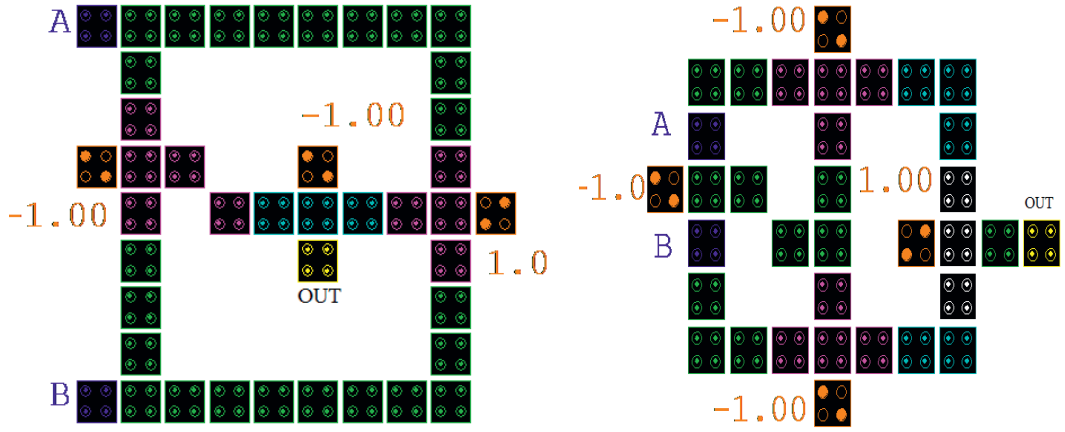
Flip-flops are also very important sequential circuits in quantum-dot cellular automata (QCA) because they are expected to be used for designing and realizing large scale sequential circuits, for example counters, shift registers. Anteriority, some works about sequential circuits design has been published, like QCA R-S flip-flop [11–13], these flip-flop takes into account the timing issues associated with the adiabatic switching of the technology and its requirements. But to our knowledge, QCA flip-flop and sequential circuit designs have not been widely studied. In addition to R-S flip-flop a new method of falling edge-triggered flip-flop and counter study have been proposed by different authors [14]

In the structures of D-latch that has been proposed in [7], the cell value is kept through a closed loop. This D-latch schematic structure is shown in Figure 6a. It is clear that the input will be in effective, and the values to red in the loop will not change if Enable input is "0". However, if Enable input is set to "1", the Input value will be conducted into the loop, i.e., the value stored in the loop will be changed to Input polarity. The D-latches can be used to constructed shift registers. The layout design of schematic D-latch structure is shown in Figure 6b.

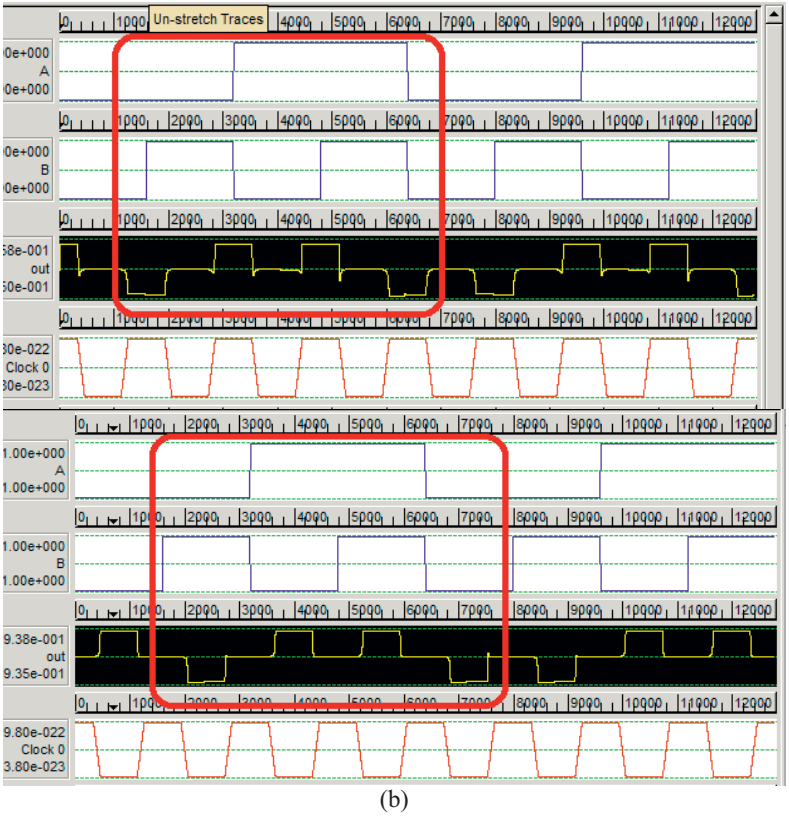
The simulation result of layout design of D-latch structure is shown in Figure 6c. The simulation has been verified and checked using QCADesigner software.

4.2 XOR gate

To realize more complicated logical functions, a division of simple logical gates is vital. The XOR gate is one of the most applicable digital gates. In comparators, error detection in the data link layer in OSI and TCP/IP in the network, designing bus in microprocessors and other various applications have been utilized. The XOR is a logical function on two operands that results in a logical value of true if and only if one of the

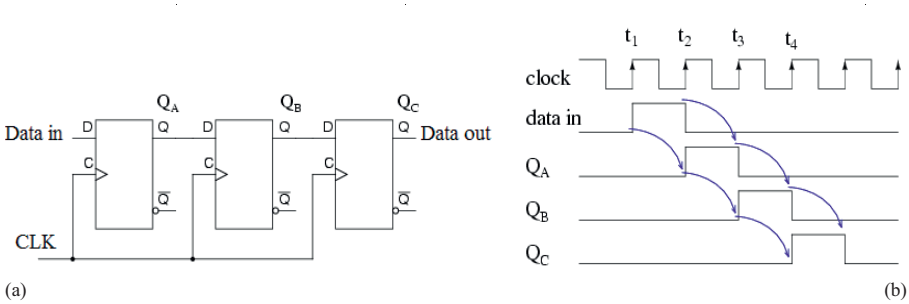


(a)



(b)

Figure 7. (a) The proposed layouts of XOR gates; (b) Simulation results of proposed designs.



(a) (b)

Figure 8. (a) Schematic of SISO register; (b) Timing diagram.

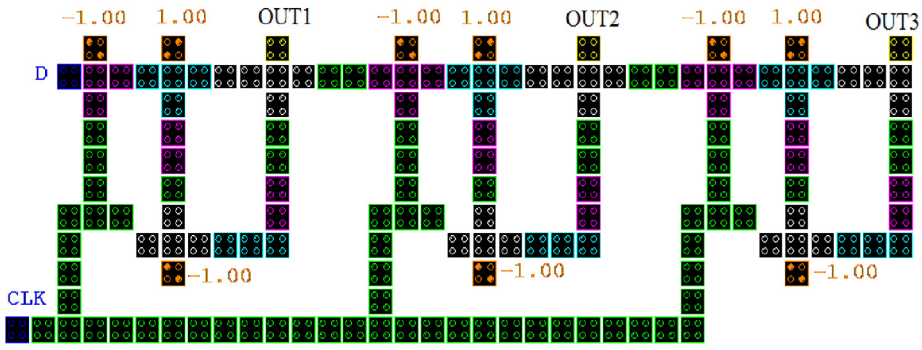


Figure 9. Layout design of SISO register.

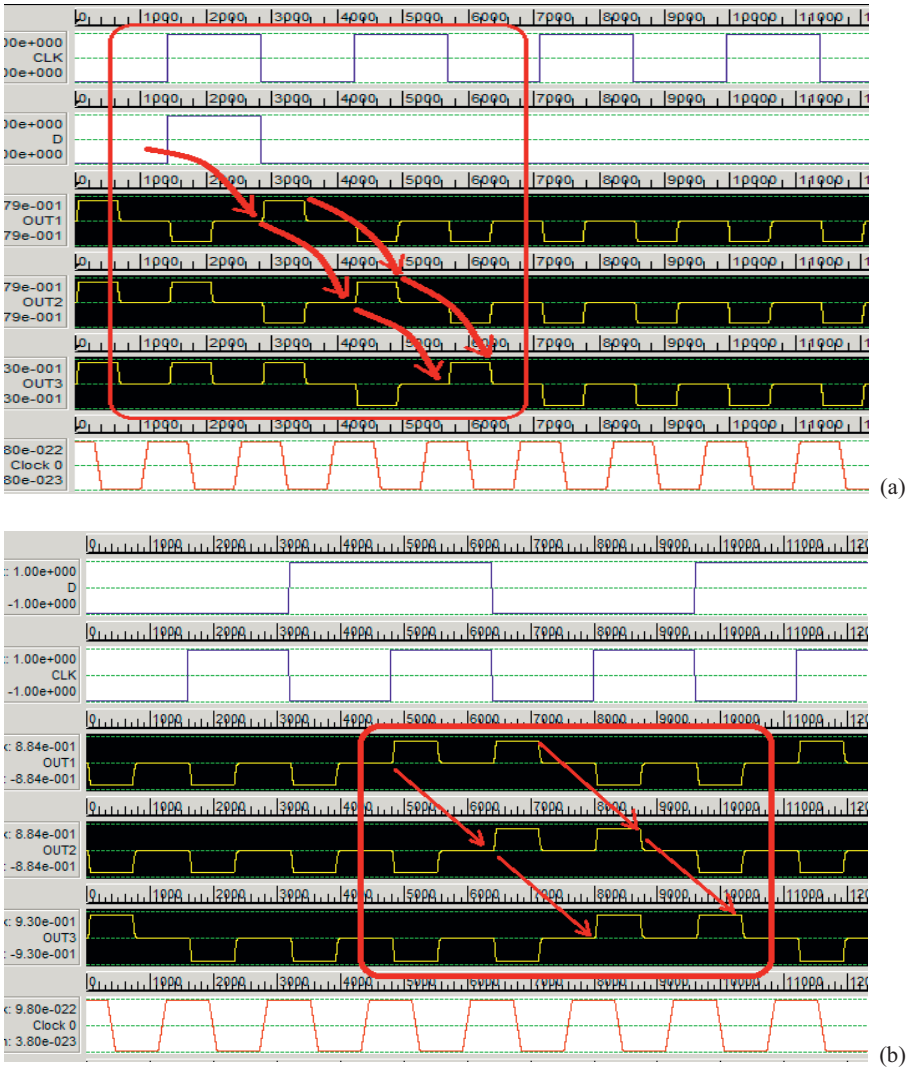


Figure 10. (a) Vector table simulation result; (b) Exhaustive simulation result.

operands, but not both, has a value of true. It is designed use of majority gates and inverters. Formula below presents the equation for this gate.

$$\mathrm{XOR} = \mathrm{MV}(\mathrm{MV}(\mathrm{A}', \mathrm{B}, 0), \mathrm{MV}(\mathrm{A}, \mathrm{B}', 0), 1). \quad (2)$$

(XOR) and exclusive-OR (NOR) gates are also used in digital circuits, have been proposed by different authors [8–10]. The design has complexity of cells and either coplanar crossovers or multiple layers to implement. In this paper there was an attempt to design simple XOR structures. The inputs

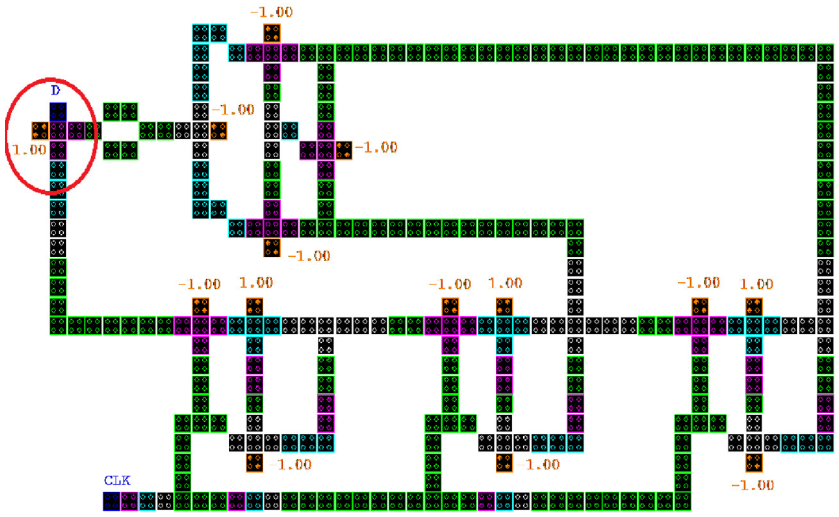


Figure 11. Layout design of pseudo-code generator.

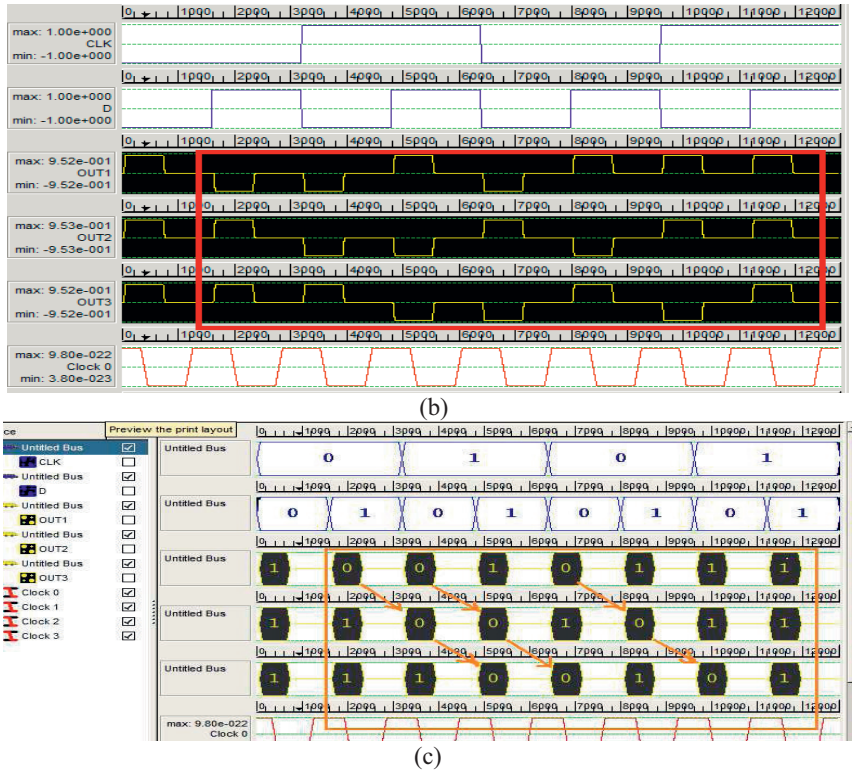


Figure 12. Exhaustive simulation result of pseudo-code.

Table 2. Comparison of conventional and proposed layouts.

Logic structures	Complexity no. of cells	Area (μm^2)	Latency clocking cycles
Conventional structures	84	0.08	1
	64	0.07	1
Proposed structures	34	0.06	1
	42	0.05	1/2

of circuit are A and B and the output is XOR signal. In this design, multilayer has not been used and this caused considerable reduction in the number of cells in circuit. The circuit layout has been displayed in Figure 7a.

The XOR gate was considered for various inputs and their simulation results have been shown in Figure 7b.

The number of cells of the proposed gates and their polarities has been compared with previous structures [8–10]. Comparisons between the proposed and conventional XOR gates are given in Table 2.

4.3 Serial-in/serial-out (SISO) shift register

In this subsection, serial-in/serial-out (SISO) shift register has been proposed using QCA D-latch [7] structure. These registers are a type of sequential logic circuit, mainly for storage of digital data. They are a group of D-flip-flops, connected in a chain so that the output from one flip-flop becomes the input of the next flip-flop. Most of the registers possess no characteristic internal sequence of states. All flip-flops are driven by a common clock, and all are set or reset simultaneously. The schematic structure of serial-in/serial-out (SISO) shift registers is shown in Figure 8a and its timing diagram is shown in Figure 8b respectively.

The layout design of serial-in/serial-out shift register is shown in Figure 9. A serial-in/serial-out shift register has a clock input, a data input, and a data output from the last stage. In general, the other stage outputs are not available otherwise; it would be a serial-in, parallel-out shift register.

Both vector table setup, exhaustive verification simulation result of serial-in/serial-out shift register is shown Figures 10a and 10b respectively. The waveforms below are applicable to either one of the preceding two versions of the serial-in, serial-out shift register. The three pairs of arrows show that a three stage shift register temporarily stores 3-bits of data and delays it by three clock periods from input to output.

4.4 Pseudo-code generator

Pseudo-code generator is shown in Figure 11. The dashed circle used in Figure 11 can be used to initilize the inputput serial data bit of pseudo-code generator. Thereafter length of sequence will repeat itself as per formula (1). To distinguish clock zones, four different colors have been employed. These different zones are required for timing circuit.

The simulation result of exhaustive simulation result of pseudo-code generator is shown in Figure 12.

5 Conclusion

QCA implementation of pseudo-code generator is discussed in this paper. The main blocks of this algorithm which are D-latch, XOR-gate and shift registers, have been implemented and simulated separately. It is demonstrated that the implementation of cryptographic algorithms in this technology has considerable advantageous as compared to conventional CMOS approach. It has shown that the QCA design of logic circuits is challenged with unique features at logic level. In this respect, it has been proved that design of PN-Code generators offers the best advantage of versatility and ease of implementation using the communication circuits. Simulations were performed using a set of Vector table simulation and exhaustive simulation result of QCA designer.

Acknowledgements. One of the authors is highly thankful to UGC, India for providing Dr. D.S. Kothari UGC-PDF for carrying out this work. We are also thankful to Prof.G.M. Bhat and Mr. Hilal A. Khan of Electronics and Instrumentation Department for help and necessary discussions.

References

1. Lent CS, Taugaw PD, Porod W, Berstein GH. 1993. Quantum cellular automata. Nanotechnology, 4, 49.
2. Cho H, Swartzlander EE Jr. 2009. Adder and multiplier design in quantum-dot cellular automata. IEEE Transactions on Computers, 58(6), 721–727.
3. Amlani I, Orlov AO, Toth G, Bernstein GH, Lent CS, Snider GL. 1999. Digital 7 logic gate using quantum-dot cellular automata. Science, 284(5412), 289–291.
4. Lent CS, Tougaw PD. 1997. A device architecture for computing with quantum dots. Proceedings of the IEEE, 85(4), 541–557.
5. Tougaw PD, Lent CS. 1996. Dynamic behavior of quantum cellular automata. Journal of Applied Physics, 80(8), 4722–4735.
6. Tougaw PD, Lent CS, Porod W. 1993. Bistable saturation in coupled quantum-dot cells. Journal of Applied Physics, 74(5), 3558–3565.
7. Walus K, Vetteth A, Jullien GA, Dimitrov VS. 2003. RAM design using quantum-dot cellular automata. Technical Proceedings of the Nanotechnology Conference and Trade Show, 2, 160–163.
8. Karthigai Lakshmi S, Athisha G. 2010. Efficient design of logical structures and functions using nanotechnology based quantum dot cellular automata design. International Journal of Computer Applications, 3(5), 0975–0887.
9. Tougaw PD, Lent CS. 1999. Logical devices implementation using quantum dot cellular automata. Journal of Applied Physics, 75, 1818.
10. Ch VT, Polisetti S, Santhosh K. 2008. QCA based multiplexing of 16 arithmetic & logical subsystem-a paradigm for nano computing, 3rd Annual IEEE-International conference on Nano/Micro Engineering Molecular System, Hainan Island, China.
11. Vankamamidi V, Ottavi M, Lombardi F. 2008. Two dimensional schemes for clocking/timing of QCA circuits. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 27(1), 34–44.
12. Momenzadeh M, Huang J, Lombardi F. 2005. Defect characterization and tolerance of QCA sequential devices and circuits, IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems, Monterey, California, USA. p. 199.
13. Huang J, Momenzadeh M, Lombardi F. 2007. Design of sequential circuits by quantum-dot cellular automata. Microelectronics Journal, 38, 525–537.
14. Yang XK, Cai L, Zhao XH, Zhang NS. 2010. Design and Simulation of Sequential Circuits in Quantum-Dot Cellular Automata: Falling Edge-Triggered Flip-Flop and Counter Study. College of Science, Air Force Engineering University: Xi'an, China.

Cite this article as: Ahmad F, Mustafa M, Wani AN & Mir AF: A novel idea of pseudo-code generator in quantum-dot cellular automata (QCA). Int. J. Simul. Multisci. Des. Optim., 2014, **5**, A04.