Indian Journal of Pure & Applied Physics
Vol. 52, March 2014, pp. 203-209

# Novel linear feedback shift register design in quantum-dot cellular automata

M Mustafa & M R Beigh*

Department of Electronics & Instrumentation Technology, University of Kashmir, Kashmir, India

*E-mail: mrbku@yahoo.com

*Received 25 February 2013; revised 3 September 2013; accepted 23 December 2013*

Quantum-dot Cellular Automata (QCA) is one of the candidates among future nanoelectronic computing technologies. QCA is based on cells of coupled quantum dots. The QCA cells have features on the very low nanometer scale, much smaller than the present state of art size of the smallest transistor. The physical interaction between neighbouring cells has been exploited in the implementation of logic functions. This paper reviews the basic paradigm of QCA and presents efficient design and layout of a novel linear feedback shift register (LFSR) with minimum complexity and cell count. The proposed LFSR can be effectively used to design more complex circuits like scramblers and pseudo random pattern generators. The proposed design was verified by carrying out simulation using the QCADesigner tool. These implementations and simulations are useful for building more complex digital communication based circuits in QCA.

**Keywords:** Shift register, Nanoelectronics, Quantum cellular automata, Linear feedback shift register, QCADesigner

## 1 Introduction

The VLSI CMOS technology dominated the integrated circuit industry from the past number of decades unless it reached its fundamental limits. Gordon Moore predicted, in 1965, that the capacity of a computer chip would grow exponentially with time. Moore's Law had governed the development and manufacturing of microprocessors and other VLSI devices since then. Shrinking transistor size has been the major trend to achieve circuits with better performance parameters. However, when scaling is brought down to submicron level, many problems occur with regard to performance of the circuits. Physical limits such as quantum effects and non-deterministic behaviour of small currents together with technological limits like power dissipation, heat removal and design complexity may hinder the further progress of microelectronics using conventional circuit scaling. Consequently to maintain trends of increasing microprocessor performance, alternative technologies need to be explored and developed. As an alternative to present CMOS, researchers have proposed an approach to replace conventional transistor with quantum dots in the form of cell, the quantum-dot cellular automata[1] (QCA). QCA is based upon the encoding of binary information in the charge configuration within quantum dot cells. The Coulombic interaction between neighbouring QCA cells provides the computational power. The local interconnections between cells are provided by the physics of cell-to-cell interaction due to the rearrangement of electron positions[2]. Recent work showed that QCA can achieve high density, fast switching speed and room temperature operation[3-5]. In recent years, various QCA based logic circuit designs have been proposed[6-11]. The objective of this paper is to propose a detailed design, layout and simulation of a novel LFSR with minimum complexity and cell count. An optimal design for a Shift Register and Linear Feedback Shift Register (LFSR) is proposed to maximize the circuit density and focus on the layouts that are simple and minimal in their number of cells. The proposed QCA circuits have been designed and simulated using the QCADesigner tool.

## 2 Quantum-dot Cellular Automata

Quantum-dot Cellular Automata has emerged as a new paradigm, beyond current switches to encode binary information. QCA encodes binary information in the charge configuration within a cell. Coulomb interaction between cells is sufficient to accomplish the computation in QCA arrays, thus no interconnect wires are needed between cells. Low power dissipation is possible as no current flows out of the cell[12,13].

QCA cells perform computation by interacting coulombically with neighbouring cells to influence each other's polarization. A high-level diagram of a four-dot QCA cell appears in Fig. 1. Four quantum dots are positioned to form a square. Quantum dots are small semiconductor or metal islands with a diameter that is small enough to make their charging

energy greater than $k_BT$ (where $k_B$ is Boltzmann's constant and $T$ is the operating temperature in degrees kelvin). In the future with advent of technology, they will shrink to regions within specially designed molecules[4]. If this is the case, they will trap individual charge barriers[14,15]. Exactly two mobile electrons are loaded in the cell and can move to different quantum dots in the QCA cell by means of electron tunneling. Coulombic repulsion will cause the electrons to occupy only the corners of the QCA cell resulting in two specific polarizations[16] as shown in Fig. 1.

For an isolated cell there are two energetically minimal equivalent arrangements of the two electrons in the QCA cell, denoted by cell polarizations P = +1 and P = −1 representing a binary 1 and a binary 0, respectively. It is also worth noting that there is an unpolarized state as well. In an unpolarized state, interdot potential barriers are lowered which reduce the confinement of the electrons on the individual quantum dots. As a result, the cells exhibit little or no polarization and the two-electron wave functions delocalize across the cell[15].

The fundamental QCA logical circuit is the three-input majority gate[14] that appears in Fig. 2. Computation is performed with the majority gate by driving the device cell to its lowest energy state. This occurs when it assumes the polarization of the majority of the three input cells.

All QCA circuit proposals require a clock not only to synchronize and control information flow but the clock actually provides the power to run the circuit. QCA computation is performed by controlling the tunneling with a four phase "clock" signal as shown in Fig. 3. The clocking of QCA can be accomplished by controlling the potential barriers between adjacent quantum-dots[17,18]. The clock used in QCA consists of four phases: hold, release, relax, and switch. The lag between adjacent phases is 90°. Rather, it can be said that the clock changes phase when the potential barriers that affect a group of QCA cells (referred to as a clocking zone) are raised or lowered or remain raised or lowered.

During the switch phase, the inter-dot barrier is gradually raised, and the QCA cell settles down to one of the two ground polarization states as influenced by its neighbours. During the hold phase, the inter dot barrier is held high, preventing electron tunneling and maintaining the current ground polarization state of the QCA cell.

During the release and relax phases, the inter dot barriers are comparatively low, and the excess electrons gain movement. In these two phases, a QCA cell remains unpolarized. Overall, the polarization of a QCA cell is determined when it is in its switch phase by the polarizations of its neighbours that are in switch and hold phases. The unpolarized

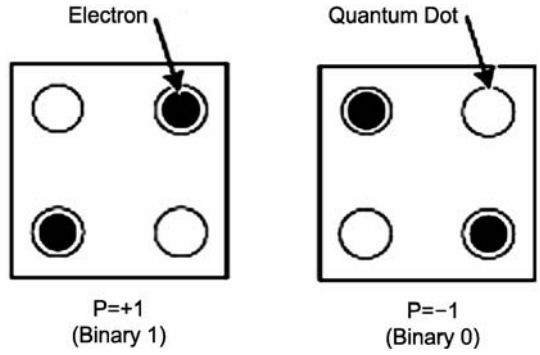



Fig. 1 — QCA cell polarizations and representations of binary 1 and 0

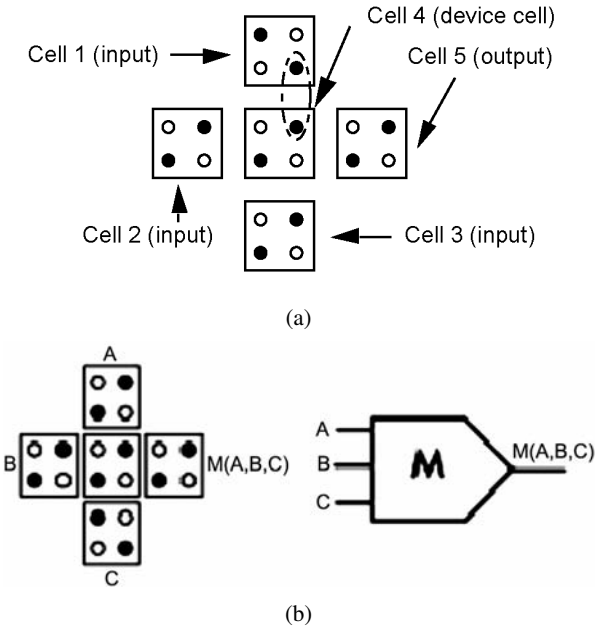



Fig. 2 — (a) The fundamental QCA logic device- the majority gate. (b) The circuit symbol

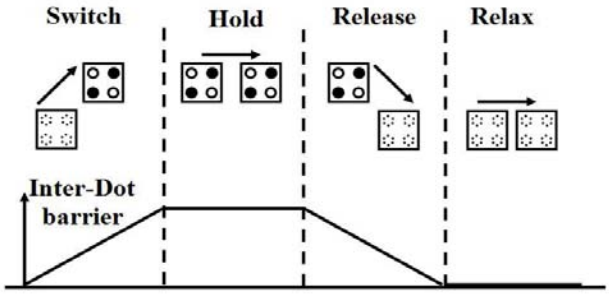



Fig. 3 — 4 phases of QCA Clock

neighbouring cells in release and relax phases have no effect on determining the state of the QCA cell[19]. The clock signals (through an induced electric field) can be generated by CMOS wires embedded below the QCA plane[15].

All the layouts and simulations presented in this paper have been carried out using QCADesigner[21]. QCADesigner is the product of an ongoing effort to create a rapid and accurate simulation and layout tool for QCA. QCADesigner is capable of simulating complex QCA circuits on most standard platforms. Initially developed at the ATIPS Laboratory, University of Calgary, Calgary, Canada, QCADesigner has attracted some important new developers, including top researchers worldwide. The objective of the project is to create an easy to use simulation and layout tool available freely to the researchers in the field via the internet. Additionally, extensive effort is being put forth by the developers to enhance QCADesigner and create a large, useful toolbox for QCA designers.

Included in the current version of QCADesigner are three different simulation engines. The first is a digital logic simulator, which considers cells to be either null or fully polarized. The second is a nonlinear approximation engine, which uses the nonlinear cell-to-cell response function to iteratively determine the stable state of the cells within a design. The third uses a two-state Hamiltonian to form an approximation of the full quantum mechanical model of such a system. One of the main problems in implementing more accurate simulations is the lack of experimental data for QCA systems with a large number of cells. However, several small QCA systems have been developed as proof-of-concept experiments[22-25].

Many causes of inefficiency of QCADesigner tool have been identified and need to be improved upon for efficient designing of the logic circuits. A very good work could be taken up for development of a library for QCADesigner tool that would convert conventional digital schematic containing AND gates, OR gates, NAND gates, XOR gates, etc. to a schematic layout consisting of QCA majority gate logic. Research and design efforts which are currently underway focus on improving the "user-friendliness" of QCADesigner.

A lot of work is being done by various researchers at various levels in development of standalone tools as to incorporate with the QCADesigner. An example tool that is currently under development is a fault modeling tool to examine the various effects of manufacturing defects within QCA systems that will aid in the development of defect tolerant QCA systems[26]. Additionally, QCADesigner will be capable of handling new QCA technologies as they become viable for implementing systems. The capabilities already available and the inherent ability for growth within QCADesigner provide a robust tool for designing QCA circuits. Planned improvements for future versions include a design rule checker (DRC), hierarchical design capabilities, a fault tolerance simulation, and more accurate simulation and clocking models.

One of the main differences between circuit design in QCA and circuit design using conventional CMOS technologies and devices is that the circuit has no control over the clocks. This means that information is transmitted through each cell and not retained. Each cell erases its own state every cycle of the clock. As a result, memory units must be created using loops of cells that continuously circulate the stored information. Additionally, implementing sequential logic with shift registers is an attractive approach to the design for testability. It requires little extra circuit for scan and since feedbacks are usually to the first flip-flop of a shift register test generation is expected to be easier than for circuits with unstructured feedbacks.

## 3 QCA Implementation

The AND and OR gates are realized by fixing the polarization to one of the inputs of the majority gate to either P = −1 (logic "0") or P = 1 (logic "1"). The NAND function is the complement of AND function. It is realized by connecting AND gate followed by an inverter. Similarly the NOR gate is realized by connecting OR gate followed by an inverter. If the last two cells are arranged as shown in Fig. 4 then it acts as an inverter. By using this 2 cell inverter, the area required and complexity can be minimized.

Exclusive OR, also known as Exclusive disjunction and symbolized by XOR, is a logical operation on two operands that results in a logical value of true if and only if one of the operands, and not both, has a value

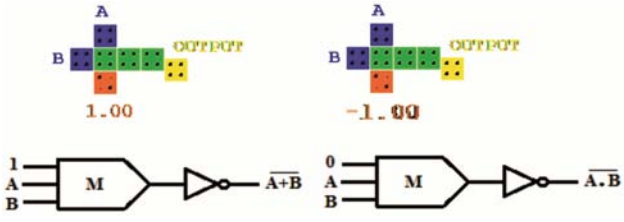



Fig. 4 — Layout of NOR and NAND gate

of true. This forms a fundamental logic gate in many operations to follow. If a specific type of gate is not available, it can be build from other available gates. An XOR gate can be constructed from an XNOR gate followed by a NOT gate. Considering the basic expression, $\overline{A}B + A\overline{B}$ an XOR gate can be constructed directly using AND, OR and NOT gates. This approach, however, requires five gates of three different kinds.

Logically, the exclusive OR (XOR) operation can be seen as either of the following operations:

1 $\overline{A}B + A\overline{B}$

2 $(A + B)(\overline{AB})$

These two operations can be implemented by the gate arrangements shown in Fig. 5(a and b) respectively. They also can be implemented using NAND gates only.

The QCA implementation for the layout shown in Fig. 5(a) has been proposed by different researchers[14,20]. This design needs either coplanar crossovers or multiple layers to implement. The design provided as a sample file with QCADesigner[21] Version 2.0.3 needs crossovers and uses three layers to implement.

We have already proposed the QCA design and layout of XOR gates[27,31] based on logic gate arrangements shown in Fig. 5(b and c). These layouts are shown in Fig. 6. These designs do not require any crossover and have minimum number of cell count. According to QCADesigner, the design shown in Fig. 6(a) has latency of only one clock cycle and consists of just 41 cells (including input and output cells) and an area of approximately 0.07 μm$^2$.

The simulation results for the layouts as shown in Fig. 6(a and b) are shown in Fig. 7(a and b), respectively. As observed from the simulation results the first layout has a latency of one clock cycle and the second layout has a latency of two clock cycles. These layouts can be easily used to design complex circuits based on XOR operation. We present here the design of a LFSR based on the layout shown in Fig. 6(b).

A QCA D-type Flip-flop (FF) can be constructed by a QCA binary wire with four clocking zones[28] as shown in Fig. 8. Latching can be accomplished through timing[28] by using a four-phase clocking arrangement as shown in Fig. 8.

In this case, the input signal is delivered to the output after at least one complete clock cycle delay and control is accomplished by timing. The comparative simplicity of a D-FF seems to suggest

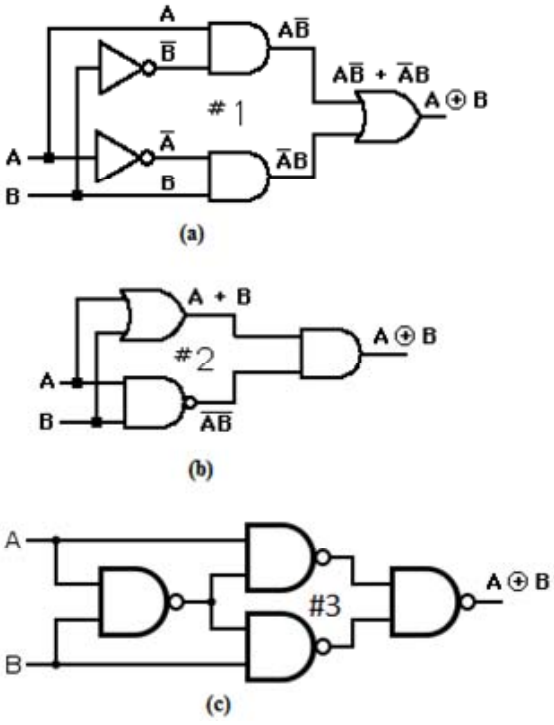



Fig. 5 — Implementations of XOR

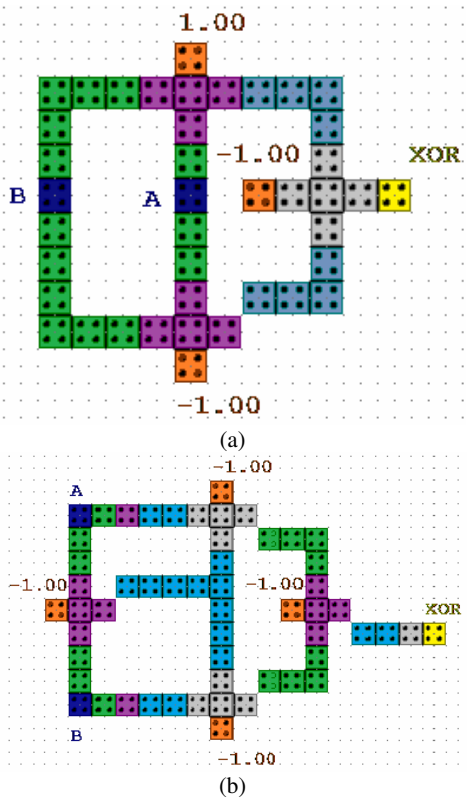
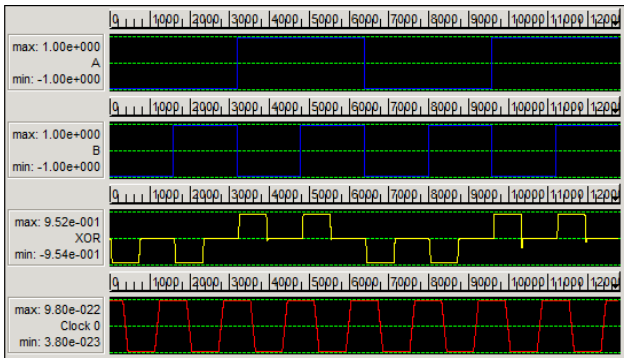
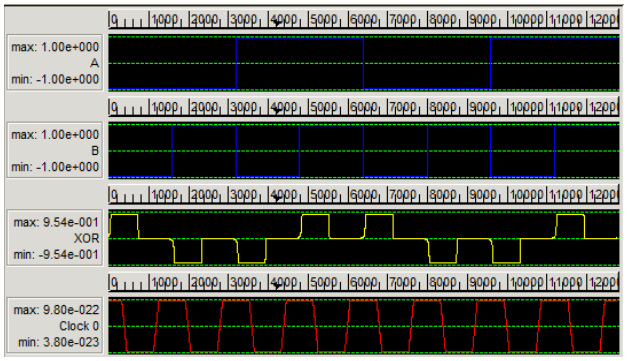



Fig. 6 — QCA XOR implementations




(a)




(b)

Fig. 7 — Simulation results of XOR layouts

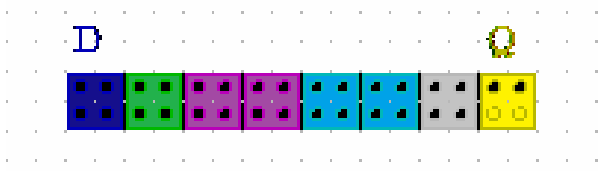



Fig. 8 — QCA D flip-flop

that sequential design in QCA could be achieved at ease within the Cartesian layout[28]. However, this implementation does not have a separate clock control signal. Also timing and signal delay must be carefully considered. We have used the QCA implementation of D flip-flop[29] as shown in Fig. 9. It has an additional clock control signal.

The current QCA technology does not specifically set the possible operating frequency and actual propagation delays. However, the most interesting aspect of the use of molecules, in case of Molecular QCA implementation[5], is that the switching speed expected, from one charge configuration to another: it grants the possibility to obtain operating frequency of some THz. Moreover, the dimensions of such molecules are very small (a few nanometers), allowing the generation of circuits with a very high device densities[30].

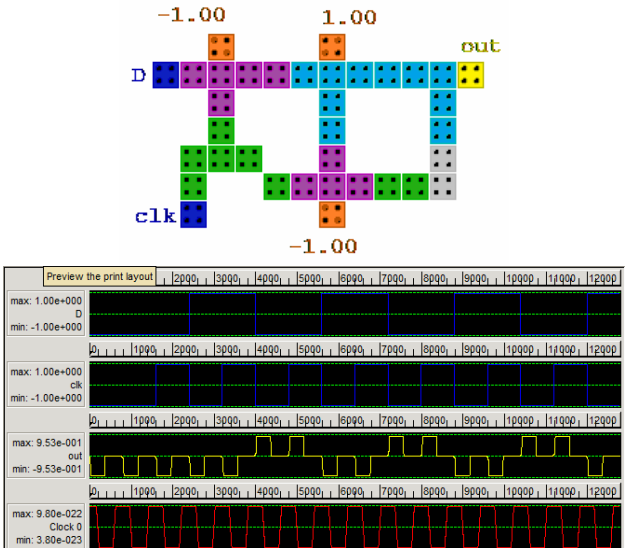



Fig. 9 — QCA D flip-flop with clock control and Simulation results

A shift register is a sequential circuit that, when clocked, advances the signal through the register from one bit to the next most-significant bit as shown in Fig. 10.

The 3-bit QCA shift register design is shown in Fig. 11. The proposed QCA shift register consists of 3 D-FFs and has been designed and simulated using the QCADesigner[21] tool. This tool allows users to do a custom layout and then verify QCA circuit functionality by simulations. The present QCA technology does not distinctively set the possible operating frequency and actual propagation delays. Therefore, the maximum cell count can be set as a design parameter.

A Linear Feedback Shift Register (LFSR) is a sequential shift register with combinational logic that causes it to pseudo-randomly cycle through a sequence of binary values. In order to form a feedback mechanism, some of the outputs are combined in exclusive-OR configuration. A linear feedback shift register 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 Fig. 12.

Linear feedback shift registers make extremely good pseudorandom pattern generators. When the flip-flops are pre-loaded with a seed value (anything except all zeroes, which would cause the LFSR to produce all 0 patterns) and when the LFSR is clocked, a pseudorandom pattern of 1s and 0s will be generated. The only signal necessary to generate the test patterns is the clock. Linear feedback shift

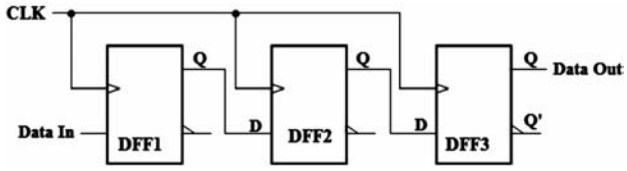
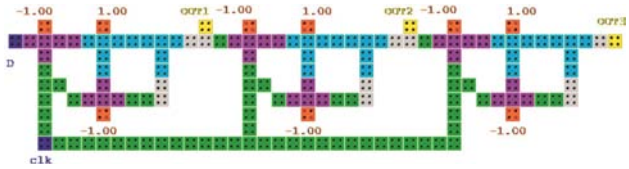



Fig. 10 — Schematic of 3-bit Shift register



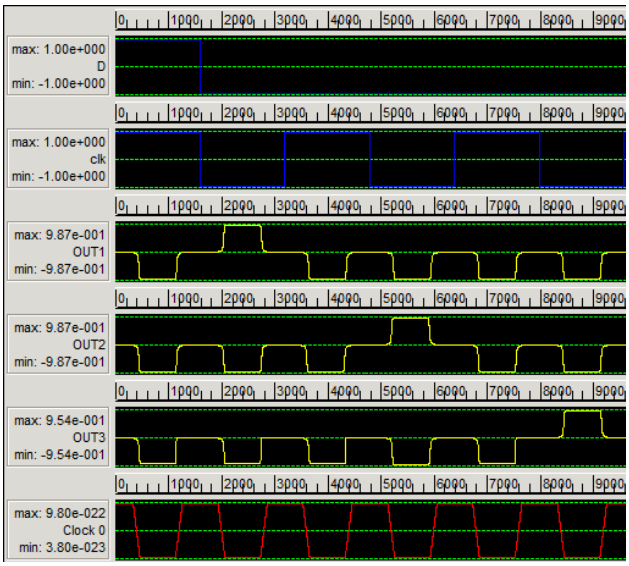



Fig. 11 — QCA layout and simulation results of 3-bit Shift register

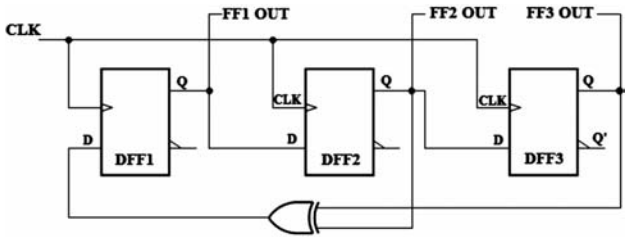
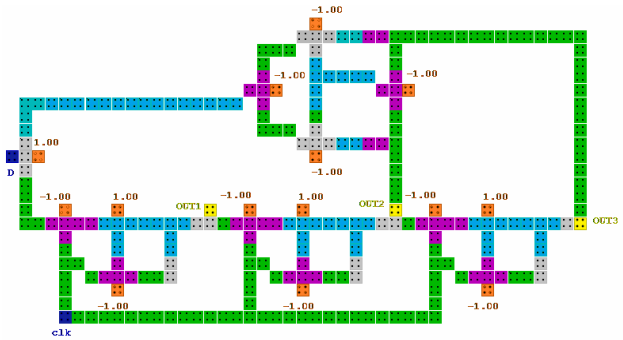
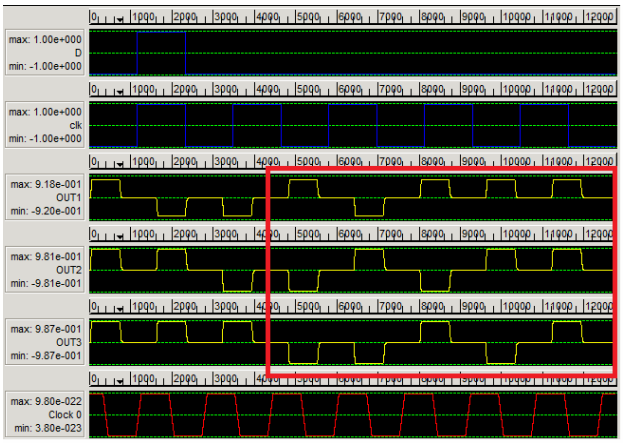



Fig. 12 — Schematic of Linear feedback shift register

registers have multiple uses in digital systems design. Its applications include Data Encryption/Decryption, Digital Signal Processing, Data Compression, Wireless Communications, Built-in Self Test, Data Integrity Checksums, Pseudo-random Number Generation, Scrambler/Descrambler, Optimized Counters and Direct Sequence Spread Spectrum.



(a)




(b)

Fig. 13 — QCA layout and simulation results of 3-bit LFSR

Feedback around an LFSR's shift register comes from a selection of points (taps) in the register chain and constitutes XORing these taps to provide tap(s) back into the register. Those register bits that do not need an input tap, operate as a normal shift register. It is this kind of feedback that causes the register to loop through repetitive sequences of pseudo-random value. The number of values in a given sequence before the sequence repeats depends upon the choice of taps.

The QCA layout of 3-bit LFSR is shown in Fig. 13(a). It consists of 252 cells with an area of 0.47 $\mu m^2$. Figure 13(b) shows the simulation results for this layout. The D input induces the seed value into the XOR gate which is then rotated by the Shift Register to produce the desired pseudo random pattern. The proposed LFSR can be used to design more complex circuits like scramblers and pseudo random pattern generators.

## 4 Conclusions

This paper presents the design, layout and simulation of a novel Shift Register and a Linear Feedback Shift Register. We proposed an optimal

design for LFSR based on efficient D flip-flop and EXOR gate configurations. The proposed layouts were simulated using QCADesigner, the design and simulation tool for QCA based circuits. We conclude QCA technology as one of the promising nanotechnologies in future that can be used to build more complex digital communication circuits, arithmetic logic units and microprocessors etc. There are further opportunities for optimization which could lead to densities greater than reported in our present work and could be taken up for further studies. The current QCA technology does not specifically set the possible operating frequency and actual propagation delays, but it can be investigated as an important parameter in future works. This research work is an attempt to find a reasonable and optimum, way of realizing combinational and sequential circuits designed from a simple QCA based XOR gates and Shift Registers.

## References


1 Lent C S, Tougaw P D, Porod W & Bernstein G H, *Nanotechnology*, 4 (1993) 49.
2 Orlov A O, Amlani I, Bernstein G H, Lent C S & Snider G L, *Science*, 277 (1997) 928.
3 Smith C G, *Science*, 284 (1999) 274.
4 Isaksen B & Lent C S, *Third IEEE Conference on Nanotechnology,* IEEE-NANO 2003, 1 (2003) 5.
5 Lent C S & Isaksen B, *IEEE Transactions on Electron Devices*, 50 (2003) 1890.
6 Mariam Z M & Keivan N, *Microelectronics Journal*, 43 (2012) 377.
7 Keivan N, Horialsadat H S, Reza F M, Mohammad H M, Ali J, Omid K, *Integration the VLSI Journal*, 44 (2011) 155.
8 Mohsen M A Mohsen S, Keivan N, Mohammad A. Tehrani, *2012 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, (2012) 147.
9 Mohsen Hayati & Abbas Rezaei, *ETRI Journal*, 34 (2012) 284.
10 Karthigai Lakshmi S & Athisha G, *European Journal of Scientific Research*, 53 (2011), 524.
11 Liang Lu, Weiqiang Liu, Maire O'Neill & Earl E. Swartzlander, *IEEE Transactions on Computers*, (2011).
12 Timler J & Lent C S, *Journal of Applied Physics*, 91 (2002) 823.
13 Kummamuru R K, Timler J, Toth G, Lent C S, Ramasubramaniam R, Orlov A O, Bernstein G H, & Snider G L, *Appl Phys Lett*, 1 (2002) 1332.
14 Tougaw P D & Lent C S, *Journal of Applied Physics*, 75 (1994) 1818.
15 Lent C S & Tougaw P D, *Proceedings of the IEEE*, 85, (1997) 541.
16 Ramprasad Ravichandran, Nihal Ladiwala, Jean Nguyen, Michael T. Niemier, Sung Kyu Lim, *Integration, the VLSI Journal*, 38 (2005) 541.
17 Snider G, Orlov A, Lent C S, Bernstein G H, Lieberman M, Fehlner T, *ICONN 2006*, (2006) 544.
18 Toth G & Lent C S, *Journal of Appl Phys*, 85 (1999) 2977.
19 Kyosun Kim, Kaijie Wu, and Ramesh Karri, *IEEE Trans. On Computer-Aided Design of Integrated Circuits and System*, 26 (2007) 176.
20 Vishnu T C, Polisetti S & Santhosh K, *3rd Annual IEEE-International conference on Nano/Micro Engineered Molecular Systems*, *China*, (2008) 758.
21 Walus K, Dysart T & Jullien et al, *IEEE transactions on Nanotechnology conf*, 3 (2004) 26.
22 Amlani I, Orlov A O, Kummamuru R K *et al*., *Appl Phys Lett,* 77 (2000) 738.
23 Orlov A O, Amlani I, Kummamuru R K *et al*, *Appl Phys Lett,* 77 (2000) 295.
24 Amlani I, Orlov A O, Toth G *et al.*, *Science*, 284 (1999) 289.
25 Orlov A O, Amlani I, Toth G *et al.*, *Appl Phys Lett,* 74 (1999) 2875.
26 Dysart T J & Kogge P M, *Proc IEEE 3rd Conf Nanotechnology,* 1 (2003).
27 Mustafa M & Beigh M R, *Indian J Pure & Appl Phys*, 51 (2013) 60.
28 Huang J, Momenzadeh M & Lombardi F, *Microelectronics Journal*, 38 (2007) 525.
29 Abbas Shahini Shamsabadi, Behrouz Shahgholi Ghahfarokhi, Kamran Zamanifar & Naser Movahedinia, *Journal of Systems Architecture*, 55 (2009) 180.
30 Mariagrazia Graziano, Marco Vacca & Maurizio Zamboni, *Cellular Automata-Innovative Modelling for Science and Engineering*, InTech Press, (2011) 37.
31 Beigh M R, Mustafa M & F Ahmad, *Circuits & Systems*, 4 (2013) 147.