Microelectronics Journal 55 (2016) 92–107

Contents lists available at ScienceDirect

Microelectronics Journal

journal homepage: www.elsevier.com/locate/mejo

Design of QCA based Programmable Logic Array using decoder

Debashis De [a,b,*], Tamoghna Purkayastha [a], Tanay Chattopadhyay [c]

[a] Department of Computer Science and Engineering, West Bengal University of Technology, BF-142, Sector-I, Salt Lake City, Kolkata 700064, West Bengal, India
[b] School of Physics, University of Western Australia, M013, 35 Stirling Highway, Crawley, Perth, WA 6009, Australia
[c] Kolaghat Thermal Power Station, West Bengal Power Development Corporation Ltd., Govt. of West Bengal Enterprise, Mecheda, Purba Medinipur 721137, West Bengal, India

ARTICLE INFO

Article history:
Received 10 September 2015
Received in revised form
25 April 2016
Accepted 13 June 2016

Keywords:
QCA
Decoder
PLA
Interfacing
Recursion
Cell count
Crossover

ABSTRACT

A novel cost effective design of Programmable Logic Array (PLA) is proposed by recursive use of XOR gate, which is used to design 2 × 4, 3 × 8 and 4 × 16 decoders. The 4 × 16 decoder is coupled with an OR-Array to implement the proposed PLA using Quantum-dot Cellular Automata (QCA). The design is made effective by substantially reducing QCA wire crossing and cell count. A comparative study shows the minimization of cell count and clock-cycle delay of the XOR and Decoders. The PLA is utilized to design an efficient and delay effective 2 bit full adder.

1. Introduction

Silicon technology is in a challenging phase due to its high power consumption and physical designing limits. Different alternative technologies are developing that might replace CMOS technology. Quantum Dot Cellular Automata (QCA) [1] is one of the notable paradigm shifts in the Nano-scale computational domain. Since its inception in 1993 by Lent et.al, QCA has proved its potential to be a sturdy alternative of CMOS technology. With experimental verification in 1997, the QCA devices with low power dissipation and high device density are presumed to provide around a THz processing speed, occupying few nanometres in area. With high speed switching capabilities, QCA layouts are collection of cells operated by the Columbic interactions. One of the major advantage of QCA is that it acts on the principle of electron polarization [2] as opposed to the transmission of current through devices, thus minimizes the energy consumption. In QCA paradigm the electrostatic influence on the neighbour cells is the key for information transfer. Furthermore room temperature operability of QCA which has been one of its major limitations is made possible in [3].

Each QCA cells are made of four quantum dots situated at four different corners of a square. Extra electrons reside diagonally apart from each other at maximum distance because of Columbic repulsion. The positions of the dots with additional electrons define the polarization (p) state of the cell. A binary 0 and binary 1 is encoded by $p=-1$ and $p=+1$ respectively as depicted in Fig. 1(a). The QCA wire is shown in Fig. 1(b) [1–17]. A QCA cell produces a complementary polarization at the cell, placed diagonally to it. Implementation of this knowledge leads to the foundation of QCA inverters which is shown in Fig. 1(e) [18–22]. Electrostatic interaction properties between the neighbour cells generate the realization of Majority gate in QCA as depicted in Fig. 1(c). The output cell then reflects the information as provoked by the driver cell. Therefore, the majority function Y for three inputs A, B, C can be suggested as: M(A,B,C)=AB+BC+CA. The polarization state of a cell can be measured as the effective energy produced through the Columbic interactions with its neighbour cells which reside in the radius of effect [23]. The wire crossing in QCA can be performed in two ways namely co-planar crossings and multi-layered crossings [24–27,33] as shown in Fig. 1(d). It is predicted that the coplanar coupling is to some extent weaker than it would have been in a normal wire [5]. The stability in coplanar crossover is less compared to multilayer type crossover [6,7]. But implementation of multilayer crossing in QCA is a challenging issue [4]. This is because of the fact that QCA cells have been fabricated only in single layer [28]. A number of variations in coplanar crossing have been

[*] Corresponding author at: Department of Computer Science and Engineering, West Bengal University of Technology, BF-142, Sector-I, Salt Lake City, Kolkata 700064, West Bengal, India.
E-mail address: dr.debashis.de@gmail.com (D. De).

http://dx.doi.org/10.1016/j.mejo.2016.06.005
0026-2692/© 2016 Elsevier Ltd. All rights reserved.

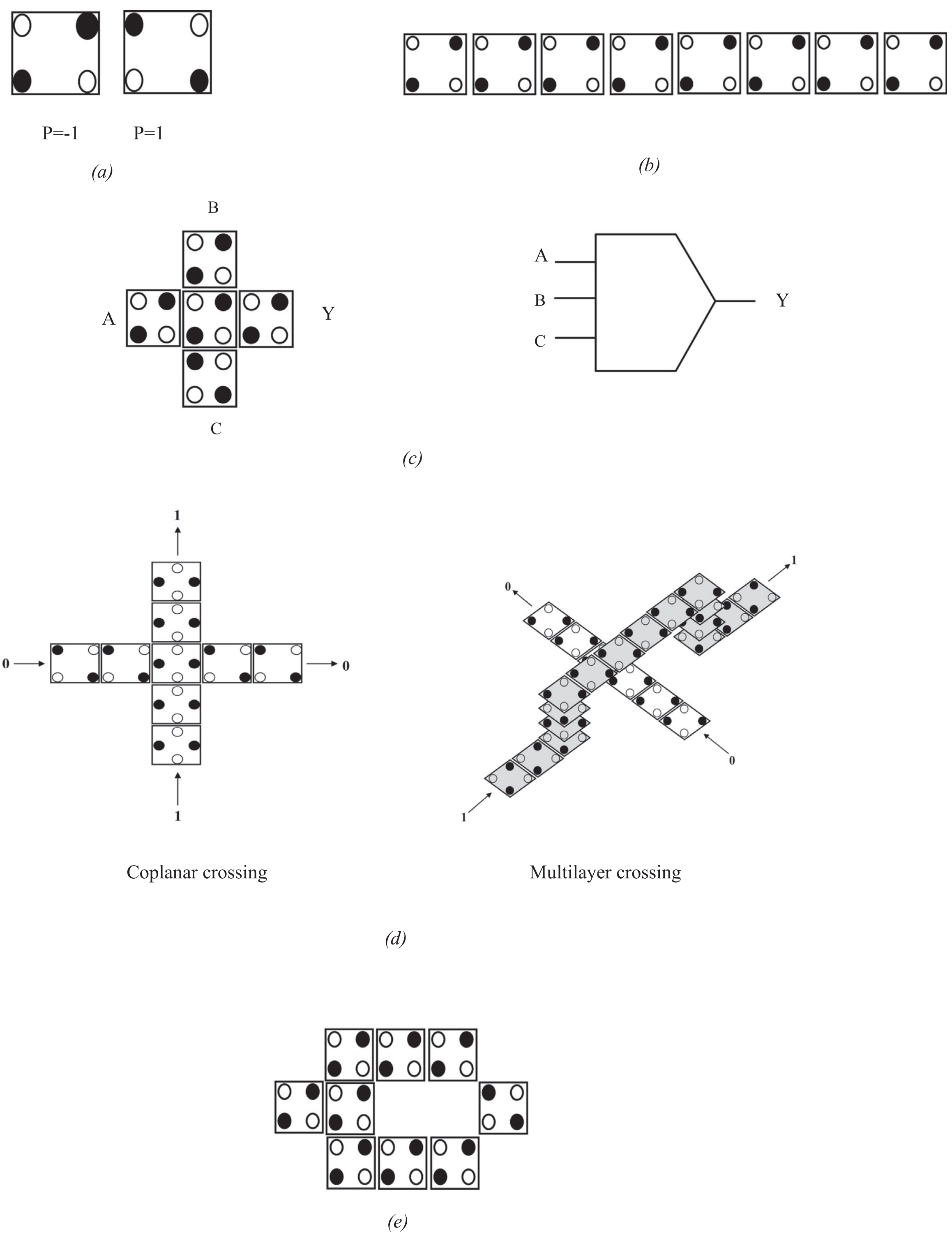


Fig. 1. (a) QCA Cell Polarization. (b) QCA Wire. (c) QCA Majority Voter. (d) QCA Crossover. (e) QCA Inverter.

proposed [6] in order to improve its efficiency. In the proposed design of XOR and Decoders multilayer design have been used but with minimized wire crossing. Thus this proposed design is more efficient and stable then traditional QCA coplanar designs.

Programmable Logic Array (PLA) is a reconfigurable system level architecture. In a PLA the programmable memory device and logic gates are arranged in an array form. The circuit functions of it can be reconfigured even after the fabrication. The advantage of PLA is that a single device can be used to perform various operations. Crossovers in QCA are of important consideration. Basically there are two types of crossovers namely multilayer and coplanar. Though multilayered crossover is difficult to fabricate, [4] the stability of coplanar crossover is less compared to the multilayered crossover [5–7]. The XOR gate and Decoders have been designed using multilayer design with reduced number of crossovers. Furthermore as only 90° cells are used the stability of the circuit is no way hampered. The complexity and performance of QCA circuits [8–10] are evaluated based on certain parameters as cell count, area, latency, stability of output, etc. A strong attempt is made to derive an optimized design of XOR and Decoders. While getting

Table 1
Design parameters for Simulation in QCADesigner [34,36].

Parameter	Value
Cell width	18 nm
Cell height	18 nm
Dot diameter	5 nm
Number of samples	12,800
Radius of effect	80 nm
Relative permittivity	12.9
Clock high	9.8e – 22 J
Clock low	3.8e – 23 J
Clock amplitude factor	2
Layer separation	11.5 nm

Table 2
Truth Table of XOR Logic.

Inputs		Output
A	*B*	$Y = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

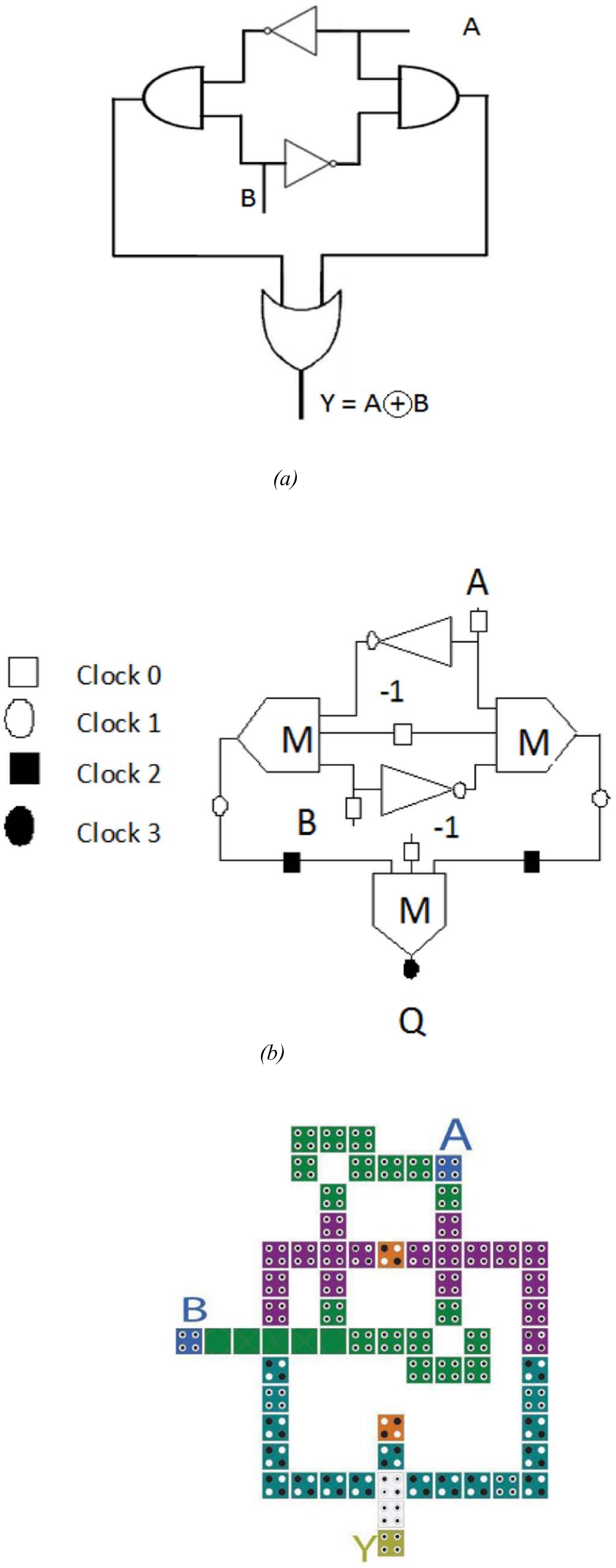
the outputs with a relatively lower delay it is checked whether some of them could be liberated from QCA wire crossovers. The following are the contributions of the proposed work:

1. A novel design of XOR gate is proposed. The model is prepared by recursive use of XOR gate to decoder circuits. The XOR layout has 37% benefits in cell count as compared with [11].
2. The proposed design of 2 × 4 Decoder circuit shows approximately 33% reduction in cell count as compared with previous work [12–17].
3. The 2X4 Decoder design also achieves 50% delay reduction as compared with [12–17].
4. Furthermore an estimation for the number of wire crossings that might be needed for any extension or interfacing with other modules with the proposed designs has also been suggested.
5. Finally a novel design of QCA based PLA is developed with the help of the proposed decoder. The PLA is utilized to design a 2-bit Full Adder.

Rest of the paper is organized as follows. Section 2 provides a review of QCA based Decoders and PLAs. The methodology of the proposed design is discussed in Section 3. Section 4 deals with the design of QCA decoder and its basic building blocks as well as the proposed PLA. The experimental methods to develop QCA system is discussed in Section 5. Analysis of the simulation results and comparative study of our work with other available designs is performed in Section 6. Finally the concluding thought is provided in Section 7.

2. Related works

XOR and Decoder being two of the most useful functions in the Boolean circuit realm have been in the leading attention of many researchers in the QCA domain for a long time. Conventional XOR logic implementations had been proposed during the initial stages



(a)

(b)

(c)

Fig. 2. (a) Schematic diagram of the proposed XOR gate. (b) Majority gate implementation of QCA XOR. (c) QCA circuit for XOR logic Design.

of QCA era. It yields the basic function as ($\bar{A}B + A\bar{B}$) with 3-majority voting implementation. Improvements have been suggested in many approaches [11,29,30]. Jagarlamudi et al. in [11] has developed a XOR structure with 54 cells; those include an area of 290 nm × 220 nm.A series of designs regarding XOR logic implementation has been proposed by Beigh et al. [29] for performance evolution of efficient XOR structure using QCA. This paper includes seven different methods for XOR logic designing with a relative study for efficiency with respect to different parameters. Improvements in different parameters as delay, cell-count, area etc. have been shown in different implementations in this work. Programmable Logic Array (PLA) structure addressing the XOR implementation has been proposed on the course of the developments as well [30].

Classical AND–OR–NOT logics for constructing Decoders also brought into shape in early practices with QCA. Most of the work, with 3-input majority gate implementations, mainly addressed the 2 × 4 decoders. They have been used quite successfully for designing memory architectures in QCA [31]. Later on, efforts have been made intending improvements in decoder circuit implementations as well. 5-input majority gate implementations for designing the same arrived with a reduced structural complexity in the work of Malekpour et al. in [32]. They have suggested a 2 × 4 decoder module with enable line and its reusability to set up decoders of higher cardinality. Another proposal by Zhou et al. [12] revealed layouts for 2 × 4 decoder avoiding wire crossings. The design is composed of a CMVMIN gate with 3 inputs and 2 outputs, which produces the result of majority voting in one of its output lines, producing the outcome of minority in the other one at the same time. This design with latency of 1 clock cycle, proved itself as a potentially good one. Some other recent works on decoders are by Kianpour and Nadooshan [13–15]. Kianpour et al. [13] has proposed a QCA layout of 2 × 4 decoder using 5-input majority voter. Furthermore this 2 × 4 decoder is used as a module to implement higher order i.e. 3 × 8 decoder circuit. In [14] and [15] QCA based decoders are designed to develop a Configurable Logic Block (CLB) structure for FPGA implementation. A 4 bit row/column decoder for QCA ROM design is reported by Mukherjee et al. in [16].

3. Methodology

The results have been obtained from the QCADesigner simulator with version 2.0.3 [33]. The simulation engine has been selected as Coherence Vector. Each QCA cell has a crossectional area of 18 nm × 18 nm and spacing of 2 nm. Furthermore while using multi-layer crossing a minimum of 3 layers is required for error free data transfer as per the design rule mentioned in [34–36]. Apart from these basic design rules the QCADesigner has a set of default parameter which is shown in the Table 1 [34–37]. The proposed 4 × 16 Decoder is utilized to design a PLA. In order to design the PLA circuit the output of the 4 × 16 Decoder to the OR arrays has to be connected. So, the delay of the PLA is increased to 7 clock cycles in PLA as it is observer from the output in Fig. 7e.

4. QCA based design of the proposed work

4.1. QCA based XOR

A novel layout of QCA XOR gate is explored in this work. Truth table of the XOR gate is provided in Table 2. Analysis conveys the fact that the overall circuit can be optimized if the relevant outputs as $\bar{A}B$ and $A\bar{B}$ can be aligned. So one single fixed cell is used with polarization −1 in the majority functions to perform the AND operations in both of the cases i.e. $\bar{A}B$ and $A\bar{B}$. Fig. 2(a) and (b) suggests this concept.

Two MVs generating $\bar{A}B$ and $A\bar{B}$ are ORed together to obtain the output of XOR gate. The Majority gate implementation and QCA layout are shown in Fig. 2(b) and (c) respectively. The proposed 3-input majority function implementation of a XOR gate is shown in Eq. (1):

$$\begin{aligned} Y &= M\left(F1\left(\bar{A}, 0, B\right), 1, F2\left(A, 0. \bar{B}\right)\right) \\ &= M\left(\bar{A}B, 1, A\bar{B}\right) \\ &= \bar{A}B + A\bar{B} \end{aligned} \tag{1}$$

4.2. QCA Decoder

The proposed XOR gate of Section 3.1 is extended to design 2 × 4 Decoder. A 2 × 4 decoder consists of those two previous components as $\bar{A}B$ and $A\bar{B}$, along with two other, $\bar{A}\bar{B}$ and AB. The proposed procedure for designing decoders results in reduced cell

Table 3
Truth table for 2 × 4 decoder.

Inputs		Outputs			
A	B	$\bar{A}\bar{B}$	$\bar{A}B$	$A\bar{B}$	AB
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

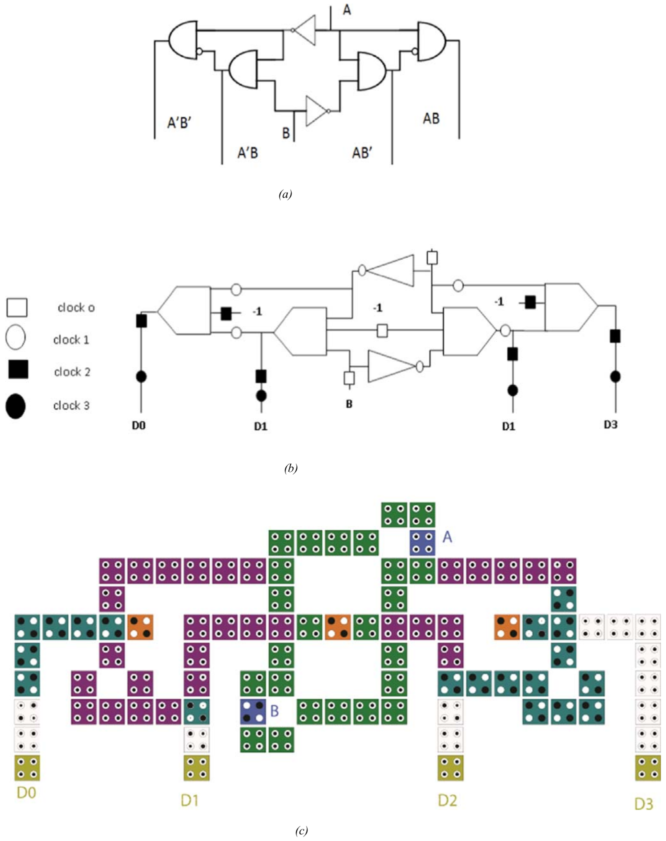
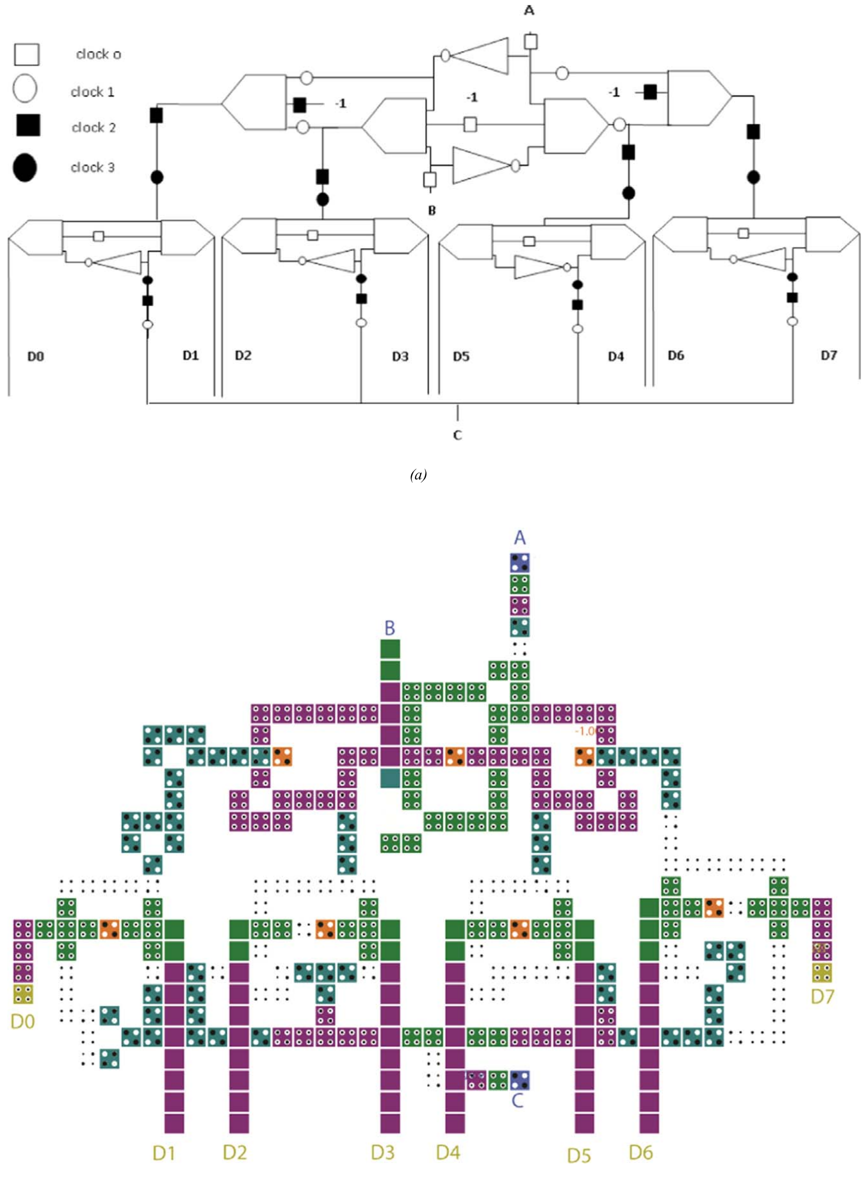


Fig. 3. (a). Schematic for 2 × 4 decoder circuit. (b) 3-input majority gate implementation of 2 × 4 QCA Decoder. (c). QCA circuit for 2 × 4 QCA Decoder Circuit.



(a)

(b)

Fig. 4. (a) 3-input Majority gate implementation of 3 × 8 QCA Decoder. (b) QCA circuit for 3 × 8 QCA Decoder Circuit.

count. The equations to implement the Decoder are given in Eqs. (2) and (3).

A. **Calculation for** $\bar{A}\bar{B}$

$$
\begin{aligned}
P &= (\overline{\bar{A}B}).\bar{A}\\
&= (\bar{\bar{A}} + \bar{B}).\bar{A}\\
&= (A + \bar{B}).\bar{A}\\
&= A.\bar{A} + \bar{A}.\bar{B}\\
&= \bar{A}.\bar{B} \qquad (2)
\end{aligned}
$$

B. **Calculation for AB**

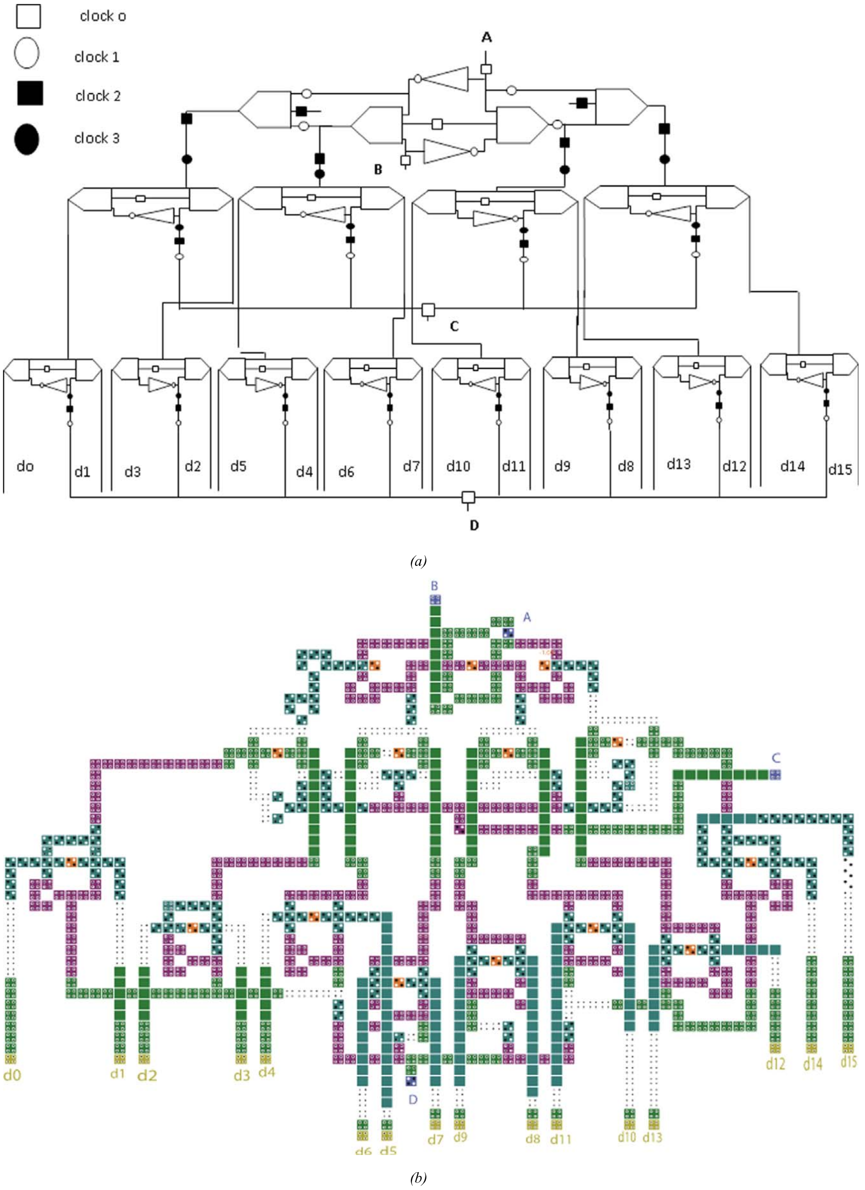
$$
\begin{aligned}
P &= (\overline{\bar{B}A}).A\\
&= (\bar{\bar{B}} + \bar{A}).A\\
&= (B + \bar{A}).A\\
&= A.\bar{A} + A.B\\
&= A.B \qquad (3)
\end{aligned}
$$

Table 3 and Fig. 3(a), show the truth table and the schematic diagram of the proposed addressed 2 × 4 Decoder. A total 4 clock zones or 1 clock cycle are required to synchronize the entire circuit. In the proposed design the outputs of any Majority Voter (MV) acts as the input to the next MV at the same clock zone. Thus the delay of the entire circuit has been reduced to a large extent.

Majority functions relevant to this implementation are given in Eqs. (4) and (5). The Majority gate representation and the QCA layout of the proposed circuit is given in Fig. 3(b) and (c) respectively.

$$
\begin{aligned}
R &= F(\bar{A},0,(\overline{\bar{A}B}))\\
&= \bar{A}.(A + B)\\
&= \bar{A}\bar{B} \qquad (4)
\end{aligned}
$$

$$
\begin{aligned}
S &= F(A,\ 0,\ (\overline{\bar{B}A}))\\
&= A.(A + \bar{B})\\
&= A.B \qquad (5)
\end{aligned}
$$



(a)

(b)

Fig. 5. (a) 3-input Majority gate implementation of 4 × 16 QCA decoder circuit. (b) QCA circuit for 4 × 16 QCA decoder circuit.

This 2 × 4 Decoder module has been extended to implement 3 × 8 decoder. AND operation is performed with the derived components from 2 × 4 decoder modules to achieve the outputs of the proposed 3 × 8 decoder. The third input as C has been brought into non-inverted and inverted forms through QCA wires to perform AND operations with each of the previously arrived components to derive corresponding outputs. Here, the wire crossings are needed only to extend the inputs and outputs of the circuit outside of the circuit. Total 8 Clock zones (2 clock cycles) have been maintained as per needed to synchronize the operations. Similar operations have been carried through to design the 4 × 16 counterpart as well. In this case and onwards crossovers could not be escaped. The corresponding Majority gate implementation and the QCA layout of the 3 × 8 decoder circuit is given in Fig. 4(a) and (b) respectively. And the same of the proposed 4 × 16 decoder is given in Fig. 5(a) and (b) respectively.

Lemma 1. *$C(n_e) \le C(n_m)$, $\forall\ n \ge 2$, where $C(n_e)$ is the number of cells in extendable approach and $C(n_m)$ is the cell count for modular approach.*

Proof. If the conventional methods which is being called as the modular approach, would have been followed, two 2 × 4 modules would have been interfaced with the third input to design the decoder of higher order version i.e. the 3 × 8 decoder. That method would have included double number of cells as in the 2 × 4 decoder with addition of number of cells needed for the AND modules. So, it can be deduced, if the total number of cells for decoder at n$^{\text{th}}$ level is considered to be $C(n)$ and the total number of cells for AND modules is $A(n_m)$, then:

$$C(n) = 2.\, C(n-1) + A(n_m) \tag{6}$$

Here, the wire crossings, if any, are ignored in the calculation. But in the addressed approach, only one decoder module of lower order, the AND modules as $A(n_e)$, and crossover cells are required to devise the higher modules. So, the formula, in this case becomes:

$$C(n) = C(n-1) + W(n) + A(n_e) \tag{7}$$

where $W(n)$ is the number of crossings needed at that stage.

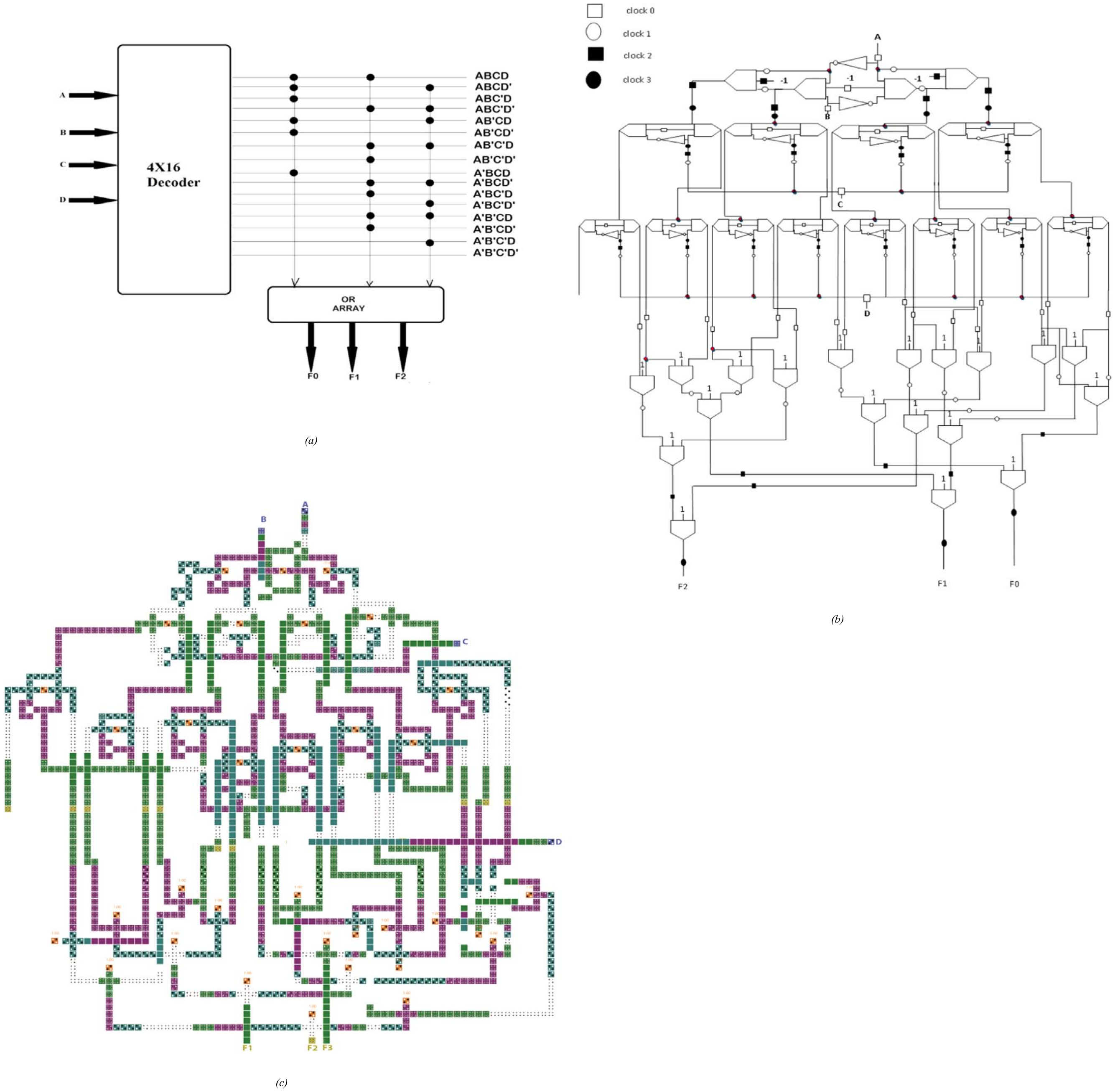


Fig. 6. (a) Block Diagram of PLA based adder. (b) Majority gate realization of decoder based PLA. *(c)* QCA Layout of the proposed PLA.

This formulae reveals that, even when wire crossings are not counted in conventional approach, the value of $C(n)$ in Eq. (7) is less than the value in Eq. (6) as $C(n-1) \gg W(n)$ in practice and $A(n_e) \equiv A(n_m)$. Eventually Eq. (6) is the amount of $C(n_m)$ and Eq. (7) gives the value of $C(n_e)$.

Lemma 2, which relates the number of crossovers, can be suggested as:

Lemma 2. *$F(n) = F(n-1) + 2^{n-1} - 2, \forall\, n > 3$, where $F(n)$ is the number of crossovers needed at nth stage and $F(3) = 0$.*

Proof. It is observed that the number of crossovers are affected only by the number of output lines from the previous decoder segment, i.e. in the nth stage the number of crossovers is dependent on the number of all restricted output lines to design the $(n-1)$th stage. Thus, the amount can be defined recursively as in Eq. (8):

$$F(n) = F(n-1) + 2^{n-1} \tag{8}$$

It can be found from the implementation that two outputs at two ends can be kept free from the crossovers as they are not bounded. So, at each stage essentially the number of needful crossovers would be reduced by 2. Hence, the formula can be revised as:

$$F(n) = F(n-1) + 2^{n-1} - 2 \tag{9}$$

As, the layouts are free from any kind of crossovers up to 3 × 8 version, it can be considered as the base condition of this recurrence relation. Thus the actual formula can be represented as:

$$F(n) = F(n-1) + 2^{n-1} - 2 \tag{10}$$

Table 4
Truth table of PLA adder.

Input				Output		
A	*B*	*C*	*D*	F_0	F_1	F_2
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0

Based on the truth table the connections are made to utilize the proposed PLA to act as a full adder. The block diagram is given in Fig. 6(a).

where, n is the input number and $F(n) = 0$; for $n = 3$.

Hence, it is concluded that, the number of crossovers is not only a exponential expression; rather it decreased by 2 at each step.

4.3. Design of Programmable logic array (PLA) using decoder

Programmable logic array consist of a decoder followed by a logical OR-array. The decoder generates product of input variables whereas the OR-array generates sum-of-product (SOP) expressions. It has M-inputs, n-product terms and N-outputs with $n < 2^M$, and can be used to implement a logic function of M-variables with N-outputs. Normally it can be said $2^M \times N$ PLA. If $I_0, I_1, . . , I_{M-1}$ are M-numbers of inputs to Decoder. The outputs of this AND-array are $P_0, P_1, . . , P_{M-1}$. It can be written as, $P_i = (I_0\overline{I_0}I_1\overline{I_1}... I_{M-1}\overline{I_{M-1}})$ where, $i=$ 0 to (n-1).

After the addition of OR-array the final output is, $f_k = \sum_{i=0}^{(n-1)} [\prod_{j=0}^{(M-1)} P_i = (I_j\overline{I_j})]$, where $k = 0$ to (N-1).

The block diagram of QCA based PLA is given in Fig. 6(a). The QCA based PLA consists of two planes namely 4 × 16 decoder and OR Array. The proposed PLA is utilized to design a 2-bit adder. Let F_0, F_1 and F_2 are the output of the OR Array. Truth table of PLA Full Adder it given in Table 4.

The truth table of Table 4 is realized in the block diagram of Fig. 6(a).The majority gate realization and the QCA layout of the proposed PLA is given in Fig. 6(b) and (c) respectively.

4.4. Algorithm for n-bit PLA circuit operation

An algorithm have been proposed to describe the operation N-bit PLA. In order to design an N-bit PLA a $N \times 2^N$ Decoder is needed. Here N=Number of decoder inputs, M=N_max, this is the maximum number of inputs to the Decoder. Furthermore K= Number of OR array and P=K_max, or the maximum number of the OR array that the designer will choose. With these conditions the algorithm for N-bit PLA has been presented.

```
Algorithm
PLA circuit _ design (N,M,K,P)
Begin
1.  Input M
2.  Make product term using AND logic gate
3.  Set N=1
4.      Check for N is less than M
5.          IF not Then go to step 8
6.                  Do for N=1 to M in step of 4 do
7.                       Calculate number of product term
8.                          Increment value of N
9.                          Go to step 4
10.                        End of 1st loop
11.                        K = 0
12.                    Make Sum term using OR logic gate
13.                Set K=1
14.              Check for K is less than P
15.          IF not Then go to step 18
16.      Do for K=1 to P in step of 14 do
17.    Calculate number of Sum term
18.  Increment value of K
19.Go to step 14
20.End of 2nd loop
21.Build [N * 2^N * k] circuit
End
```

The complexity of the proposed algorithm for N-Bit PLA is, $(2N \times 2^N + 2^N \times M + M)$.

5. Experimentation

The proposed design can be experimentally fabricated using both molecular QCA as well as Semiconductor QCA. Here the fabrication process of both types of QCA is briefly described.

5.1. Molecular QCA

In [38] Lent et. al produced a quantum dot at the redox centre of the 1,4-diallyl butane radical cation. Lent et al. also in [39] proposed a three dot molecule based on the same principle as in [38]. The molecule in [39] has three allyl groups connected in a "V" like structure by alkyl bridge. It represents a "QCA half-cell" which can be in the state '1', '0' or NULL. An unsymmetrical Ru-Fc complex QCA cell is prepared and synthesized in [40]. Further XPS and Spectrographic studies are performed to support the experimental observations.

5.2. Semiconductor QCA

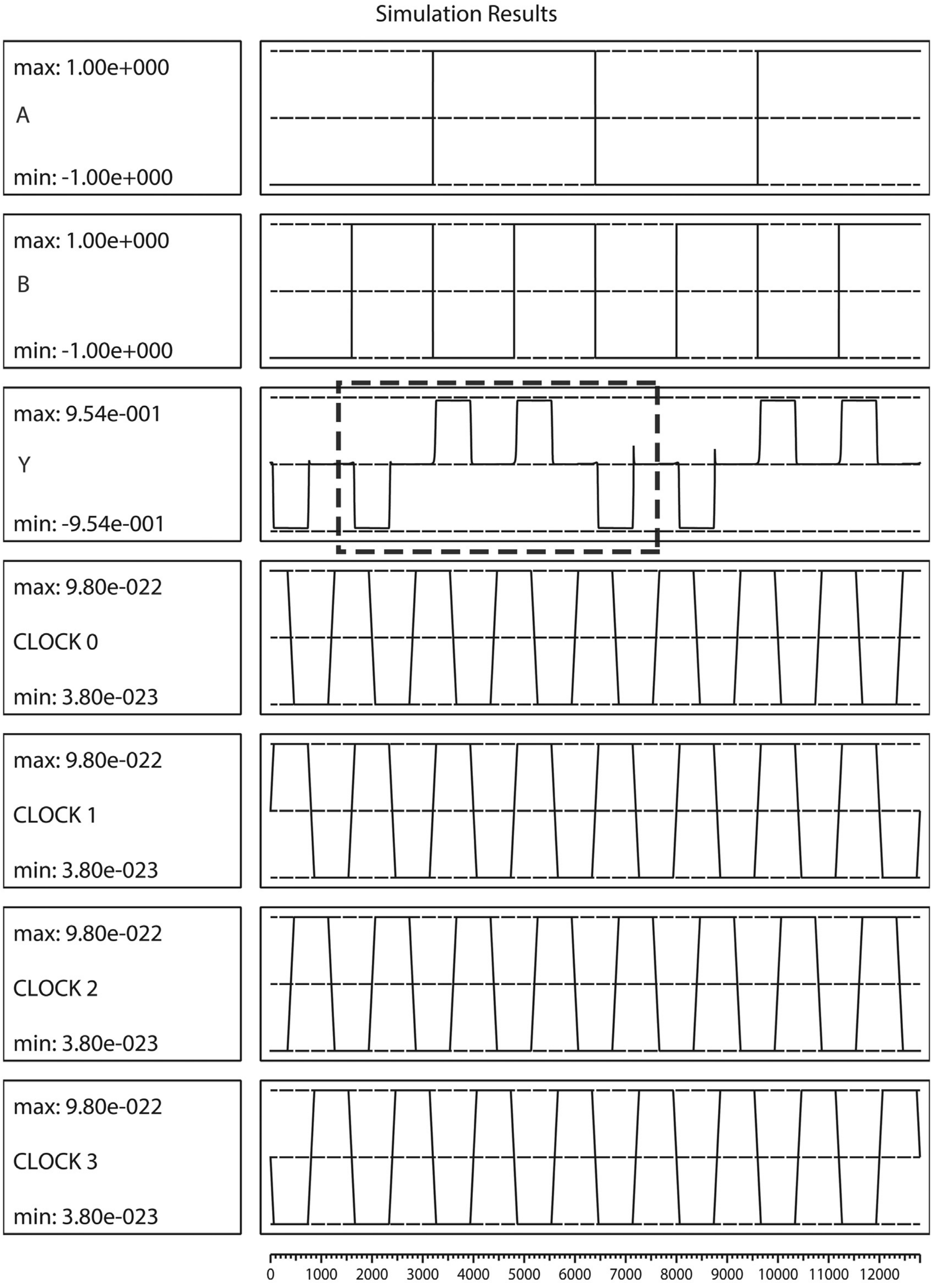
QCA cell is realized in Silicon system. In a heterojunction semiconductor material viz GaAs/AlGaAs four quantum dots can be fabricated with a high mobility two dimensional electron gas below the surface. The idea behind developing semiconductor QCA system is to pattern electrons confined in 2D Electron Gas (2DEG) using metal top gate. The 2DEG is formed at the interface of a semiconductor substrate and dielectric layer. Preferable semiconductor materials are Silicon-Silicon Dioxide or III–V Heterojunctions materials. Electric field is applied through the metal top gate which depletes the electrons in the 2DEG. Finally at places metal gates are etched away to form Quantum dots (QD) at the exposed surfaces.

6. Result and discussion

The outputs of the XOR and Decoders are shown in Fig. 7(a)–(d) respectively. Finally the output of the PLA full adder is presented in Fig. 7(e). Results of the proposed design for both of the XOR and 2 × 4 Decoder have been noticed to have a delay of 1clock cycle i.e. the output is obtained after 1clock cycle, since 4 clocking zones forms a clock cycle in QCA technology [1,2,8]. That is the output is taken at the fourth clock zone. The output of 3 × 8 and 4 × 16 decoders has 2 and 3 clock cycle delay respectively. The structures are free from any kind of QCA wire crossing, which eventually has appeared to be the key for reducing the number of cells and circuital delays. The figures reflect that the output pulses are quite stable. Complexity of the proposed designs is presented in Table 5.

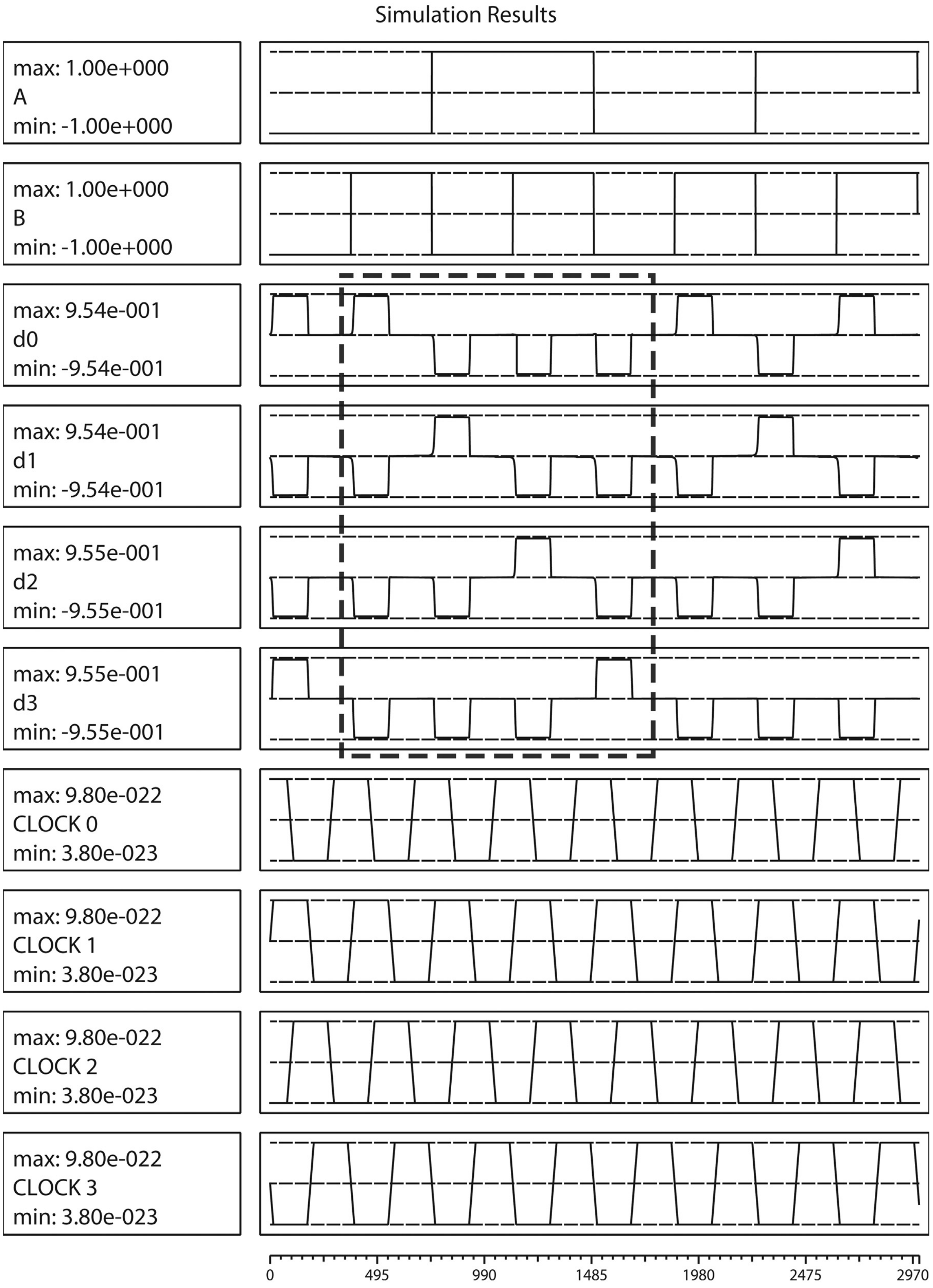
In Tables 5 and 6 comparison of the designs of all the decoders with that proposed in [12–17] with respect to Majority Voter count and Clock cycle delay. It is observed from Table 6 the proposed design of decoders have significant delay improvement.

Graphs are depicted according to the Eqs. (9) and (10) and the respective values, obtained from simulations. Fig. 8(a) reflects the number of crossovers in the Y-axis and the input lines in the X-



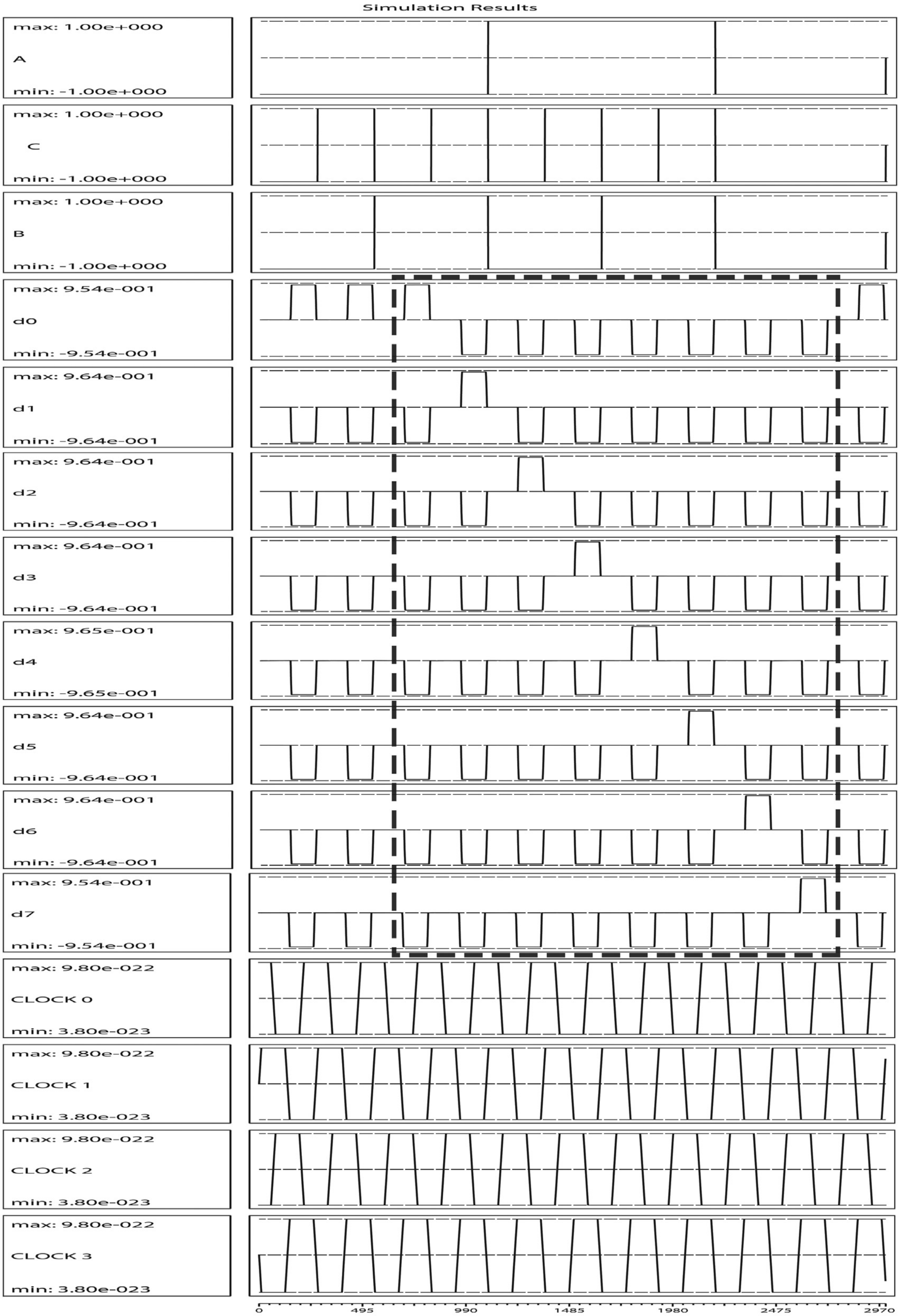
(a)

Fig. 7. (a) Output of XOR. (b) Output of 2 × 4 decoder. (c) Output of 3 × 8 decoder. (d) Output of 4 × 16 decoder. (e) Simulation result of PLA adder.



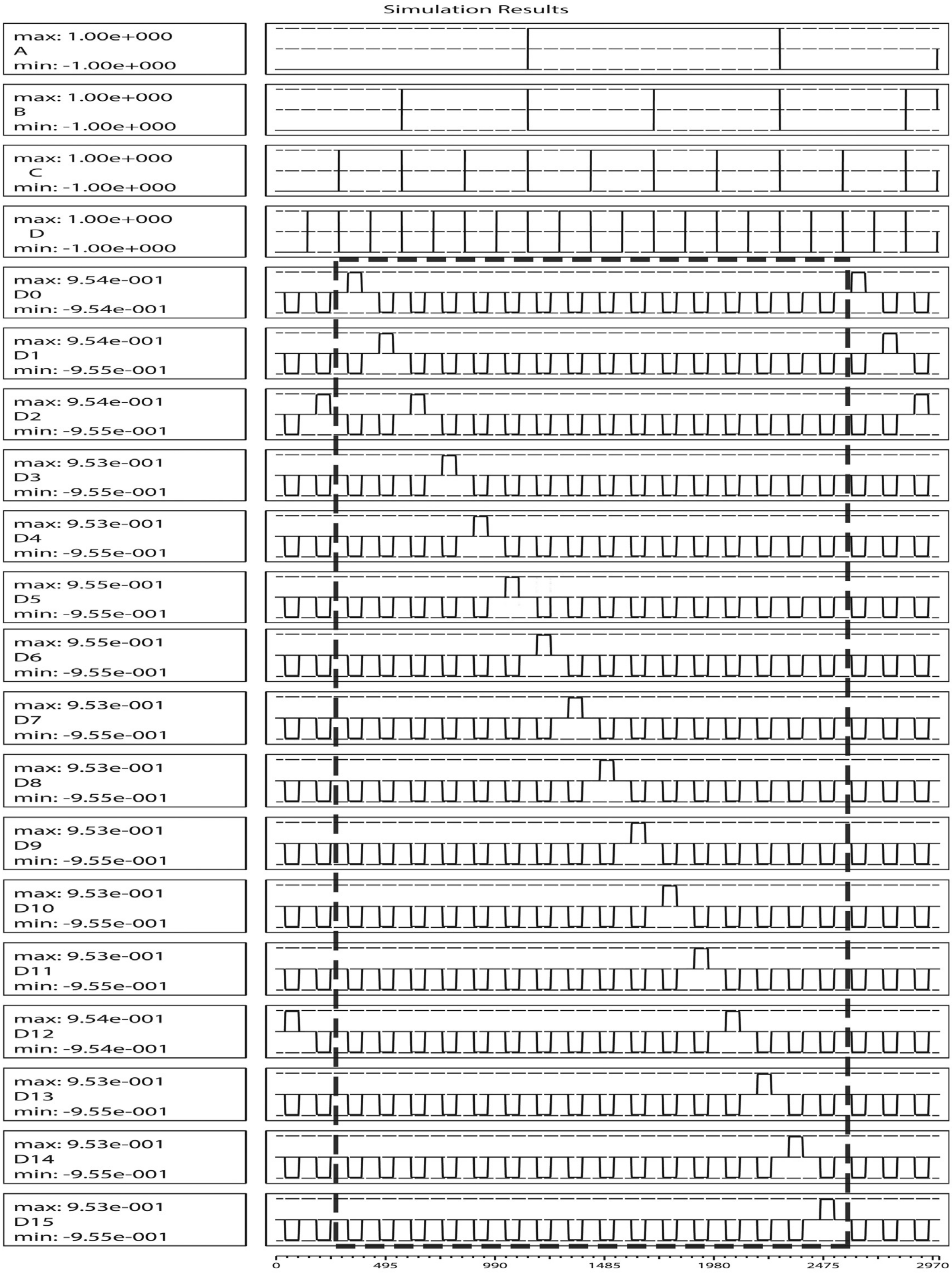
(b)

Fig. 7. (*continued*)



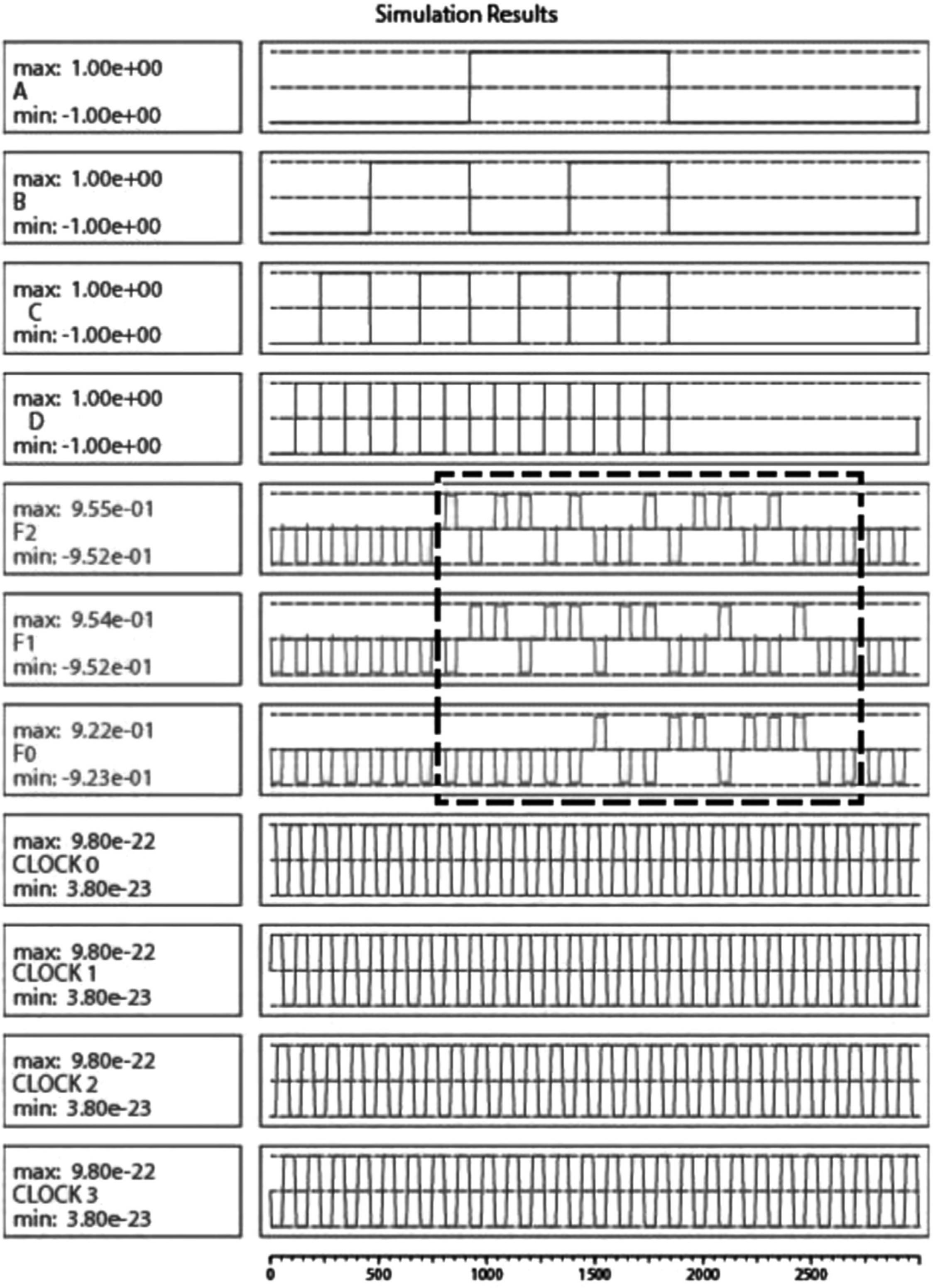
(c)

Fig. 7. (*continued*)



(d)

Fig. 7. (*continued*)



(e)

Fig. 7. (*continued*)

Table 5
Comparison of Proposed and Existing Method [17].

2 × 4 decoder							
	[12]	[17]	[35]	[16]	[14]	[13]	**Proposed**
Cell count	139	154	302	36	270 (3 MV) and 268 (5 MV)	–	**93**
Majority voter	–	–	–	–	8 (3 MV) and 4 (5 MV)	4	**4**
Clock cycle delay	1	–	–	–	1.75	7	**1**

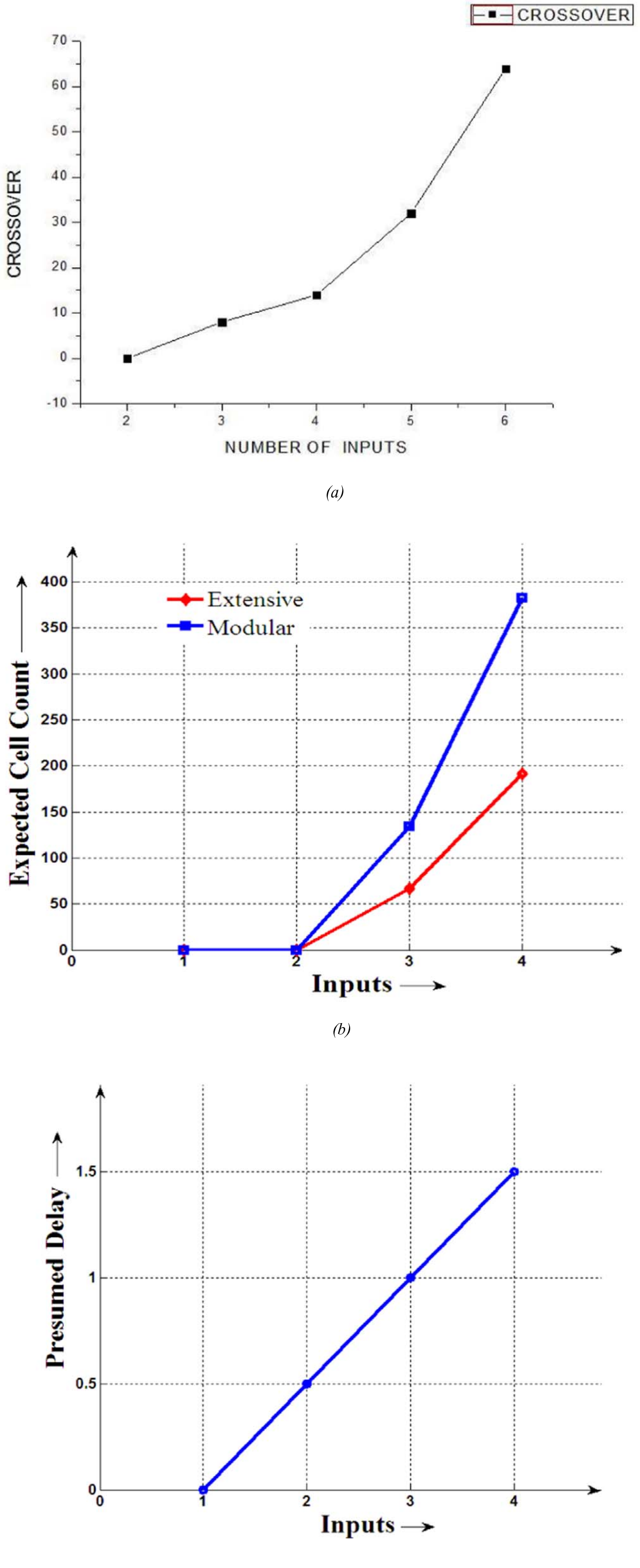
Table 6
Comparative study of clock cycle delay and the majority voter count.

Decoders proposed in [13]			**Decoders proposed in this paper**	
Decoder	Majority voters	Clock cycle delay	**Majority voters**	**Clock cycle delay**
2 × 4	4	7	**4**	**1**
3 × 8	8	11	**12**	**2**
4 × 16	20	15	**28**	**3**

axis. Fig. 8(b) substantiates the growth of expected number of cells with a comparison between two approaches. Here, observing the tested layouts it could be perceived that the $A(n)$ and $W(n)$ parts are equivalent in both of the methods, so only the C(n) part has been highlighted to plot the comparative study. It can be observed that the outputs $\overline{A}B$ and $A\overline{B}$ is arriving at the same clock phase where the inputs are given to the XOR circuit but they, however, have been read after a $\frac{1}{4}$ delay. But in the 2 × 4 decoder circuit, they arrive after a delay, this is done for synchronization and stability of the whole design which was not needed in case of XOR. In 3 × 8 decoder the delay is exactly 1 and the 4 × 16 circuit goes through 6 clock zones causing a delay of $1\frac{1}{2}$. Thus it can be presumed from the simulation data that the delay is supposed to increase linearly with respect to the cardinality of the circuit which is depicted in Fig. 8(c).

The results show that this method of extensibility could be helpful for implementing Decoder circuits of any cardinality in a more efficient way than the conventional approach where more than one lower decoder modules are interfaced to achieve higher modules.

For QCA, there is a minimum manufacturing width for a QCA wire which is the size of a QCA cell. There is also a minimum unit of time needed to transmit information in QCA wires, which is a clocking zone delay. The power dissipation should also be considered in QCA cost function designs. The cost metrics for QCA circuits need to be carefully chosen as these can significantly affect the choice of QCA circuit design. In a QCA circuit the number of cells in a QCA circuit is generally proportional to its area and including both would result in a double weighted area metric. Therefore either the number of cells or the area can be used as a rough measure of the complexity of a QCA circuit. The area of QCA designs is heavily dependent on types of crossovers. To measure the complexity of a QCA circuit, the numbers of logic gates and crossovers are better choice. The circuit complexity in QCA is actually the sum of the three primitives: majority gates, inverters and crossovers. Delay is always an important metric in assessing the performance of circuits. Therefore, the delay of a QCA circuit should be included in a cost function. For these reasons, the delay, number of logic gates, and number of crossovers are used to measure the performance, complexity, irreversible power dissipation, and the fabrication difficulty of a QCA circuit. Taking all this into consideration the authors in [37,41] have proposed a new QCA-specific cost function calculation. The number of Majority voters, Inverters and crossover are taken into account along with the circuital delay in order to calculate the QCA-specific cost. The



(a)

(b)

(c)

Fig. 8. (a) Expected growth of crossover with increasing input lines. (b) Expected Growth of Cell Count. (c) Presumed delay with increasing circuit cardinality.

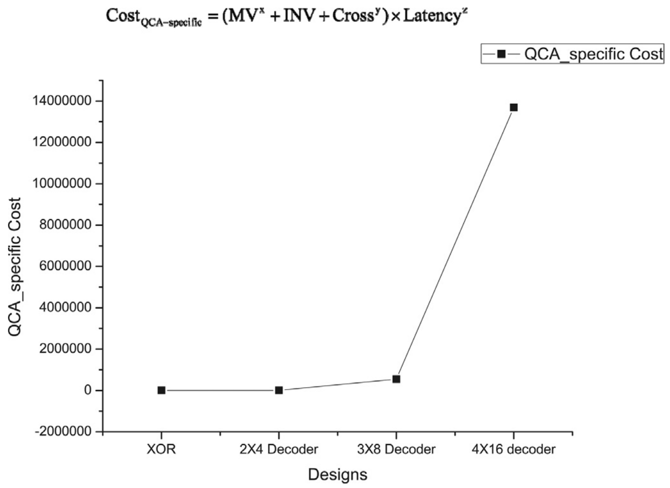


Fig. 9. QCA-specific cost calculation of XOR, 2 × 4, 3 × 8 and 4 × 16 decoders.

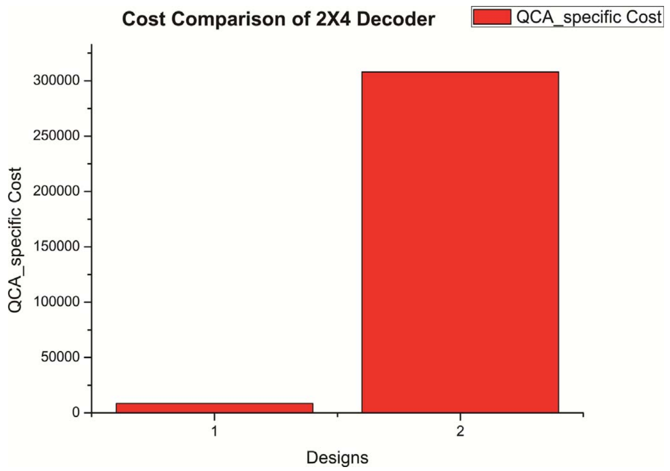


Fig. 10. Cost comparison of 2 × 4 decoder; 1 denotes the proposed decoder and 2 is the decoder in [13].

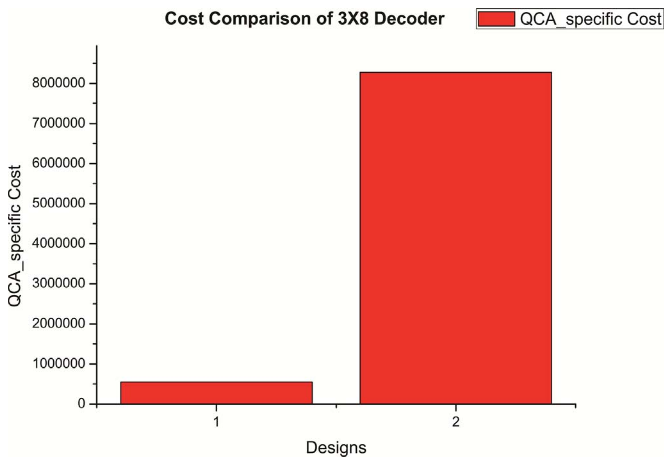


Fig. 11. Cost comparison of 3 × 8 decoder; 1 denotes the proposed decoder and 2 is the decoder in [13].

equation for the cost function calculation is given at Eq. (1). Here MV, INV, Cross denotes the number of Majority Voter, Inverters and crossovers respectively. The coefficient x, y and z are required for power dissipation calculation [37,38]. The values of x, y and z are kept constant at 2 [37]. Based on this method the Cost estimation of the proposed XOR Gate, 2 × 4, 3 × 8 and 4 × 16 decoders have been calculated. The corresponding values are given in Fig. 9.

$$Cost_{QCA-specific} = (MV^x + INV + Cross^y) \times Latency^z.$$

Furthermore a comparative study of the cost function of proposed 2 × 4 and 3 × 8 have been performed with that in [13]. The comparative study provided in Figs. 10 and 11 highlights the extent of the cost optimization that has been achieved in this design. Fig. 2 shows the cost comparison of 2 × 4 decoder proposed in this paper to that in [13]. Similar cost comparison of 3 × 8 decoder to that in [13] is also performed and shown in Fig. 11. The Comparative study is also represented in tabular form in Table 7.

Table 7
Cost calculation of Proposed design and that in [13].

Proposed				[13]	
XOR	2 × 4	3 × 8	4 × 16	2 × 4	3 × 8
3855	8669	551,172	13,689,891	308,061	8,274,736.25

7. Conclusion

In this paper an optimized and efficient design of XOR and Decoders are implemented using QCA. The design of the decoders by repeated use of the XOR gate helps to reduce the cell count to a large extent .The design can be fabricated and it gets rid of the stability issue which might have raised if 45 degree cells were used. Furthermore it is observed from the comparative studies that the decoders designed here have significant delay, cell-count and Majority Voter count benefits compared to existing designs. Finally the 4 × 16 decoder is utilized to design a decoder based PLA circuit. The PLA can be used to perform various logic operations; here a 2-bit adder has been designed. The work can be extended to design a Configurable Logic Block which will be highly delay efficient design.

Acknowledgement

Authors T. Purkayastha and D. De are grateful to The University Grants Commission, India, for providing with the grant for accomplishment of the project under the UGC Major Project File no. 41-631/2012(SR).

References

[1] P.D. Tougaw, C.S. Lent, Logical devices implemented using quantum cellular automata, J. Appl. Phys. 75 (3) (1994) 1818–1825.
[2] C.S. Lent, P.D. Tougaw, W. Porod, G.H. Bernstein, Quantum cellular automata, Nanotechnology 4 (1) (1993) 49.
[3] G.A. DiLabio, R.A. Wolkow, J.L. Pitters, P.G. Piva, U.S. Patent No. 8,816,479, Washington, DC: U.S. Patent and Trademark Office, 2014.
[4] H. Cho, E.E. Swartzlander, Adder designs and analyses for quantum-dot cellular automata, IEEE Trans. Nanotechnol. 6 (3) (2007) 374–383.
[5] A. Chaudhary, D.Z. Chen, X.S. Hu, M.T. Niemier, R. Ravichandran, K. Whitton, Fabricatable interconnect and molecular QCA circuits, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 26 (11) (2007) 1978–1991.
[6] D. Rajeswari, K. Paul, M. Balakrishnan, M., Clocking-based coplanar wire crossing scheme for QCA, in: Proceedings of the 23rd International Conference on VLSI Design, 2010, VLSID'10, IEEE, 2010, January, pp. 339–344.
[7] L. Lu, W. Liu, M. O'Neill, E.E. Swartzlander, QCA systolic array design, IEEE Trans. Comput. 62 (3) (2013) 548–560.
[8] Huanqing Cui, et al., Design of non-restoring binary array divider in quantum-dot cellular automata, Micro Nano Lett. IET 9.7 (2014) 464–467.
[9] D. Abedi, G. Jaberipur, M. Sangsefidi, Coplanar Full Adder in Quantum-Dot Cellular Automata via Clock-Zone Based Crossover, 2015.
[10] P. Wang, M. Niamat, S. Vemuru, M. Alam, T. Killian, Synthesis of Majority/Minority Logic Networks, 2015.

[11] Sandhya Hema Jagarlamudi, Jagarlamudi Pavan Kumar Saha Mousumi, Quantum dot cellular automata based effective design of combinational and sequential logical structures, World Acad. Sci. Eng. Technol. 60 (2011).

[12] R. Zhou, X. Xia, F. Wang, Y. Shi, H. Liao, A logic circuit design of 2–4 decoder using quantum cellular automata, J. Comput. Inf. Syst. 8.8 (2012) 3463–3469.

[13] Moein Kianpour, Sabbaghi-Nadooshan Reza, A Novel Modular Decoder Implementation in Quantum-dot Cellular Automata (QCA), in: Proceedings of the International Conference on Nanoscience, Technology and Societal Implications (NSTSI), IEEE, 2011, pp. 1–5.

[14] Moein Kianpour, Sabbaghi-Nadooshan Reza, A conventional design for CLB implementation of a FPGA in quantum-dot cellular automata (QCA), in: Proceedings of the 2012 IEEE/ACM International Symposium onNanoscale Architectures (NANOARCH), IEEE, 2012, pp. 36-42.

[15] Reza Sabbaghi-Nadooshan, A novel quantum-dot cellular automata CLB of FPGA, J. Comput. Electron. 13 (3) (2014) 709–725.

[16] C. Mukherjee, S. Pramanik, R., Chakraborty, D. De, Decoder segment optimization of ROM design in quantum dot cellular automata. in: Proceedings of the 2014 International Conference on High Performance Computing and Applications (ICHPCA), IEEE, 2014, pp. 1–5.

[17] Vetteth Anoop, Konrad Walus, Vassil S. Dimitrov, Graham A. Jullien, Quantum-dot cellular automata carry-look-ahead adder and barrel shifter, in: Proceedings of the IEEE Emerging Telecommunications Technologies Conference, 2002, pp. 2-4.

[18] Das Kunal, De Debashis, A study on diverse nanostructure for implementing logic gate design for QCA, Int. J. Nanosci. 10 (1–2) (2011) 263–269.

[19] Das Kunal, De Debashis, Characterisation, applicability and defect analysis for tiles nanostructure of quantum dot cellular automata, Mol. Simul. 37 (03) (2011) 210–225.

[20] Das Kunal, De Debashis, Novel approach to design a testable conservative logic gate for QCA implementation, in: Proceedings of the 2010 IEEE 2nd International Advance Computing Conference (IACC), IEEE, 2010.

[21] Das Kunal, De Debashis, Characterization, test and logic synthesis of novel conservative and reversible logic gates for Qca, Int. J. Nanosci. 9 (3) (2010) 201–214.

[22] A. Dey, k Das, D. De, M. De, Probabilistic defect analysis model for quantum dot cellular automata design at analytical phase, Int. J. Comput. Appl. 55 (7) (2012) 33–41.

[23] S. Sayedsalehi, A. Roohi, K. Navi, A different design approach for high performance in nanostructure using Quantum Cellular Automata, Can. J. Electr. Electron. Eng. 2 (11) (2011) 526–530.

[24] Jin Wook Chung, Kyu Lim Sung Smith Brian, QCA physical design with crossing minimization, in: Proceedings of the 5th IEEE Conference on Nanotechnology, IEEE, vol. 200, 2005.

[25] Mohsen M. Arjmand, Mohsen Soryani, Keivan Navi, Coplanar wire crossing in quantum cellular automata using a ternary cell, Circuits Dev. Syst. IET 7 (5) (2013).

[26] R. Devadoss, K. Paul, M. Balakrishnan, Coplanar QCA crossovers, Electron. Lett. 45 (24) (2009) 1234, http://dx.doi.org/10.1049/el..2819, 1235.

[27] Dominic A. Antonelli, Danny Z. Chen, Timothy J. Dysart, Xiaobo S. Hu, Andrew B. Kahng, Peter M. Kogge, Richard C. Murphy, Michael T. Niemier., Quantum-dot cellular automata (QCA) circuit partitioning: problem modeling and solutions, in: Proceedings of the 41st Annual Design Automation Conference, ACM, 2004, pp. 363–368.

[28] A. Pulimeno, M. Graziano, A. Sanginario, V. Cauda, D. Demarchi, G. Piccinini, Bis-ferrocene molecular QCA wire: Ab initio simulations of fabrication driven fault tolerance, IEEE Trans. Nanotechnol. 12 (4) (2013) 498–507.

[29] Rafiq Mohammad Beigh, Ahmad Firdous Mustafa Mohammad, Performance Evaluation of Efficient XOR Structures in Quantum-Dot Cellular Automata (QCA), 2013.

[30] E.N. Ganesh, Implementation of programmable logic devices in quantum cellular AUTOMATA technology, GESJ: Comput. Sci. Telecommun. 2 (2011) 31.

[31] K. Walus, A. Vetteth, G.A. Jullien, V.S. Dimitrov, RAM design using quantum-dot cellular automata, Nanotechnol. Conf. 2 (2003).

[32] M. Malekpour, R. Sabbaghi-Nadooshan, A Novel Simulation of Decoders using HDLQ in QCA, 2011, May.".

[33] Walus Konrad, Timothy J. Dysart, Graham A. Jullien, R. Arief Budiman, QCA-Designer: a rapid design and simulation tool for quantum-dot cellular automata, IEEE Trans. Nanotechnol. 3 (1) (2004) 26–31.

[34] L.H. Sardinha, A.M. Costa, O.P.V. Neto, L.F. Vieira, M. Vieira, Nanorouter: a quantum-dot cellular automata design, IEEE J. Sel. Areas Commun. 31 (12) (2013) 825–834.

[35] M. Malekpour, R. Sabbaghi-Nadooshan, A novel simulation of decoders using HDLQ in QCA, Can. J. Electr. Electron. Eng. 3 (7) (2012).

[36] W. Liu, L. Lu, M. O'Neil, E.E. Swartlander Jr, Design rules for quantum-dot cellular automata, in: Proceedings of the IEEE International Symposium on Circuits & Systems (ISCAS), IEEE, 2011, pp 2361–2364.

[37] D. Abedi, G. Jaberipur, M. Sangsefidi, Coplanar Full Adder in Quantum-Dot Cellular Automata via Clock-Zone Based Crossover, 2015.

[38] C.S. Lent, I. Beth, L. Marya, Molecular quantum-dot cellular automata, J. Am. Chem. Soc. 125 (4) (2003) 1056–1063.

[39] C.S. Lent, I. Beth, Clocked molecular quantum-dot cellular automata, IEEE Trans. Electron Dev. 50 (9) (2003) 1890–1896.

[40] Z. Li, A.M. Beatty, T.P. Fehlner, Molecular QCA cells. 1. Structure and functionalization of an unsymmetrical dinuclear mixed-valence complex for surface binding, Inorg. Chem. 42 (18) (2003) 5707–5714.

[41] W. Liu, L. Lu, M. O'Neill, E.E. Swartzlander, A first step toward cost functions for quantum-dot cellular automata designs, IEEE Trans. Nanotechnol. 13 (3) (2014) 476–487.