

Design of an Optimal Decimal Adder in Quantum Dot Cellular Automata

B. Ramesh¹ and M. Asha Rani^{2*}

¹*Department of Electronics and Communication Engineering,
Kamala Institute of Technology & Science, Huzurabad 505468, India.*

²*Department of Electronics and Communication Engineering,
JNTUH College of Engineering, Hyderabad, 500085, India.*

Abstract

Integrated Circuit (IC) Technology is growing day by day to enhance the circuit performance and to increase the density for compact systems. Conventional CMOS technology is playing a vital role in digital computation for past four decades. But there are certain challenges in scaling the CMOS devices for the last few years. A novel technology “Quantum dot Cellular Automata (QCA)” has been identified as an upcoming nanoelectronic technology to overcome the limitations in the transistor based CMOS technology. In order to bring a new concept of IC design in an efficient and optimized manner; it is proposed here an efficient architecture for BCD adder in QCA technology using an area optimized QCA full adder. The proposed BCD adder circuit optimize the QCA design metrics such as cost function, layout area and number of QCA cells in comparison with the existing BCD adder designs.

Keywords: BCD adder, Quantum dot Cellular Automata, majority gate, wire crossover, QCA cost function

1. Introduction

The performance and circuit density of IC technology is growing successfully with CMOS devices for the past four decades, but at nanometer scale these are facing new

challenges are short channel effects like drain-induced barrier lowering, velocity saturation, punch-through, hot carrier effects, sub-threshold leakage currents etc. [1]. Hence some alternative nanoelectronic technologies have been proposed by the researchers to invent a novel nanoelectronic technology which can work at nanometer dimensions. To overcome the scaling limitations of the current CMOS technology, the International Technology Roadmap for Semiconductors (ITRS) has identified some new technologies to replace the existing transistor based technology in future. The ITRS identified nanoelectronic technologies are Resonant Tunneling Diodes (RTD), Quantum dot Cellular Automata (QCA), Single Electron Transistor (SET), Spintronics and Carbon Nano Tubes (CNT). In all these technologies QCA seems to be the promising novel computing technology for an alternative to current CMOS technology [2].

Craig S. Lent et al, was proposed QCA technology in the year 1993 [3] as an alternative nanoelectronic technology to replace conventional transistor based CMOS technology. The QCA design is based on quantum dots which are more suitable for logic circuits and has significant advantages, high speed, very high density and low power consumption [4], [5]. Therefore, it is considered to be an attractive nanoelectronic technology for the development of digital circuits. Efficient QCA circuits which include binary adders [6], [7] have been designed and verified using QCA simulation tool QCADesigner [8]. Design methods [9] and performance evolution functions such as QCA design metrics; area, delay and cost functions [10] for complex QCA designs have been clearly explored to achieve more efficient designs.

Decimal arithmetic operation plays a vital role in commercial, financial and other applications. So there is a need of designing very efficient decimal adders and that requires proper design strategies at logic level as well as at layout design level to improve the performance and circuit density. To improve the circuit performance and optimize the layout area different BCD adders in the QCA technology were presented [11-14].

This paper proposes an efficient architecture for QCA based BCD adder using an area optimized QCA full adder, and which is able to achieve very good improvement in the QCA design metrics in comparison with their existing counterparts. The results obtained in the proposed design for n-digit ($n \geq 1$) BCD adders are compared to the QCA design metrics such as area, cell count, clock cycles and QCA cost functions [10].

The rest of this paper is organized as; section 2 briefly covers QCA technology introduction, fabrication methods, tunnel junction capacitances, wire crossovers and QCA design metrics. Section 3 covers about the designs of previous BCD adders. The proposed optimum BCD adders are presented in Section 4. The Section 5 covers comparison of QCA design metrics of proposed design with its existing counterparts. Finally the conclusions are presented in the section 6.

2. QCA Background

2.1. Quantum dots, QCA cells

Quantum dot, a basic component in QCA technology, is a single electron container. A QCA cell is a basic computing device in QCA, which is a square nanostructure made up of four quantum dots normally placed at the four corners of the cell. The QCA cell is packed with two electrons, and they occupy diagonally opposite dots in the cell due to columbic repulsion between the electrons. The electrons in a cell can tunnel only between the dots within a cell, but they cannot tunnel between the cells because of the high potential barriers between the cells [3] that results very low power consumption. Two types of QCA cell polarizations are defined based on the two ground state electron configurations of a cell. Two charge configurations of a QCA cell represent two binary states '0' and '1' with polarizations of '-1' and '+1' respectively. The states of a QCA cell polarization value, $P = -1$ represents binary '0' and polarization value, $P = +1$ represents binary '1' are schematically shown in Fig. 1(a). The dynamic power in QCA is due polarization change of cell, that happens only with electron tunneling but not with electron flow, so its dynamic power consumption very low even for complex circuits. An array arrangement of QCA cells in horizontal or vertical form represents a QCA wire, which is used to transfer one bit of binary data from one end to the other end as shown in Fig. 1(b).

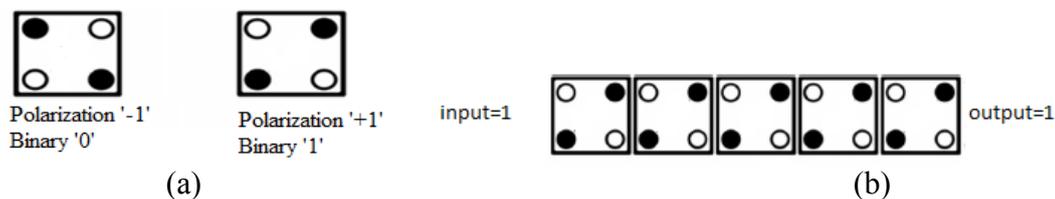


Fig. 1. Schematics of QCA cells and wire: (a) QCA cells with polarization '-1', '+1', (b) QCA wire.

In QCA the basic logic building blocks are inverters and three-input majority gates. A QCA inverter is designed by placing the cells diagonally from each other. There are two types of inverters, one with four cells and another with seven cells as shown in Fig. 2(a) and Fig. 2(b) respectively. An inverter can be used in the interconnections and they do not cause any additional delays [10]. A three-input QCA majority gate consists of five cells, in which three input cells, one evaluation cell and one output cell with the boolean function $Y = M(A, B, C) = AB + BC + CA$, as shown in Fig. 2(c). For A, B, C are binary inputs the output of a majority gate is logic '1' if the majority of inputs are '1' and logic '0' if the majority of the inputs are '0'. A two-input OR gate and AND gate can be implemented by applying any one of the inputs of majority gate to '1' or '0' respectively. Logic functions can be implemented using inverters, AND, OR and majority gates.

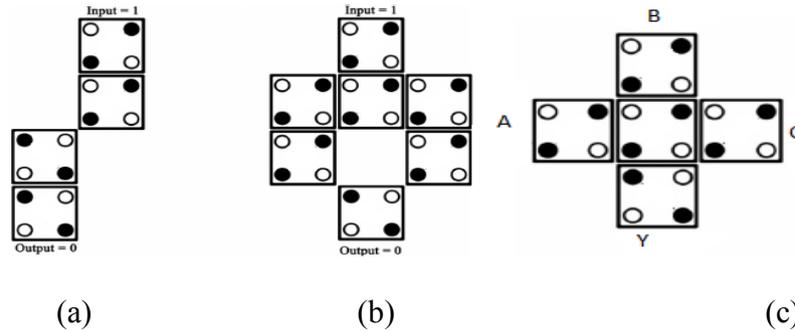


Fig. 2. Schematics Of QCA Inverters, Majority Gate: (a) Four Cell Inverter, (b) Seven Cell Inverter, (c) Majority Gate.

2.2. QCA fabrication methods

In QCA quantum dots are implemented in three ways. 1) Metal-island quantum dots [15]; 2) Semiconductor quantum dots [16] and 3) Molecular dots [17], the Metal-Island based quantum dot was first fabricated device to demonstrate QCA cell. In this method quantum dot was built as an Aluminum Island. These experiments were done with $1\mu\text{m}$ metal-island. The Semiconductor dots could be used to fabricate QCA circuits using the CMOS device fabrication methods. However, the current CMOS fabrication methods have not yet reached to nano-scale. A proposed, but not yet quantum dot fabricated method is a single molecule quantum dot to implement the QCA circuits can have lot of advantages such as 1) symmetric cell structure; 2) very high density; 3) high switching speed; 4) possibility of self assembly and 5) room temperature operation. In QCA circuit fabrication still there are some technical challenges, selection of molecule, providing interface between the cells and organization of clocking mechanism.

2.3. Clocking Scheme in QCA designs

There are four clocking phases in QCA designs, namely SWITCH, HOLD, RELEASE and RELAX with each clocking phase is shift by 90° from its adjacent clocking phase as depicted in Fig. 3. The clock signals in QCA designs are generated by applying an electric field to the QCA cells to enhance the tunneling barriers between the dots in that cell. In a QCA cell information transition takes place during the SWITCH phase. During the clock HOLD phase a QCA cell is latched. In RELEASE phase, barriers becomes low and cell become unpolarized and barriers remain at low during the relax phase [7].

2.4. Wire crossovers in QCA

Two types of wire crossovers available in QCA [10], one is coplanar crossover and second one is multilayer crossover. The coplanar crossovers are designed using four clock zones in QCA cells, they are clock 0, clock 1, clock 2 and clock 3, and each

clock zone has a phase difference of 90° . The cells with clock 0 and clock 2 have a phase shift of 180° and the intersection of these cells can form a coplanar wire crossover. Similarly cells with clock 1 and clock 3 have a phase shift of 180° can form a wire crossover as depicted in Fig. 4(a). A multilayer crossover is design using four QCA layers as depicted in Fig. 4(b). The type of crossovers used in circuits will influence the complexity of the design, because the fabrication cost of a multilayer crossover is three times of the cost of a coplanar crossover [10].

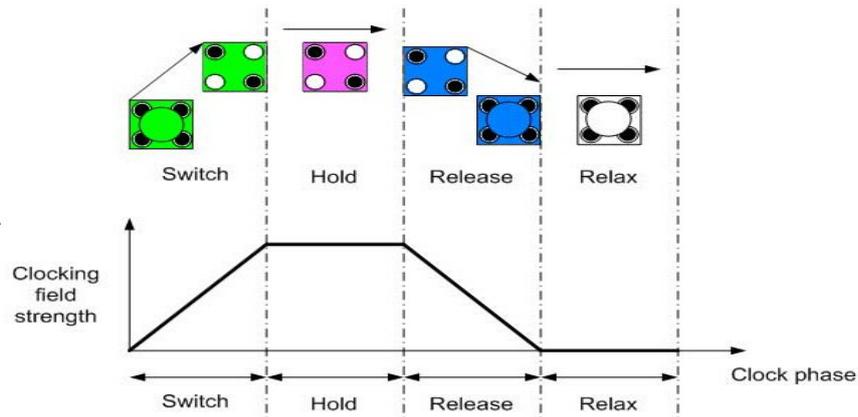


Fig. 3 Clocking zones in the QCA

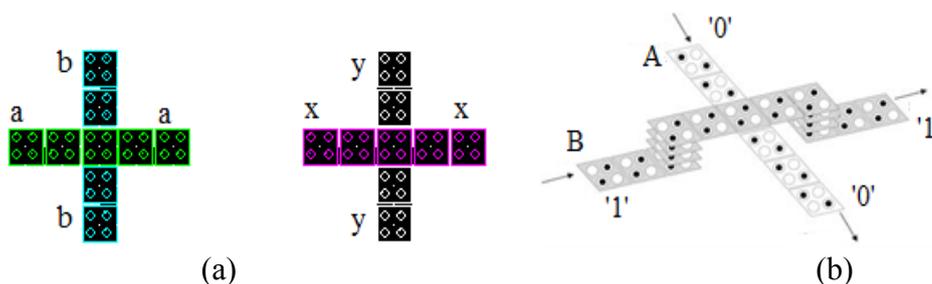


Fig. 4 QCA wire crossovers: (a) clock zone based coplanar crossovers, (b) multilayer crossover.

2.5.QCA Design Metrics

The complexity and performance of CMOS technology is defined with the design metrics delay, power dissipation and area. The power-speed product (product of power dissipation and delay in number of clock cycles) of a CMOS circuit gives its Figure of merit. Similar to CMOS design metrics QCA design metrics are number of clock cycles, majority gates, inverters, wire crossovers, QCA cells and area of the QCA layout design [10]. The number of QCA cells in this technology is similar to the number of MOS transistors in the CMOS technology. In QCA designs, both the logic elements and the interconnections are composed of QCA cells. As a result the QCA layout area is generally proportional to the number of cells in the layout [7]. The delay is a very important design metric in any technology; it represents the performance of the technology. The delay in CMOS circuit is the number of clock cycles in a critical path. In QCA technology the minimum delay is one clocking zone delay, it is $\frac{1}{4}$ of a

QCA clock cycle and the delay in a QCA circuit is defined as the number of zones in a critical path of the circuit [10]. The power consumption is an important factor in modern integrated circuit technologies. In QCA circuits electrons can only tunnel between the dots within a cell, but not between the cells, because of high potential barriers between the cells. As a result the power dissipation in QCA circuits is very small, but still it could incur some thermal problems due to very high circuit densities. The power dissipation in QCA circuits is defined based on the number QCA cells in the circuit [10]. Another important design metric in QCA circuits is number of wire crossovers. A coplanar crossover runs in a single layer and multilayer crossover can be designed with three or four QCA layers. The fabrication of multilayer crossover is extremely complex then the coplanar crossover. A QCA cost function like power speed-product in a CMOS circuit is defined based on the number of majority gates, number of inverters, number of wire crossovers and numbers clock cycles in a critical path. A generalized form of QCA function is given in equation (1) [10].

$$QCA_{\text{cost}} = (M^k + I + C^l) * T^p, \quad 1 \leq k, l, p \quad (1)$$

Where, M is number of majority gates, I is number of inverters, C is number of crossovers ($C_{ml} = m \times C_{cp}$, where C_{ml} is number of multilayer crossovers, C_{cp} is number of coplanar crossovers, m assumed to be 3 for the cost difference between multilayer and coplanar crossovers), T is the delay in number of clock cycles, k, l, p are exponent weights of the majority gates, crossovers and delay respectively. The values of k, l are selected as 2 and $p = 2$ when circuit complexity is the criteria, $p = 4$ when delay is the criteria [10].

3. Related Works

The QCA BCD adders proposed earlier in [12], [13] use a top level architecture shown in Fig. 5. In this first block adds two input numbers in binary format, carry is evaluated by a logic function in sum, carry output of the 4-bit adder block and then perform conversion from binary result to BCD number by adding 6 to the result if the binary sum in the first block is greater than 9, otherwise 0. A ripple carry adder is used to implement binary addition in [13], whereas two types of implementations presented in [12] make use of a carry flow adder [9] and a carry-look-ahead adder [7]. A serial decimal adder demonstrated in [11] using the Johnson-Mobius Code (JMC).

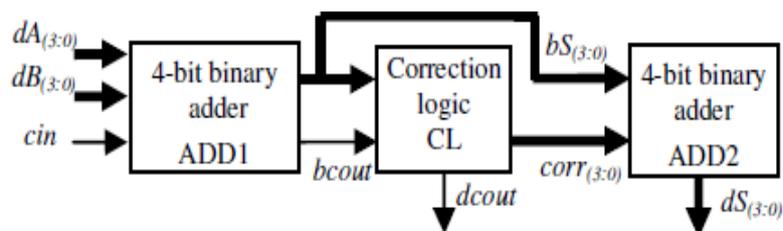


Fig. 5 Block diagram of 1-digit BCD adder.

The efficient BCD adders presented in [14] is a very recent design of BCD adders. Two types of implementations were presented for the architecture given in the Fig. 5. The ADD1 receives digits $dA_{(3:0)}$, $dB_{(3:0)}$ and carry cin are inputs and produce binary outputs $bcout$, $bS_{(3:0)}$. The logic expressions used to find carry with carry generate and propagate are $g_i = dA_i \cdot dB_i$ and $p_i = dA_i + dB_i$. The majority gate expressions of carry C_{i+1} , and C_{i+2} in terms of g_i , p_i and inputs are given in (2) and (3) respectively.

$$C_{i+1} = M(dA_i, dB_i, dC_i) \quad (2)$$

$$C_{i+2} = M(C_i, M(dA_{i+1}, dB_{i+1}, g_i), M(dA_{i+1}, dB_{i+1}, p_i)) \quad (3)$$

where C_{i+1} is produced from C_i through one majority gate given in (2), and C_{i+2} would require 2 cascaded majority gates given in (3). To reduce delay the majority gate function for C_{i+2} was given in terms of only inputs, without using g_i and p_i is given in (4).

$$C_{i+2} = M(C_i, M(dA_{i+1}, dB_{i+1}, dA_i), M(dA_{i+1}, dB_{i+1}, dB_i)) \quad (4)$$

By applying the conventional Carry Look-Ahead (CLA) logic the carry generate and propagate signals G_{i+1} , G_i , P_{i+1} and P_i are defined as $G_{i+1} = dA_{i+1} \cdot dB_{i+1}$, $G_i = dA_i \cdot dB_i$, $P_{i+1} = dA_{i+1} + dB_{i+1}$ and $P_i = dA_i + dB_i$, we have

$$C_{i+2} = G_{i+1} + P_{i+1} \cdot G_i + P_{i+1} \cdot P_i \cdot C_i$$

Its majority gate function is expressed in (4)

The majority gate functions of carries C_1 , C_2 , C_3 and C_4 are given in (5).

$$C_1 = M(dA_0, dB_0, cin) \quad (5)$$

$$\begin{aligned} C_2 &= G_1 + P_1 \cdot G_0 + P_1 \cdot P_0 \cdot cin \\ &= M(cin, M(dA_1, dB_1, dA_0), M(dA_1, dB_1, dB_0)) \end{aligned}$$

$$C_3 = M(dA_2, dB_2, C_2)$$

$$\begin{aligned} C_4 &= dcout = G_3 + P_3 \cdot G_2 + P_3 \cdot P_2 \cdot C_2 \\ &= M(C_2, M(dA_3, dB_3, dA_2), M(dA_3, dB_3, dB_2)) \end{aligned}$$

The majority gate functions for binary sum $bS_{(3:0)}$ was defined by applying carries and inputs are given in (6).

$$bS_i = M(C_i, \overline{C_{i+1}}, \overline{M(dA_i, dB_i, C_{i+1})}) \quad (6)$$

The majority gate functions to covert binary carry $bcout$ decimal carry $dcout$ and binary sum $bS_{(3:0)}$ to decimal sum $dS_{(3:0)}$ and are given in (7) and (8) respectively.

$$dcout = M(bcout, M(1, bcout, bS_3), M(bS_3, bS_2, bS_1)) \quad (7)$$

$$dS_0 = bS_0 \quad (8)$$

$$\begin{aligned}
dS_1 &= M(\overline{M(0, bS_1, dcout)}, M(1, bS_1, dcout), 0) \\
dS_2 &= M(\overline{M(bS_2, dcout, M(0, bS_1, dcout))}, \\
&M(M(bS_2, dcout, \overline{M(dcout, bS_2, M(0, bS_1, dcout))}), M(0, bS_1, dcout)) \\
dS_3 &= M(1, M(\overline{dcout}, bS_3, 0), M(M(dcout, \overline{bS_3}, 0), bS_1, 0))
\end{aligned}$$

The 1-digit BCD adder design using the first method could able produce the output with a delay of 3 clock cycles, second method by using 2D wave clocking could produce the output with a delay of 3.5 clock cycles.

After thorough review on recent BCD adder [14], it is observed that the boolean expression based model presented here is able to reduce the circuit delay. However, there are certain issues regarding the complexity of the design. The complex logic expressions in the design runs wires for long distances with crossing the other wires, for example the binary carry C_2 is a function of binary inputs bA_0, bB_0, bA_1, bB_1 and cin , it needs more number of wire crossovers. The multilayer wire crossovers used in the design will increase the QCA cost function [10]. As the power consumption of a design is proportionate to number of cells the long wires used in the BCD adder [14] require more power and number of cells intern increase the layout area.

To overcome the issues in the design presented [14], this paper proposed a novel BCD adder using a coplanar wire crossover based area optimized QCA full adder. The proposed design will reduce the circuit complexity (QCA cost function), area and cell count with an optimal clock delay.

4. Proposed Efficient BCD Adder

A novel architecture for QCA based BCD adder proposed in this paper is given in Fig. 6(a), using a 4-bit binary adder ADD1 and a 2-bit binary adder, an XOR gate to replace the 4-bit binary adder ADD2 in the architecture presented [14] given in Fig. 5 to reduce the layout complexity. The full adders for binary addition in the proposed architecture are implemented using a new area optimized QCA full adder. The binary addition is implemented using ripple carry addition; an early carry generation technique is used in the full adder to propagate carry to the next bit before finding the sum bit. The proposed BCD adder occupies very small area with high performance in comparison with BCD adders [11-14].

The 4-bit binary adder ADD1 is depicted in Fig. 6(a) is implemented using four area optimized full adders. The inputs of BCD adder $dA_{(3:0)}, dB_{(3:0)}$ are four bits of 1-digit decimal numbers and the input carry c_m . The outputs of ADD1 are binary sum $bS_{(3:0)}$ and output carry is $bcout$, then the binary result converted into decimal sum and carry using area optimized full adders.

4.1. An Area Optimized Full Adder

An area optimized full adder is designed to implement an efficient BCD adder. The proposed full adder in Fig. 7 is designed using only 52 QCA cells instead of 59 cells in recent area efficient full adder [18] without degradation in the performance to reduce the overall complexity of the BCD adder. In an area optimized full adder the majority gate expressions of carry output and binary sum in an i^{th} bit adder are given in (9) and (10) respectively.

$$C_{i+1} = M(dA_i, dB_i, C_i) \quad (9)$$

$$S_i = M(\overline{C_{i+1}}, M(dA_i, dB_i, \overline{C_i}), C_i) \quad (10)$$

Where dA_i , dB_i are i^{th} bits of inputs and C_i is carry input. The equations (9) and (10) can be expanded to find the logic functions of carry and sum of a full adder. To prove the functions in (9) and (10) assuming a , b , c are inputs, C_{out} carry and S sum outputs of the proposed full adder, then the expansion of carry and sum functions from majority gate expressions are given below.

$$C_{out} = ab + bc + ca$$

$$\begin{aligned} S &= \overline{(ab + bc + ca)}\{(ab + b\bar{c} + \bar{c}a) + c\} + (ab + b\bar{c} + \bar{c}a)c \\ &= (\bar{a}\bar{b} + \bar{b}\bar{c} + \bar{c}\bar{a})(a + b + c) + abc = \bar{a}\bar{b}c + a\bar{b}\bar{c} + \bar{a}b\bar{c} + abc \end{aligned}$$

The schematic diagram of a coplanar full adder with majority gates M1, M2, M3 and inverters I1, I2 is depicted in Fig. 7(a). The QCA layout of an area optimized QCA full adder using two corner cell based inverters I1, I2, two coplanar wire crossovers and three majority gates M1, M2, M3 as center cell of each majority gate directed with a blue line is designed using 52 QCA cells is shown in Fig. 7(b). Adder inputs are applied on clock0 (blue cells), the carry output generates on clock1 and the output sum produces on clock3, so the delay of carry output is two clock zones and the delay of sum is four clock zones

4.2. Proposed BCD Adder

The architecture of the 1-digit BCD adder proposed in this paper is depicted in Fig.6(a), in this the first block is a 4-bit binary adder (ADD1), which adds two input BCD numbers along with input carry and produce binary sum and carry $bS_{(3:0)}$ and $DCout$ respectively. From the result of ADD1 the decimal carry expression is given in (11). A 2-bit binary adder and an XOR gate are used to convert the binary sum $bS_{(3:0)}$ and decimal carry output $DCout$ to decimal sum $dS_{(3:0)}$. The logic functions for converting binary sum to decimal sum are given in (12). The digital logic circuit of BCD adder using (11) and (12) is depicted in Fig. 6(b). The binary sum is denoted as BS_0, BS_1, BS_2, BS_3 , carry of ADD1 as BC , decimal sum DS_0, DS_1, DS_2, DS_3 , decimal carry $DCout$ and internal carries of decimal are $c1, c2$.

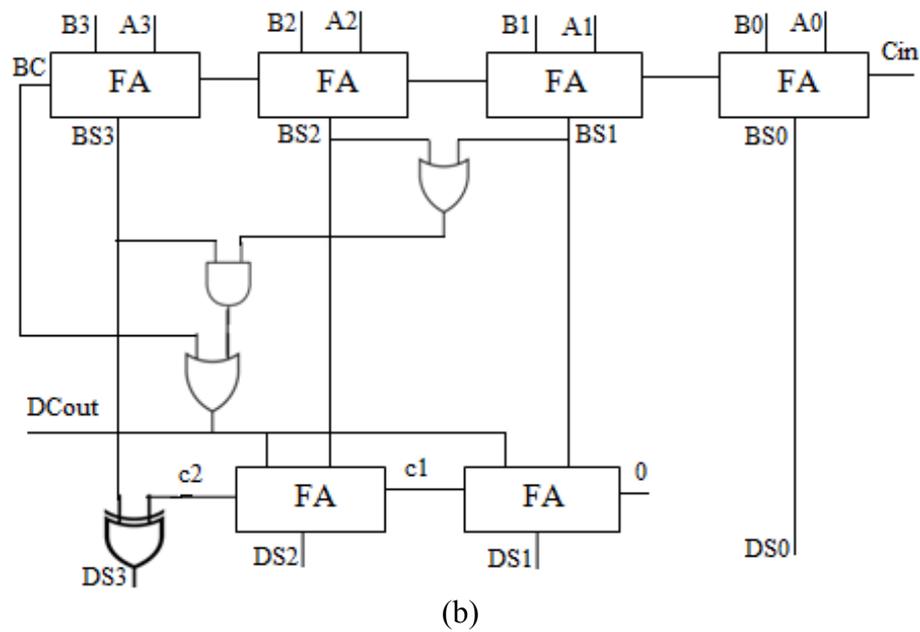
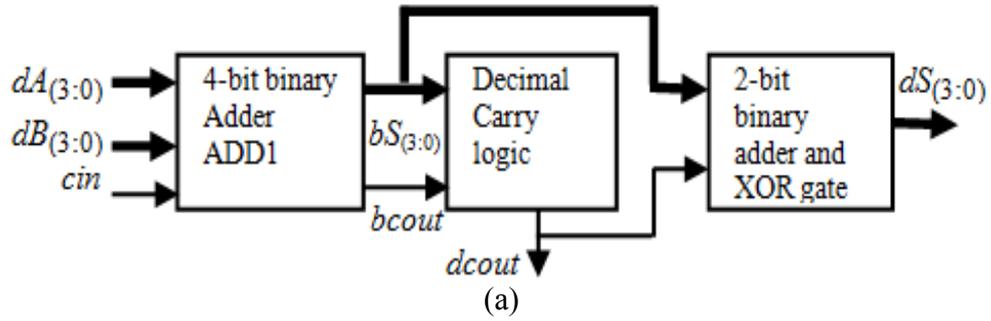


Fig. 6 BCD adder: (a) Architecture, (b) logic diagram

$$DCout = M(BC, M(bS_3, M(bS_2, bS_1, 1), 0), 1) \quad (11)$$

$$= BC + bS_3(bS_2 + bS_1)$$

$$dS_0 = bS_0 \quad (12)$$

$$c1 = M(DCout, bS_1, 0)$$

$$dS_1 = M(\overline{c1}, M(DCout, dS_1, 1), 0)$$

$$c2 = M(DCout, bS_2, c1)$$

$$dS_2 = M(\overline{c2}, M(DCout, bS_2, 1), 0)$$

$$dS_3 = M(\overline{M(bS_3, c2, 0)}, M(bS_3, c2, 1), 0)$$

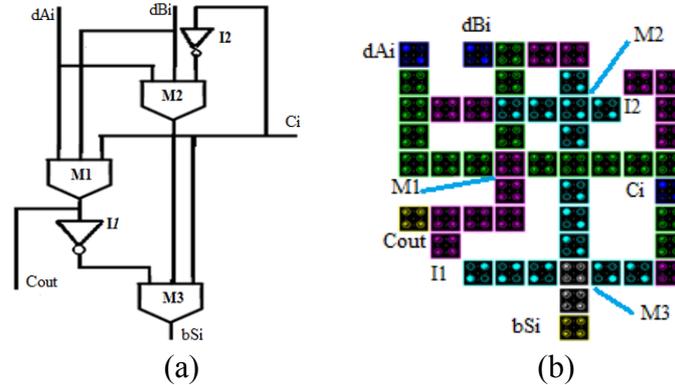


Fig. 7 Area optimized full adder; (a) block schematic, (b) QCA layout.

4.3. QCA Implementation of 1-Digit BCD Adder

A QCA layout of 1-digit BCD adder is designed for the novel BCD adder architecture given in Fig. 6(a) is implemented using six area optimized full adders, one XOR gate and three majority gates for carry logic given in eq. (4) and eq. (5). The QCA layout is depicted in Fig. 8(a), designed with 526 cells, it occupies $0.57\mu\text{m}^2$ area, the delay of the output carry is 2 cycles and delay the delay of decimal sum is 3.5 clock cycles. The layout is simulated in QCA design tool, QCADesigner [8]. The simulation results of the 1-digit BCD adder are depicted in Fig. 8(b), the delay of the output sum and carry are shown in the simulation waveforms. The results obtained in this proposed adder are compared with its existing counterparts in section 5.

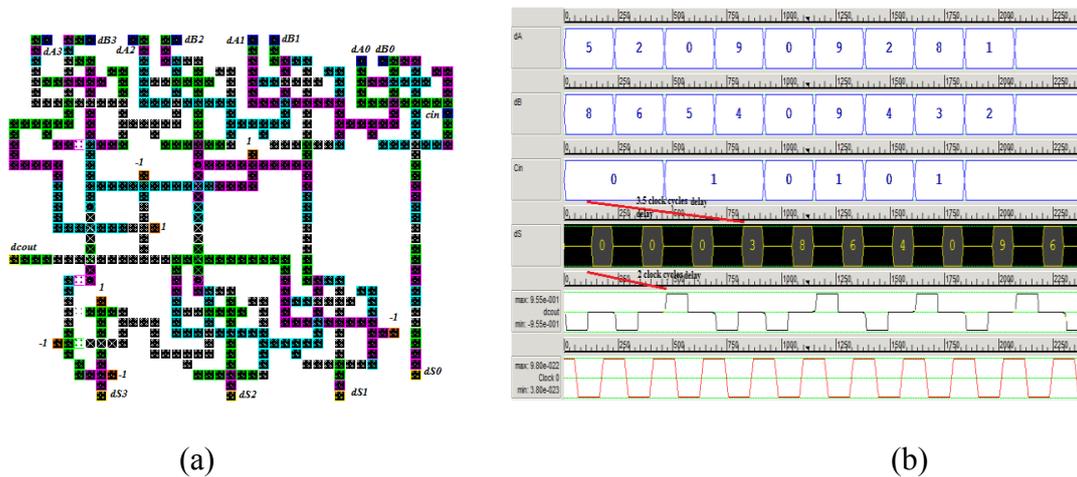


Fig. 8 BCD adder: (a) QCA layout, (b) simulation results

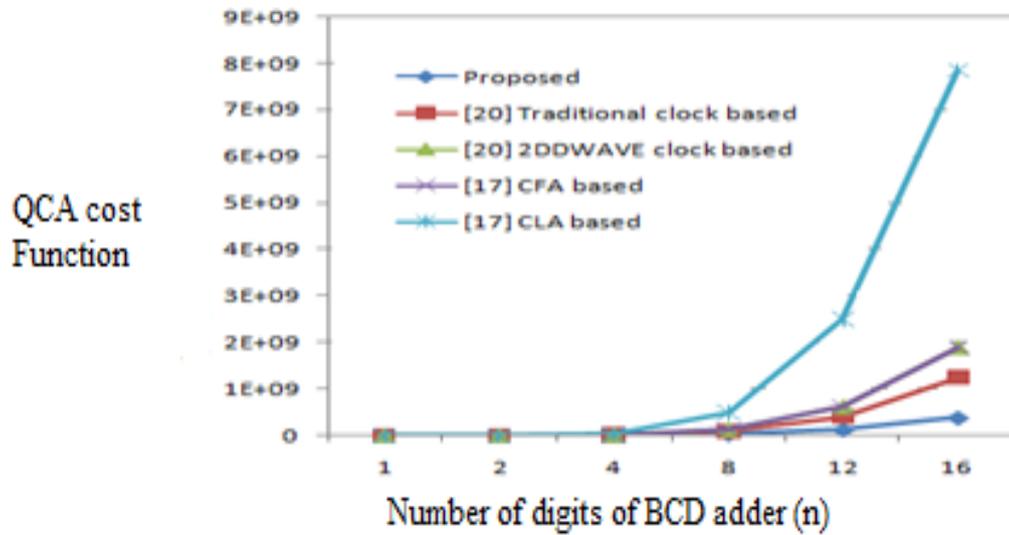


Fig. 9 Comparison of QCA_{cost} functions of different n-digit BCD adders.

4.4. Architecture of Proposed n-Digit BCD Adder

An efficient n-digit BCD adder is designed by cascading n one digit BCD adders. In the proposed design a 1-digit adder produce output carry with two clock cycles delay, and output sum with 3.5 clock cycles delay, this is an early carry generation. In an n-digit BCD adder design before finding sum digit, it propagates carry to the next digit with that it reduces the total delay of an n-digit adder. The delay of 1-digit BCD adder is 3.5 clock cycles, but the delay of a 2-digit BCD adder is only 5.5 clock cycles. Therefore the delay of an n-digit BCD adder can expressed as $2n + 1.5$ number of clock cycles. The architecture of an n-digit BCD adder is depicted in Fig. 10, input digits are represented with A, B , output digits represented with DS and output carry of n^{th} digit adder is represented with $DCoutn$.

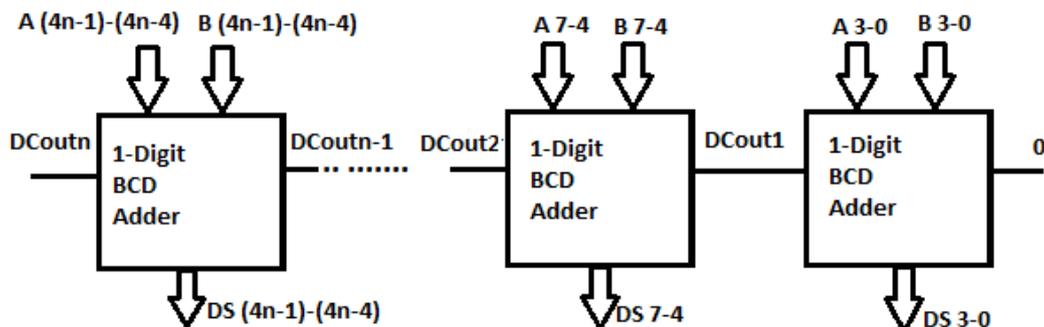


Fig. 10 Top level architecture of n-digit BCD adder.

5. COMPARISON OF RESULTS

In this section results obtained from the proposed novel QCA BCD adder are compared with the results of different existing counterparts [10-14]. In the proposed design the coplanar wire crossovers are preferred to reduce the circuit complexity. In a QCA design number of clock cycles, area and number cells are the design metric before defining QCA cost function [10]. The comparison of clock cycles, area and number of cells in proposed 1-digit BCD adder with its existing counterparts are given table 1. There is significant reduction in area and number of QCA cells with optimum number of clock cycles. The area of proposed design is reduced by 36% to 75%, further the QCA cell count is reduced by 43% to 71%.

The design performance can be compared with QCA cost function [10] in eq. (1), selected $l = 2$, $k = 2$ and $p = 2$. To evaluate the cost function of a 1-digit BCD adder here considered the number of majority gates is M , number of inverters is I , number of coplanar wire crossovers C_{cp} , the number of multilayer crossovers C_{ml} and the number of clock cycles T . The proposed BCD adder design require $M = 24$, $I = 13$, $C_{cp} = 12$, $C_{ml} = 5$ and its delay $T = 3.5$ clock cycles. A BCD adder design presented in [20] using traditional clocking scheme needs $M = 29$, $I = 8$, $C_{ml} = 27$ and its delay $T = 3$ clock cycles, whereas using 2-D wave clocking (2DDWAVE) [20] it needs $M = 29$, $I = 8$, $C_{ml} = 29$ and its delay $T = 3.5$ clock cycles. A BCD adder presented in [17] using Carry Flow Adders (CFA) requires $M = 24$, $I = 14$, $C_{ml} = 18$, its delay $T = 3.5$ clock cycles and using Carry Look-ahead Adders (CLA) it requires $M = 35$, $I = 13$, $C_{ml} = 56$, its delay $T = 2.5$ clock cycles. And a BCD adder presented in [19] requires $M = 27$, $I = 16$, $C_{ml} = 27$, its delay $T = 8$ clock cycles. These design metrics used to find QCA cost function in eq. (1) [11]. The results of cost function for different 1-digit BCD adders are given in table 2. The QCA cost function evaluated from proposed 1-digit BCD adder shows significant improvement, the cost function is reduced by 75% to 95% with existing designs.

The QCA cost function of n-digit BCD adders for $n = 1, 2, 4, 8, 12$ and 16 are estimated by the cost function given in eq. (1) [10], and plotted in Fig. 9, shows there is a reduction in QCA cost function, for 16-digit BCD adder cost function is reduced by 69% to 95% in comparison with existing designs [11-14]. The increase in cost function of proposed designs are linear for numbers digits (n) where increase for other designs is very high and nonlinear as shown in Fig. 9.

Table 1: Area, Cell count and Clock cycles Comparison of 1-digit QCA BCD Adders

BCD Adder	No. of cells	Area (μm^2)	No. of clock cycles
[14] Traditional clock based	1065	0.89	3
[14] DDWAVE clock based	1196	1.36	3.5
[12] CFA based	932	1.36	4.75
[12] CLA based	1838	1.86	2.5
[11] Serial adder	1130	1.77	10
[13] Ripple carry BCD adder	1348	2.28	8
Proposed	526	0.57	3.5

Table 2: QCA_{cost} Function Comparison of 1-digit BCD Adders

BCD Adder	QCA _{cost} function
[14] Traditional clock based	66,690
[14] 2DDWAVE clock based	103,120
[12] CFA based	79,104
[12] CLA based	184,137
[13] Ripple carry BCD adder	467,584
Proposed	16145

6. Conclusions

In this paper, design of an area efficient BCD adder is presented. To optimize the BCD adder an area optimized QCA full adder is used to perform the binary addition. The QCA layout is designed and simulated using the tool QCADesigner. The results of the proposed design shows, there is significant improvements in terms QCA design metrics. Finally, the results confirm that the proposed design reduced the complexity, area and cell count, without compromising in the delay, in comparison with existing designs.

In QCA technology there is a future scope to design complex combinational, sequential circuits, memories, ALU and different processors using proposed area optimized full adder. The efficient BCD adder proposed in this paper can be used to implement ALU.

References

- [1] Y. B. Kim, "Challenges for nanoscale MOSFETs and emerging nanoelectronics," *IEEE Trans. on Electrical and Electronics Materials*, vol. 11, no. 3, pp. 93-105, 2010.
- [2] International Technology Roadmap for Semiconductors (ITRS), <http://www.itrs2.net>, 2007.
- [3] C. S. Lent, P.D. Tougaw, W. Porod, and G.H. Bernstein, "Quantum Cellular Automata," *Nanotechnology*, vol. 4, no. 1, pp. 49-57, Jan.1993.
- [4] Y. Lu, M. Liu, and C. S. Lent, "Molecular quantum-dot cellular automata: From molecular structure to circuit dynamics," *Journal of applied physics*, vol. 102, no. 3, pp.034311-1-034311-7, 2007.
- [5] E. P. Blair, E. Yost, and C. S. Lent, "Power dissipation in clocking wires for clocked molecular quantum-dot cellular automata," *Journal of Comput. Electronics*, vol. 9, no. 1, pp. 49-55, 2010.
- [6] W. Porod, "Quantum dot devices and Quantum dot Cellular Automata," *Journal of Franklin Institute*, vol. 334B, no. 5/6, pp. 1147-1175, 1997.
- [7] H. Cho and E. Swartzlander, "Adder designs and analysis for quantum dot cellular automata," *IEEE Trans. on Nanotechnology*, vol. 6, no. 3, pp. 374-

- 383, 2007.
- [8] K. Walus, T. J. Dysart and G. Jullien and R. A. Budiman, "QCADesigner: A rapid and simulation tool for Quantum-dot Cellular Automata," *IEEE Trans. on nano.*, vol. 3, no. 1, pp. 26-31, 2004.
 - [9] H. Cho and E. Swartzlander, "Adder and multiplier design in quantum-dot cellular automata," *IEEE Trans. on Computers*, vol. 58, no. 6, pp. 721-727, Jun. 2009.
 - [10] W. Liu, L. Liang, M. O'Neill, and E. E. Swartzlander, "A first step towards cost functions for quantum dot cellular automata designs," *IEEE Trans. on Nano.*, vol. 13, no. 3, pp. 476-487, 2014
 - [11] M. Gladshtein, "Quantum dot Cellular Automata Serial Adder," *IEEE Trans. on Nanotechnology*, vol. 10. No. 6, pp. 1377-1382, 2011.
 - [12] W. Liu, L. Lu, M. O'Neill and E. E. Swartzlander, "Cost-Efficient Decimal Adder Design in Quantum dot Cellular Automata," *Proc. of IEEE Int. Symposium on Circuits and Systems*, Seoul, South Korea, pp. 1347-1350, May 2012.
 - [13] M. Taghizadeh, M. Askari, K. Fardad, "BCD Computing Structures in Quantum dot Cellular Automata," *Proc. of IEEE Int. Conference on Computers and Communications Engineering*, kuala Lumpur, Malaysia, pp. 1042-1045, May 2008.
 - [14] F. Kharbash, G. M. Chaudhry, "The Design of Quantum dot Cellular Automata Decimal Adder," *Proc. of IEEE Int. Multi topic Conference*, Karachi, Pakistan, pp. 71-75, December 2008.
 - [15] G. Cocorullo, P. Corsonello, F. Frustaci and S. Perri, "Design of Efficient BCD Adders in Quantum Dot Cellular Automata," *IEEE Trans. on Circuits and Systems*, vol. 63, no. 99, July 2016.
 - [16] M. Liu, and C. S. Lent, "High-speed metallic quantum-dot cellular automata," *Proc. of 3rd IEEE Conference on Nanotechnology (IEEE NANO)*, vol. 2, pp. 465-468, Aug. 2003.
 - [17] A. Pulimeno, M. Graziano, A. Sanginario, V. Cauda, D. Demarchi, and G. Piccinini, "Bis-ferrocene molecular QCA wire: Ab initio simulations of fabrication driven fault tolerance," *IEEE Trans. on Nanotechnology*, vol. 12, no. 4, pp. 498-507, July 2013.
 - [18] D. Adedi, G. Jaberipur, and M. Sangsefidi, "Coplanar full adder in quantum-dot cellular automata via clock-zone-based crossover," *IEEE Trans. on Nanotechnology*, vol. 14, no. 3, pp. 497-504, May 2015.

