QCA Based Design of Novel Low Power n-bit Ripple Carry Incrementer and Ripple Carry Decrementer

Bikash Debnath[1], JADAV CHANDRA DAS[2], and Debashis De[2]

[1]Sikkim Manipal Institute of Technology - Majitar
[2]Maulana Abul Kalam Azad University of Technology

November 21, 2022

Abstract

This paper shows the design of the newly made 'n' bit incrementer and decrementer circuit using quantum-dot cellular automata (QCA). Ripple carry incrementer and decrementer are crucial for performing two numbers' increment or decrement operation. This paper outlines the design of a novel low power ripple carry incrementer and ripple carry decrementer circuit based on QCA. The proposed ripple carry incrementer and ripple carry decrementer circuit are achieved with a new layout of the XOR gate, AND gate, and half adder circuit. This newly designed XOR and half adderare compared with state-of-the-artdesigns.QCA Designer 2.0.3 is used to design the circuits. The simulation results of 4-bit, 8-bit, and 16-bit incrementer and decrementer circuits are by the theoretical results.

1. Introduction

CMOS technology has reached its physical limit, comprehensive of quantum consequences and power dissipation. Higher design complexity might cause due to scaling down of logic circuits made up of CMOS to nanometer scale. An alternative to CMOS is essential for scaling down the device and escalating the efficiency of a microprocessor. Lent et al. Suggested QCA [1] technology as a substitute for CMOS.QCA [2-4] emerged as a rising technology to encode information. QCA is transistor-less, very low-power nanotechnology. It has a high-density interface and a quicker transfer speed. Encoding can be accomplished by the configuration of charges within a QCA cell. The coulombic association within the cells is enough to conduct the computation.

For this reason, QCA interconnecting wire is not essential between cells. However, revealing a fault in the received message is a significant issue in digital communication. At the nano-level, the unpredictability of the fault recognition scheme is the most challenging aspect regarding circuit region and energy dissipation at the equipment level.

QCA technology helps build various combinational and sequential digital logic circuits. The digital signals carry all the information in the form of digital strings of 0 and 1. These comprise a full adder, multiplier, multiplexer, comparator, counter, shifter, etc. Digital circuits that perform addition or subtraction use a full adder (FA) circuit[2]. This adder circuit created an innovative and efficient 4-bit Ripple Carry Adder (RCA) circuit. Multipliers are used extensively in DSPs and other applications nowadays. The addition and multiplication of two binary integers is an essential and widely used mathematical operation in high-performance systems such as microprocessors, DSPs, etc. A proficient coplanar 1-cycle comparator circuit [3] is proposed and assessed in the QCA. A majority gate, XNOR gate, and inverter are employed to build the QCA comparator circuit. QCA technology is used to design a novel coplanar counter circuit [4]. A unique serial to parallel QCA multiplier circuit is obtained based on the full adder [5]. A multiplexer receives several digital signals and selects one of them to send to the output at a time. Its function includes

data selection, routing, operation sequencing, parallel-to-serial conversion, waveform production, and logic function generation.

A novel low power ripple carry incrementer, and decrementer circuit based on QCA is proposed in this paper. Three incrementer and decrementer circuits of 4-bit,8-bit, and 16-bit using QCA designer are explored in this paper. Incrementer/Decrementer (INC/DEC) is a significant elementary unit in numerous computerized frameworks, for example, the location address formation unit of microcontrollers and microchips. The design utilizes a multi-layered wire crossing method. This circuit provides the basic concept of incrementer and decrementer circuits.

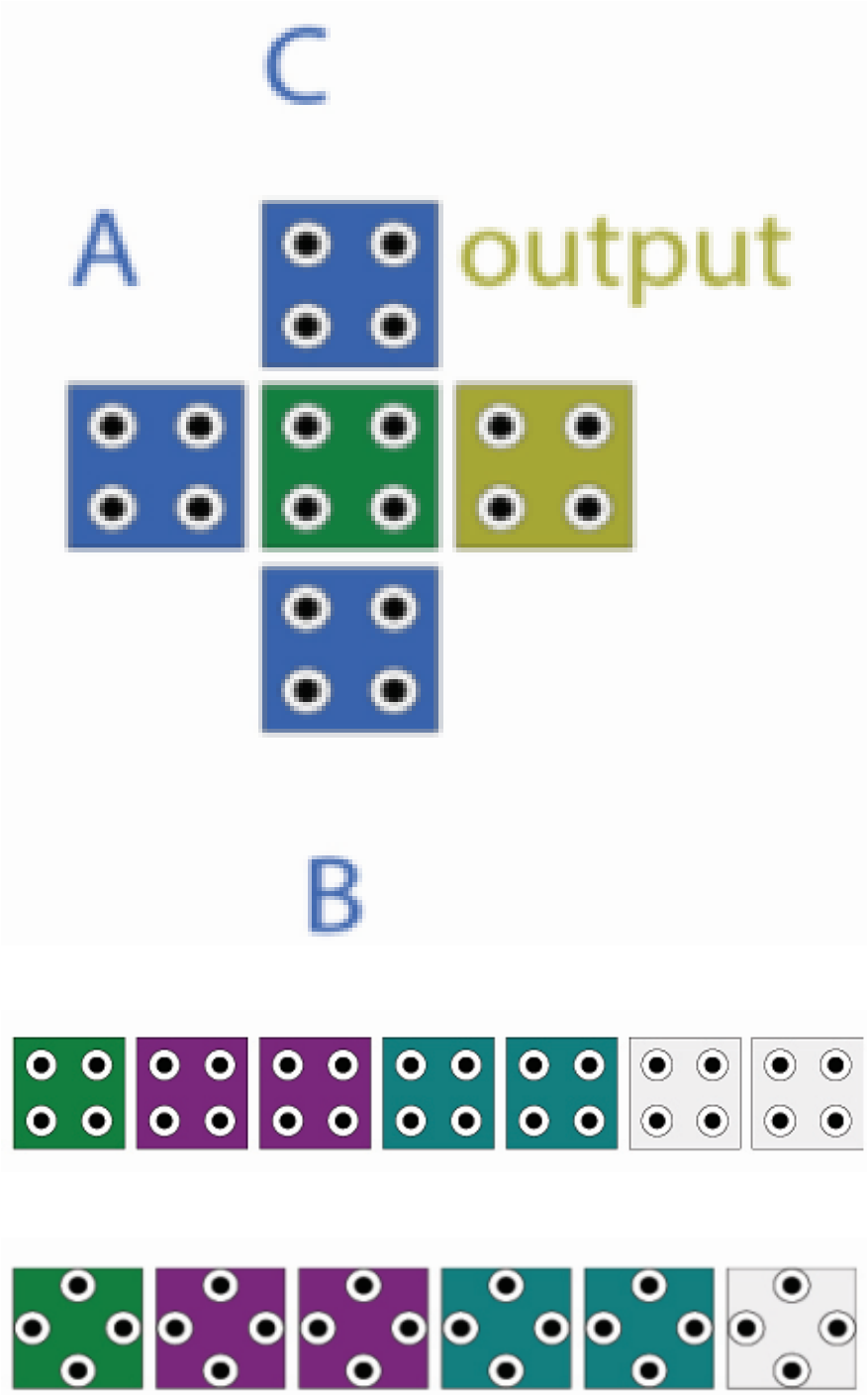
This paper is prepared as follows. The basic overview of QCA cell, majority voter (MV), wire, QCA logic gates, and the idea of incrementer and decrementer operations are presented in section 2. Section 3 demonstrates the related works. Section 4 comprises the complete structure of the proposed ripple carry incrementer and ripple carry decrementer circuit and its simulation output. Section 5 deals with the simulation parameters and design complexity and compares performance. Finally, the last, in section 6, the conclusion has been made.

2.QCA Overview

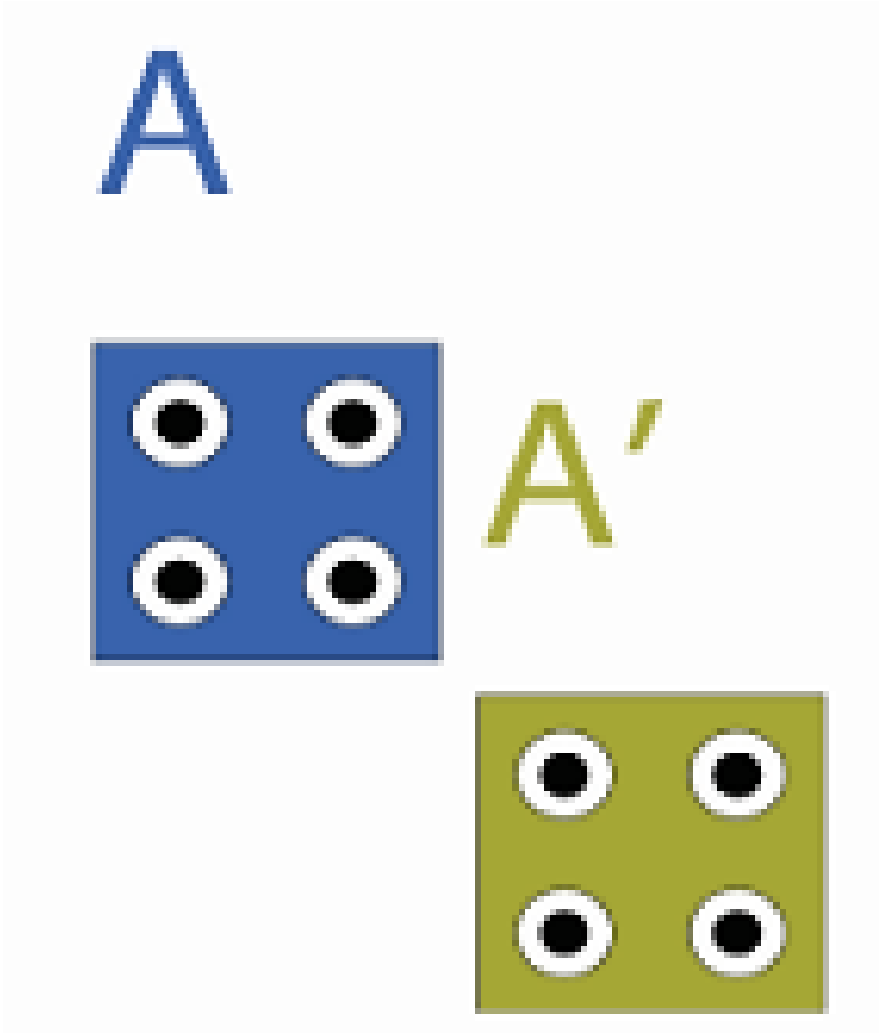
2.1 Basic of QCA

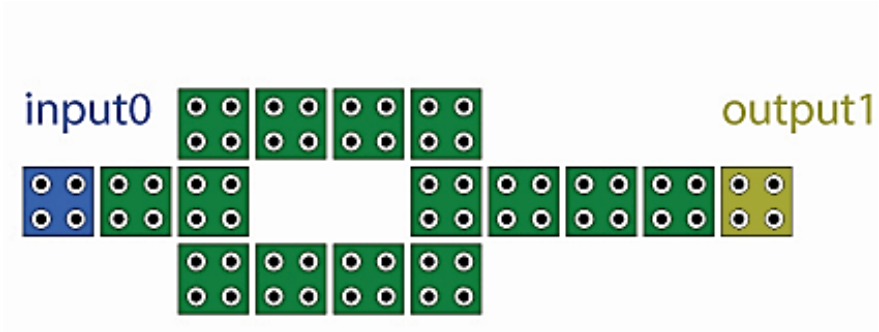
In QCA, each cell [5-6] is represented by a square box consisting of four identical quantum dots. Each dot can grip a single electron. First, 2-electrons are added, and then due to repulsive force, each electron occupies the diagonal position of the square box. The position of these electrons denotes their logic values (0 or 1), as shown in Fig. 1[5-6]. The primary majority voter (MV) comprises 5 QCA cells (Fig. 1a). It generally contains 3-inputs, and the inputs present in the majority of output will consider. By fixing the polarization value, AND and OR gatescan be achieved. Eqn. (1) shows MV expression.

$M(A, B,C)=AB+BC+CA$ (1)



(a) (b) (c)





(d) (e)

Fig.1 (a) QCA MV (b)90°QCA Wire (c)45º QCA Wire, (d,e)QCA inverters

The signal propagates from input to output using QCA wires [7-8] mainly of two type's 90° QCA wire shown in Fig. 1 (b) and 45° QCA wire shown in Fig.1(c). When QCA cells are diagonally placed, it acts as NOT

gate, i.e., inverter shown in Fig. 1(d), and 1(e).

2.2IncrementerandDecrementer

The binary incrementer increases the register's stored value by '1'.The present value stored in a register is added with '1' to increase its value by '1'. Combining 'n' half adders implement it for generating bits of 'n' numbers. Basic logic circuits required for producing incrementer are half adder and XOR[7-8] gate, as shown in Fig.2 and Fig.3, respectively. In addition, QCA-based ripple carries incrementer is shown in this paper. The binary decrementer circuit decreases the value of the stored register by '1'. The decrement can be done by adding 1 to the existing stored register. It acts according to the concept of 2's complement for subtraction purposes. It is made using an XOR gate, AND gate, and inverters.

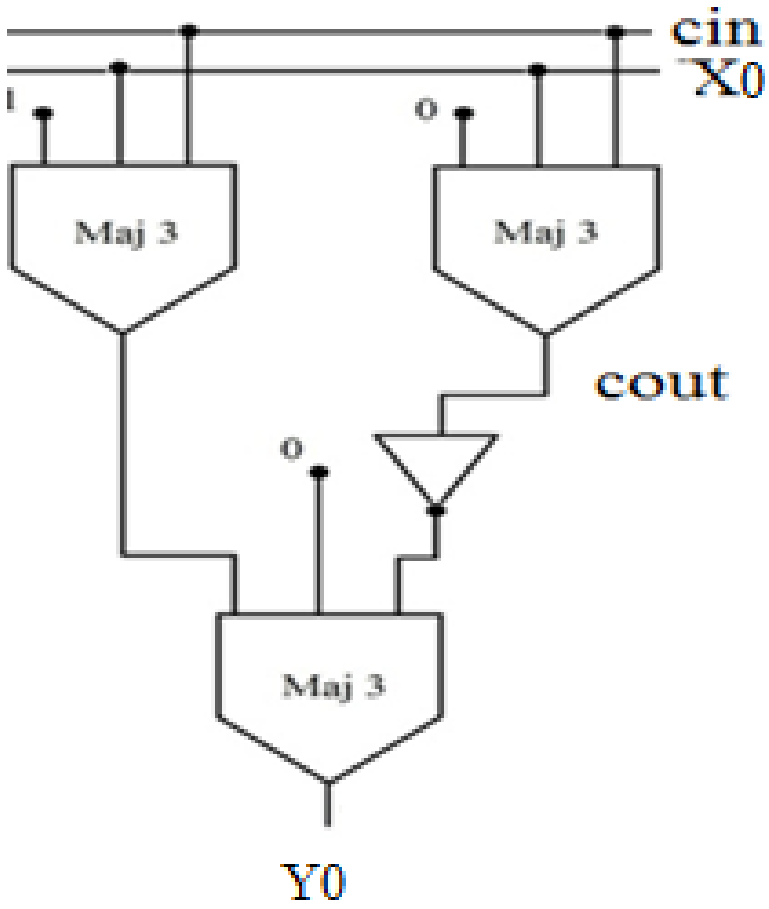
3. Related Works

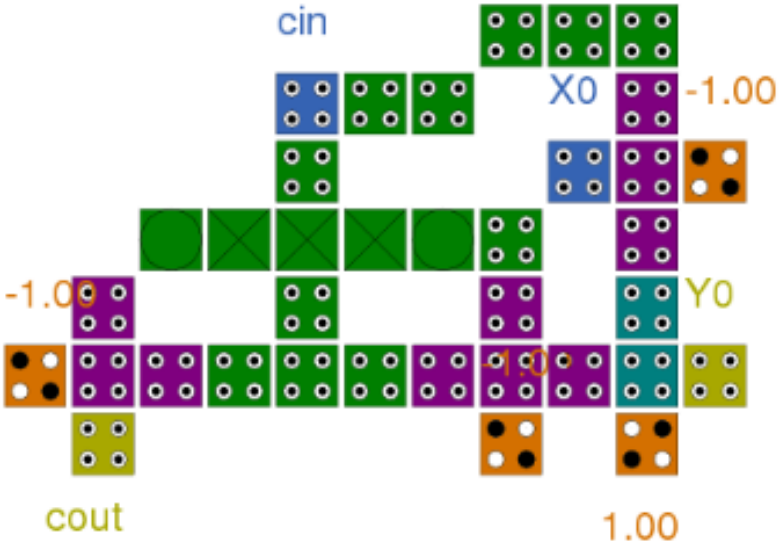
Few incrementer/decrementer circuits are made using CMOS technology [9].In [10], a technique is highly efficient in terms of space and time. Incrementer circuit is made using CMOS technology. The work reported in[11-12]presents the incrementer and decrementer design based on the adder/subtractor of the priority resolution module (PRM). Reversible binary incrementer based on QCA was reported in [13] where the incrementer circuit is made using reversible Peres gate with which half adder circuit is designed and cascading that half adder incrementer circuit is constructed.

4. Proposed Work

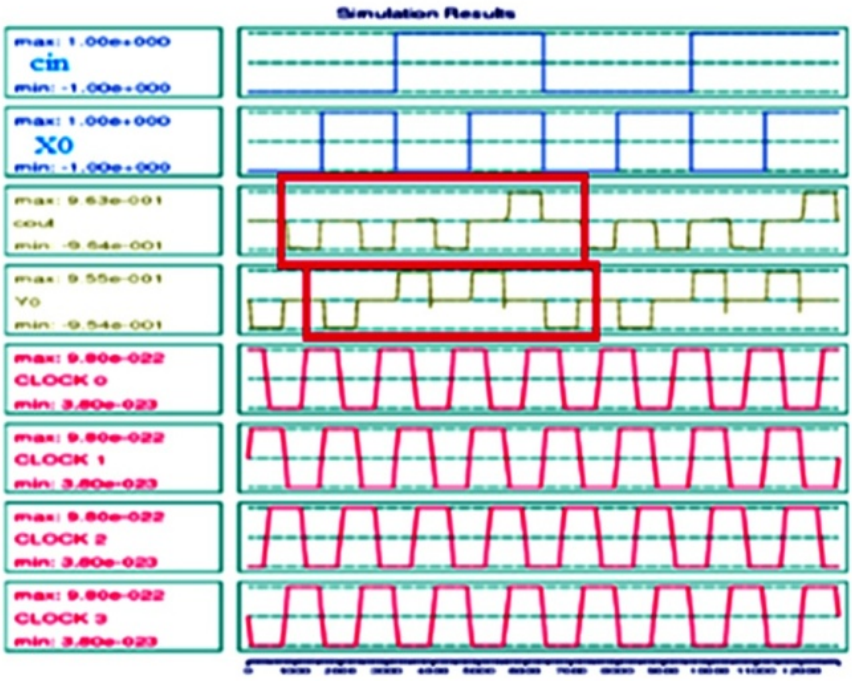
4.1 Building blocks of Ripple Carry Incrementer/ Decrementer

For designing a ripple carry incrementer circuit, half adder, and XOR are required. A half adder QCA schematic is displayed in Fig. 2(a), and its QCA implementation is displayed in Fig. 2(b). It comprises three majority gates. Its simulation result is displayed in Fig. 2(c). Another significant component for a ripple carries incrementer is the XOR gate, whose QCA schematic and QCA layout are displayed in Fig. 3. Its simulation result is displayed in Fig. 3(b).



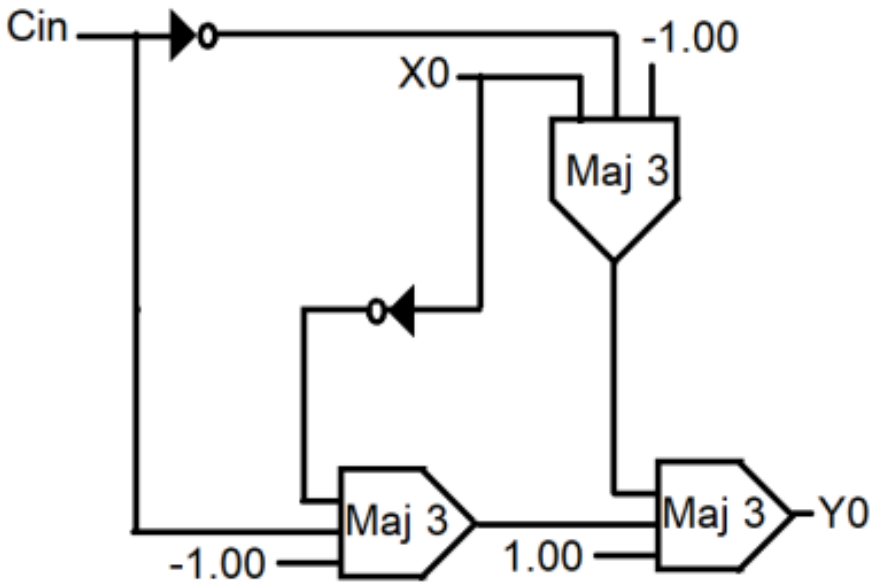


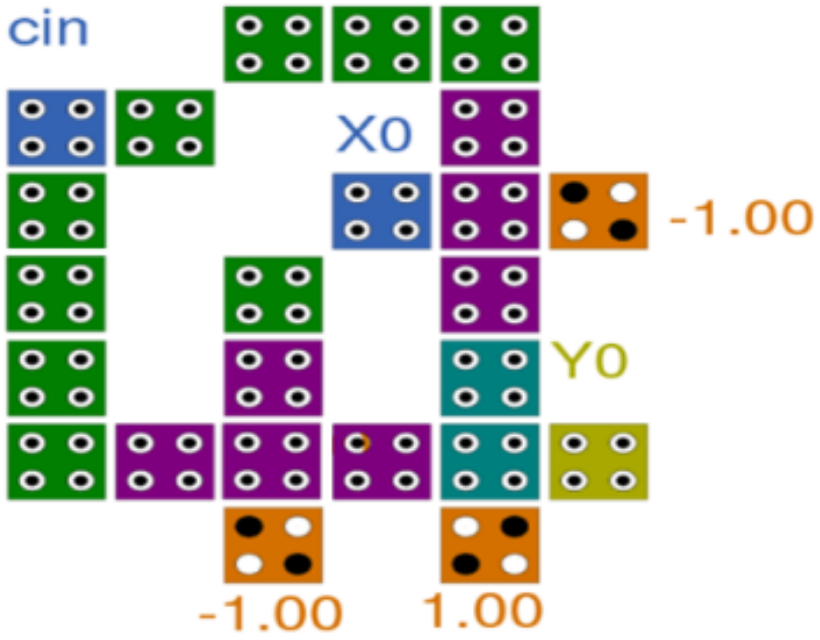
(a) (b)



(c)

Fig. 2 Half-adder (a) schematic, (b) Layout, (c) result





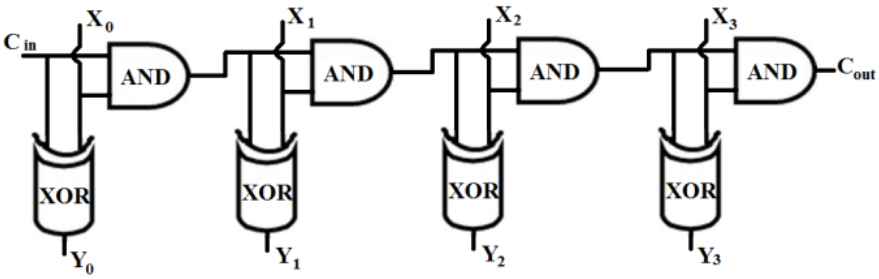
(a) (b)

(c)

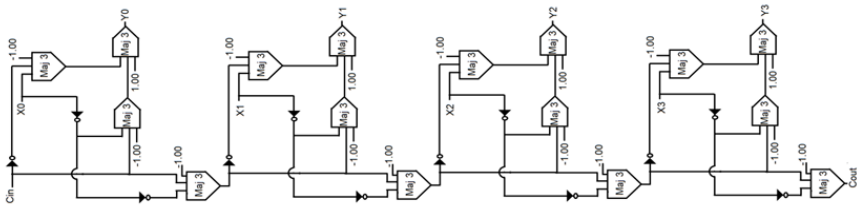
Fig. 3 XOR (a) schematic, (b) layout, (c) outputs.

4.2 4-bit Ripple Carry Incrementer

Four half adders cascaded for designing the ripple carry incrementer (4-bit) circuit. For accomplishing this circuit number of XOR gates required are four and four AND gates. The truth table of the 4-bit incrementer is described in Table 1.Logic representations for the 4-bit ripple carry incrementer is shown from equation (2)to (11).Let X_i(i=0,1,2,3)and C_{in} are the input bits taken and Y_i(i=0,1,2,3)and C_{out} are the corresponding output bits produced. The corresponding block diagram, the QCA schematic, is shown in Fig.4. The QCA layout and simulated result is shown in Fig.5.



(a)



(b)

Fig. 4 4-bit ripple carry incrementer (a) block (b) schematic.

Equations for 4-bit incrementer are shown:

$Y_0 = X_0$ [?]C_{in} (2)

$Y_1 = (X_0 .C_{in})$[?] X_1 (3)

$Y_2 = (X_0.C_{in}. X_1)$[?] X_2 (4)

$Y_3 = (X_0.C_{in}.X_1 . X_2)$[?] X_3 (5)

$C_{out} = X_0.C_{in}.X_1.X_2 .X_3$ (6)

Equations from (2) to (6) shown are represented in majority voter form through equations (7) to (11)

Y_0 = M (M (X_0´, C_{in}, 0), M (X_0, C_{in}´, 0), 1) (7)

Y_1 = M (M (M (X_0, C_{in}, 0)´, X_1, 0), M (M (X_0, C_{in}, 0), X_1´, 0), 1) (8)

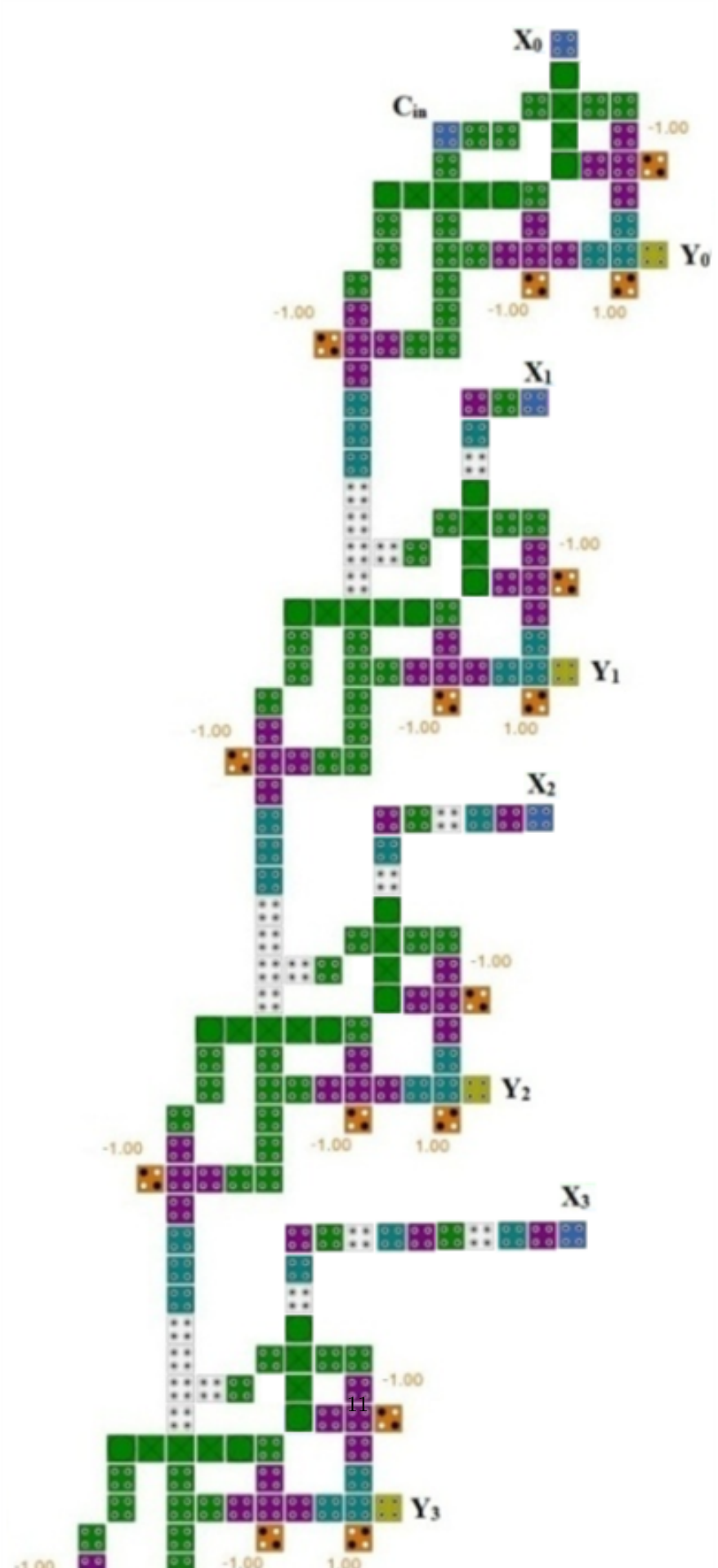
Y_2 = M (M (M (M (X_0, C_{in}, 0), X_1, 0)´, X_2, 0), M (M (M (X_0, C_{in}, 0), X_1, 0), X_2´, 0), 1) (9)

Y_3 = M (M (M (M (M (X_0, C_{in}, 0), X_1, 0), X_2, 0)´, X_3, 0), M (M (M (M (X_0, C_{in}, 0), X_1, 0), X_2, 0), X_3´, 0), 1)(10)

C_{out} = M (M (M (M (X_0, C_{in}, 0), X_1, 0), X_2, 0), X_3, 0) (11)

Table1: Truth table of 4bit incrementer

Input	**Input**	**Input**	**Input**	**Output**	**Output**	**Output**	**Output**	**Output**
$\mathbf{X_3}$	$\mathbf{X_2}$	$\mathbf{X_1}$	$\mathbf{X_0}$	$\mathbf{Y_3}$	$\mathbf{Y_2}$	$\mathbf{Y_1}$	$\mathbf{Y_0}$	$\mathbf{C_{out}}$
0	0	0	0	0	0	0	1	0
0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	1	1	0
0	0	1	1	0	1	0	0	0
0	1	0	0	0	1	0	1	0
0	1	0	1	0	1	1	0	0
0	1	1	0	0	1	1	1	0
0	1	1	1	1	0	0	0	0
1	0	0	0	1	0	0	1	0
1	0	0	1	1	0	1	0	0
1	0	1	0	1	0	0	1	0
1	0	1	1	1	1	0	0	0
1	1	0	0	1	1	0	1	0
1	1	1	0	1	1	1	1	0
1	1	1	1	0	0	0	0	1

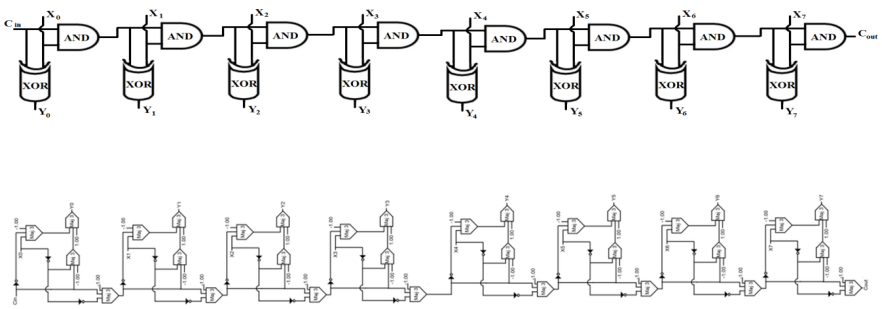


(a) (b)

Fig. 5 Proposed 4-bit ripple carry incrementer (a) QCA layout and (b) Simulation outcome

4.3 8 -bit ripple carry incrementer

For designing an 8-bit binary incrementer, eight half adders are needed, and the number of XOR and AND gates required are eight. Logic expressions have shown from equation numbers 12-20.Let X_i(where i=0,1,2,3,4,5,6,7),and C_{in} are the taken input bits and Y_i(where i=0,1,2,3,4,5,6,7)and C_{out} are the corresponding output bits we get from 8-bit incrementer circuit. The block diagram and QCA schematic of the 8-bit incrementer circuit are shown in Fig.6 (a) and Fig.6 (b), respectively. The corresponding QCA layout of the circuit is displayed in Fig.7 (a)and the simulation outcome in fig 7(b), respectively.



(a)(b)

Fig.6 8-bit ripple carry incrementer(a)block diagram (b)QCA schematic

Equations for 8 bit ripple carry incrementer is shown in eqns. 12 to 20

$Y_0 = X_0$ [?]C_{in} (12)

$Y_1 = (X_0 .C_{in})$[?] X_1 (13)

$Y_2 = (X_0.C_{in}. X_1)$[?] X_2 (14)

$Y_3 = (X_0.C_{in}.X_1 . X_2)$[?] X_3 (15)

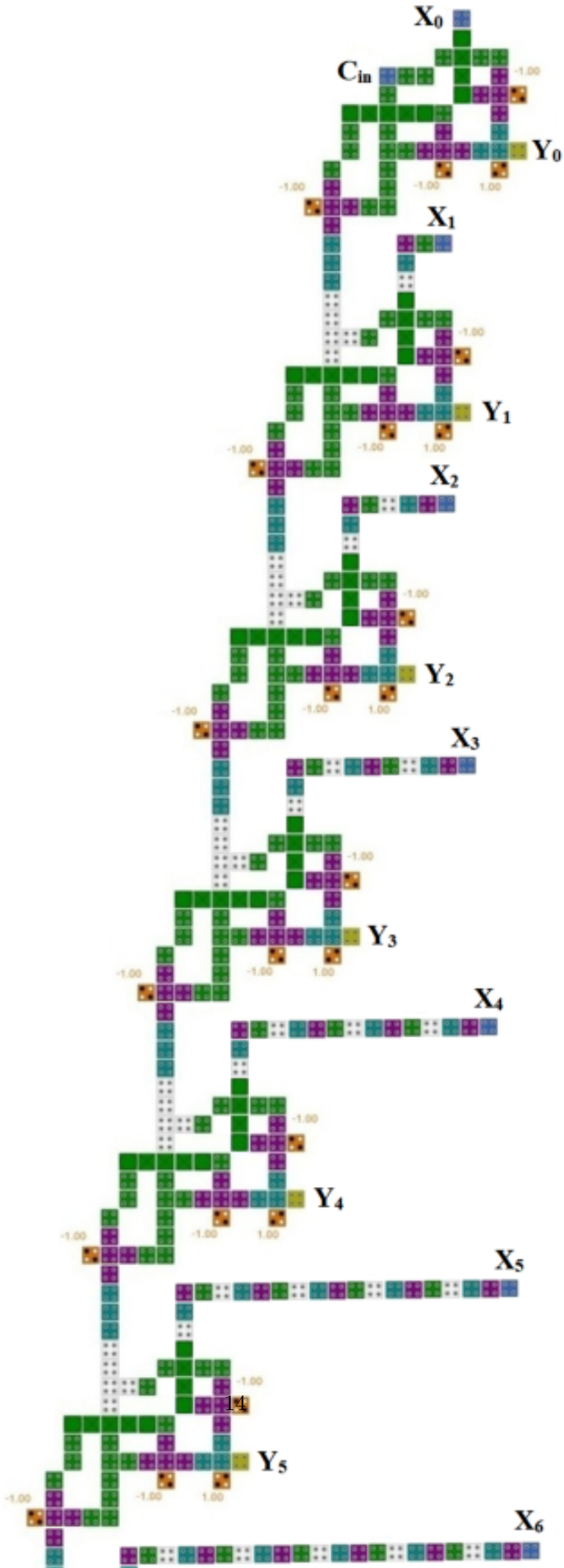
$Y_4=(X_0.C_{in}.X_1.X_2 . X_3)$[?] X_4 (16)

$Y_5 = (X_0 .C_{in} .X_1.X_2.X_3.X_4)$[?] X_5 (17)

$Y_6 =(X_0 .C_{in} .X_1.X_2.X_3.X_4.X_5)$[?] X_6 (18)

$$Y_7 = (X_0 \,.C_{in}\,.X_1.X_2.X_3.X_4.X_5X_6.)[?]\ X_7 \quad (19)$$

$$C_{out} = X_0.C_{in}.X_1.X_2\ .X_3.\ .X_4.X_5X_6.X_7 \quad (20)$$

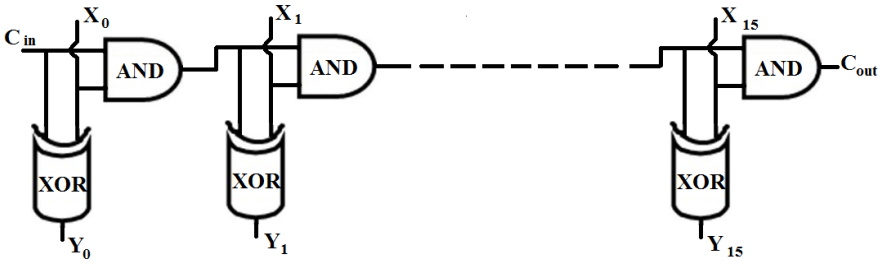


(a) (b)

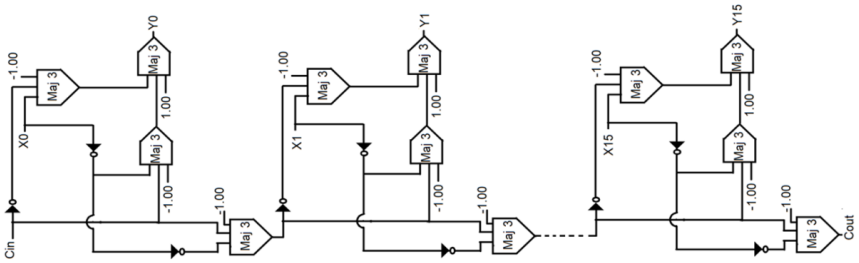
Fig.7 Proposed 8-bit ripple carry incrementer (a) QCA layout and (b) Simulation outcome.

4.4 16 bit ripple carry incrementer

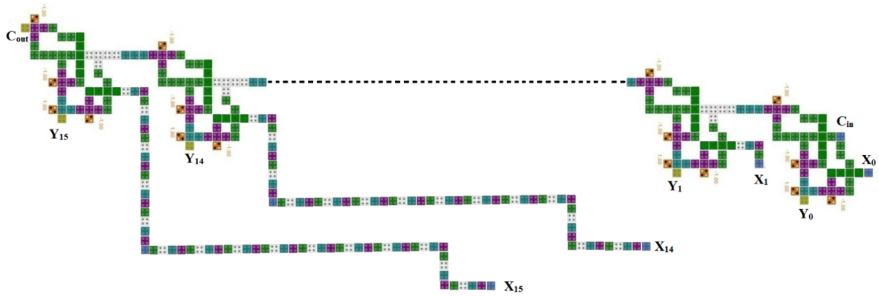
For designing 16-bit binary incrementer, sixteen half adders are needed and sixteen XOR and AND gates are required.Let X_i(i=0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)and C_{in} are taken as input bits and Y_i(i=0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)and $C_{outb}$are the corresponding output bits obtained from 16-bit incrementer circuit. The block diagram of incrementer circuit obtained here is shown in Fig.8 (a) and QCA schematic displayed in Fig.8 (b)and the corresponding QCA layout of the circuit are displayed in Fig.8(c).



(a)



(b)

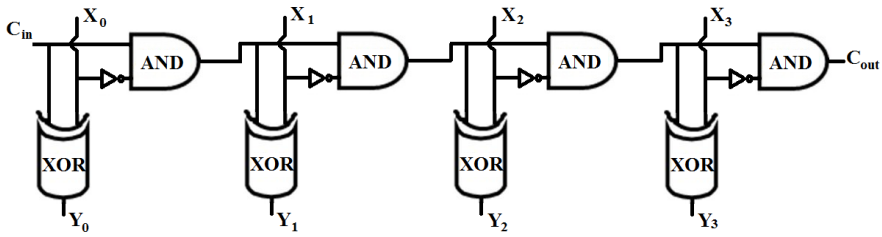


(c)

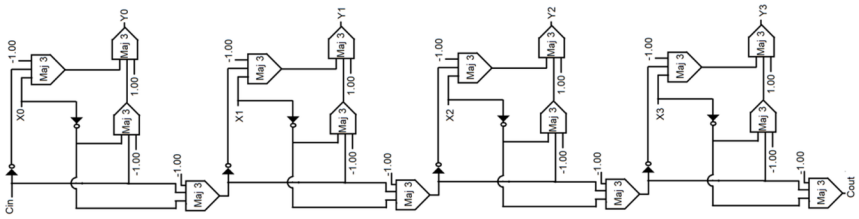
Fig. 8 Proposed 16-bit ripple-carry incrementer (a) Block Diagram(b) 16-bit ripple-carry incrementer and (c) QCA layout

4.5 4 bit decrementer

The design of a 4-bit decrementer was done using four XOR gates, four AND gates, and four NOT gates, as shown in Fig.9. The block diagram and schematic are in Fig.9 (a) and (b). Equations for a given 4-bit decrementer are shown from equation number (21) to (25). The truth table of the 4-bit decrementer is described in table 2 and verified with the corresponding equations. Let X_i(where i=0,1,2,3)and C_{in} are the input bits taken, and Y_i(i=0,1,2,3) and C_{out} are the corresponding output bits obtained. The corresponding layout and simulated result are in Fig. 10.



(a)



(b)

Fig.9 4-bit ripple carry decrementer (a) block (b) schematic

Equations for 4-bit decrementer are shown

Y0 = X_0[?]Cin (21)

Y1 = (X_0 '. Cin) [?] X_1 (22)

Y2 =(X_0'. Cin . X_1') [?] X_2- (23)

Y3 = (X_0'. Cin .X_1' . X_2') [?] X_3 (24)

Cout = X_0'. Cin .X_1' .X_2' .X_3' (25)

Equations from 21 to 25 are represented by the majority voter through equations 26 to 30

Y_0 = M (M (X_0´, C_{in}, 0), M (X_0, C_{in}´, 0), 1) (26)

Y_1 = M (M (M (X_0´ ,C_{in}, 0)´, X_1, 0), M (M (X_0´, C_{in}, 0), X_1´, 0), 1) (27)

Y_2 = M (M (M (M (X_0´, C_{in}, 0), X_1´, 0)´, X_2, 0), M (M (M (X_0´, C_{in}, 0), X_1´, 0), X_2´, 0), 1) (28)

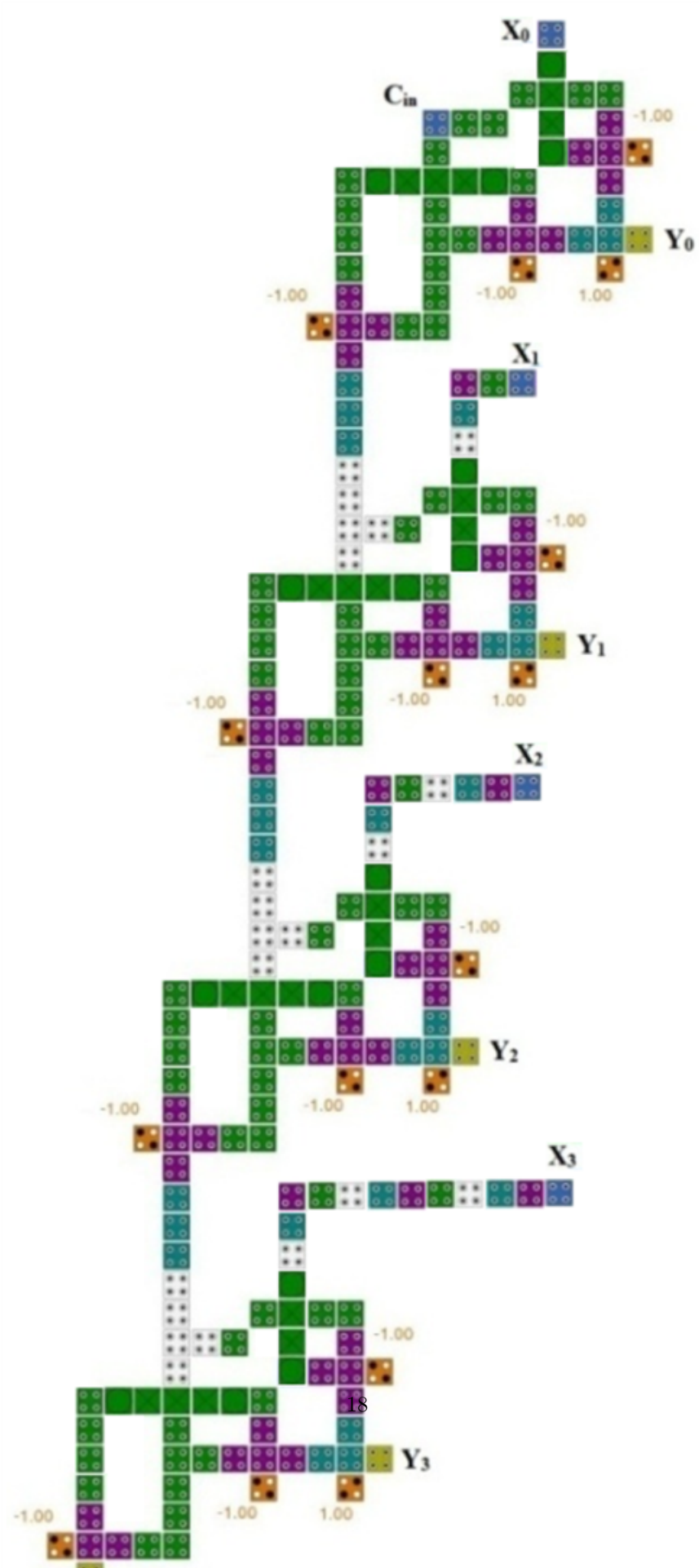
Y_3 = M (M (M (M (M (X_0´, C_{in}, 0), X_1´, 0), X_2´, 0)´, X_3, 0), M (M (M (M (X_0´, C_{in}, 0), X_1´, 0), X_2´, 0), X_3´, 0), 1)

(29)

C_{out} = M (M (M (M (X_0 ´, C_{in}, 0), X_1´, 0), X_2´, 0), X_3´, 0) (30)

Table 2: Truth table of 4bit decrementer

Input	**Input**	**Input**	**Input**	**Output**	**Output**	**Output**	**Output**	**Output**
X3	**X2**	**X1**	**X0**	**Y3**	**Y2**	**Y1**	**Y0**	**Cout**
0	0	0	0	1	1	1	1	1
0	0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	1	0	0
0	1	0	0	0	0	1	1	0
0	1	0	1	0	1	0	0	0
0	1	1	0	0	1	0	1	0
0	1	1	1	0	1	1	0	0
1	0	0	0	0	0	1	1	0
1	0	0	1	1	0	0	0	0
1	0	1	0	1	0	0	1	0
1	0	1	1	1	1	1	0	0
1	1	0	0	1	1	1	1	0
1	1	1	0	1	1	0	1	0
1	1	1	1	1	1	1	0	0

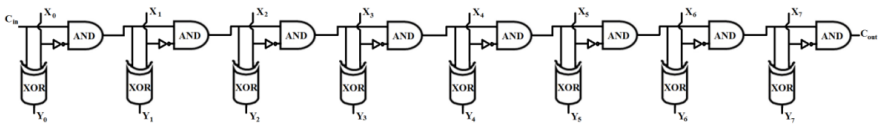


(a) (b)

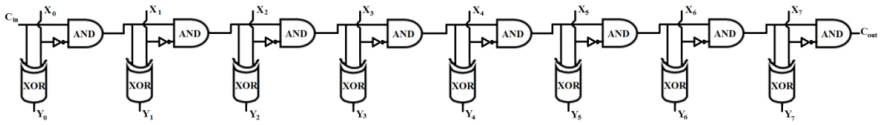
Fig. 10 Proposed 4-bit ripple carry decrementer (a) QCA layout and (b) Simulation outcome

4.6 8 -bit decrementer

The design of the 8bit decrementer consists of eight XOR gates, eight AND gates, and eight NOT gates. Equations for the given 8-bit decrementer are shown from equation number 31to 39. The block diagram and layout of the 8-bit decrementer circuit are shown in Fig.11 (a) and Fig.11 (b). X_i(i=0,1,2,3,4,5,6,7)and $C_{in}$are the taken input bits and Y_i(i=0,1,2,3,4,5,6,7)and $C_{out}$are the corresponding output bits from 8-bit decrementer circuit. The decrementer circuit is shown in fig 11(a) and Fig.11 (b) in QCA, so the corresponding QCA layout of the circuit is displayed in Fig 12(a)and the simulation outcome in Fig. 12(b).



(a)



(b)

Fig. 11 8-bit ripple carry decrementer (a) block (b) schematic

Equations for 8-bit decrementer are shown

Y0 = X_0[?]Cin (31)

Y1 = (X_0 ’. Cin) [?] X_1 (32)

Y2 =(X_0’. Cin . X_1’) [?] X_2 (33)

Y3 = (X_0’. Cin .X_1’ . X_2’) [?] X_3 (34)

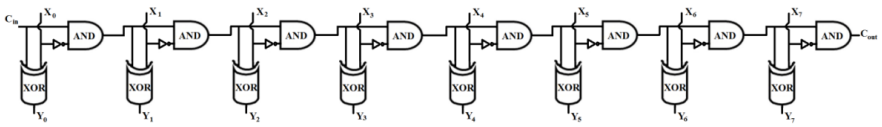
Y4 = (X_0’. Cin .X_1’ . X_2’.X_3’)[?] X_4 (35)

Y5 = (X_0’. Cin .X_1’ . X_2’.X_3’.X_4’)[?] X_5 (36)

$Y6 = (X_0'.\ Cin\ .X_1'\ .\ X_2'.X_3'.X_4'.X_5'\ )[?]\ X_6$ (37)

$Y7 = (X_0'.\ Cin\ .X_1'\ .\ X_2'.X_3'.X_4'.X_5'.X_6'\ )[?]\ X_7$ (38)

$Cout = (X_0'.\ Cin\ .X_1'\ .X_2'\ .\ X_3'.X_4'.X_5'.X_6'.X_7')$ (39)

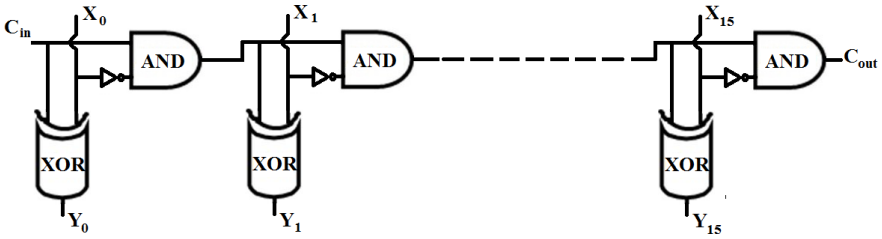


(a) (b)

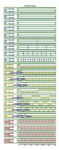
Fig. 12 Proposed 8-bit ripple carry decrementer (a) QCA layout and (b) Simulation outcome

4.716 bit Decrementer

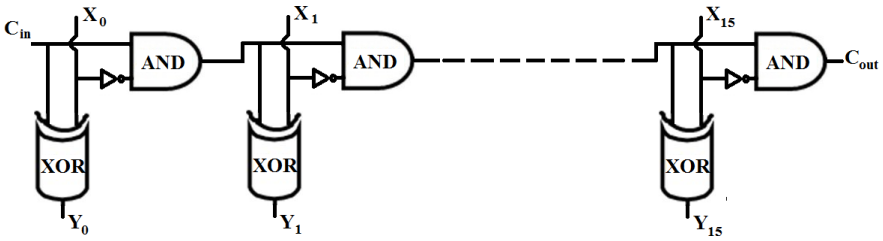
Design of 16-bit decrementer done using sixteen XOR gates, sixteen AND gates, and sixteen NOT gates. Block diagram and QCA layout of 16-bit decrementer circuit is shown in figs. 13(a) and 13(b), respectively. X_i(i=0, 1, 2, ..., 15), and C_{in} are the input bits, and Y_i(i=0, 1, 2, ..., 15) and C_{out} are the corresponding output bits of the 16-bit decrementer circuit. Fig 13(c) shows the QCA layout.



(a)



(b)



(c)

Fig.13 Proposed 16-bit ripples carry decrementer (a) Block diagram (b) schematic, and (c) layout

5. Result and Discussions

The parameter used in the QCA Designer tool to simulate the proposed design is shown in Fig. 14. The cell size is 18nm × 18nm. During simulation, the coherence vector with the Euler method is considered.

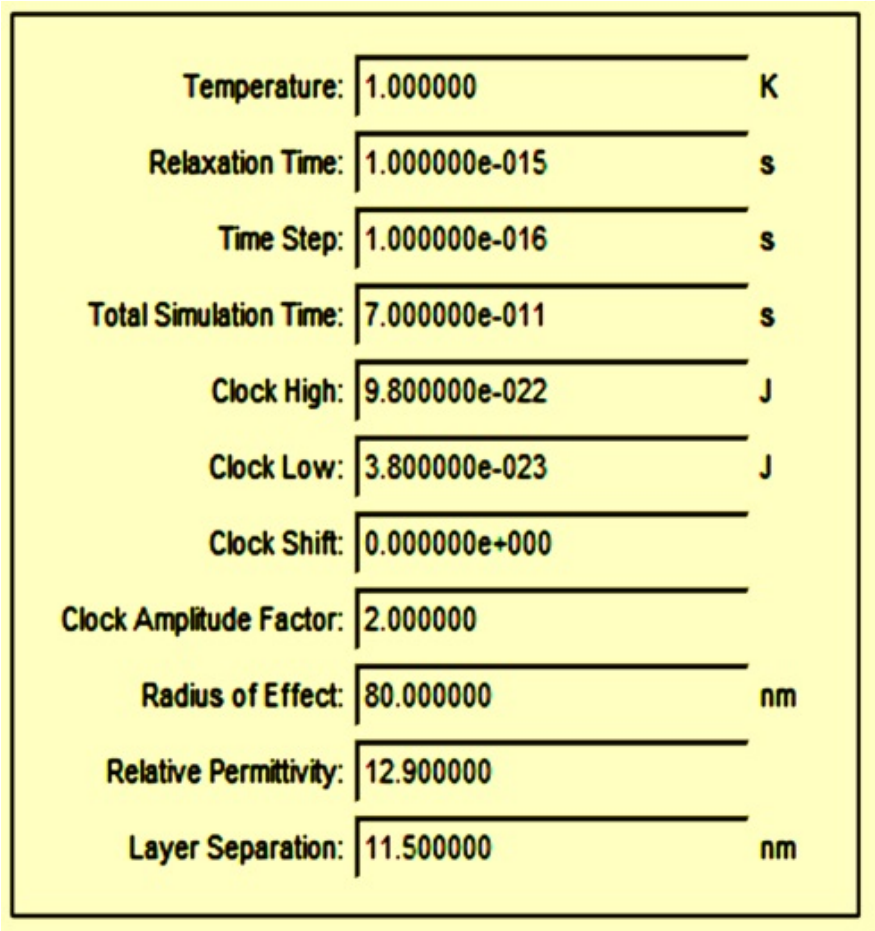


Fig .14 Lists of parameters used during simulation.

5.1. Design Complexity

Table 3 shows the design complexity of the proposed QCA layouts. The complexity is measured in terms of MVs, cells, area, and clock cycles. For example, the XOR circuit required cell count, area, and clock cycle to be 25, 0.0168, and 3. For making a half adder circuit, the required cell count, area, and clock cycle are 36, 0.031, and 3, respectively.

The QCA circuit of 4-bit incrementer requires 245 cells, 0.4536 μm^2 total area, 0.07938 μm^2 cell area, 16 MVs, 12 inverters, and 4 clock cycles. For building a QCA circuit of 8-bit incrementer,557 cells, 0.7344 μm^2total area,0.18046μm^2 cell area, 32 MVs, 24 inverters, and 8 clock cycles are required. 2168 cells, 2.3040μm^2 total area, 0.70243μm^2 cell area, 64 MVs, 48 inverters, and 16 clock cycles are required for obtaining a QCA circuit of 16-bit incrementer.

The QCA circuit of the 4-bit decrementer requires only 249 cells,0.4536 μm^2 total area, 0.08067 μm^2 cell area, 16 MVs, 8 inverters, and 4 clock cycles. For making a QCA circuit of 8-bit decrementer, 565 cells,0.7344 μm^2 total area, 0.18306μm^2 cell area, 32 MVs, 16 inverters, and 8 clock cycles are required. 2184 cells, with a 2.4576 μm^2 total area, a 0.70761μm^2 cell area, 64 MVs, 32 inverters, and 16 clock cycles are needed to make a QCA circuit 16-bit decrementer.

Comparison is done on half adder with previous designs and explored in Table 4, which explores that the proposed design requirement is less in terms of numbers of cells, cell area, and clock cycle are lesser than

others. Bar charts are prepared from this table to show the comparative study on each category displayed in Fig. 15. Figure 15(a) shows the comparison based on the numbers of cells used, Fig. 15(b) shows a comparison based on total cell area, Fig.15 (c) shows the comparison based on clock cycles used, and Fig.15 (d) shows the comparison based on area-delay cost.

Comparison is done on XOR gate with previous designs and explored in Table 5.It is observed from Table 5 that the proposed XOR gate has higher design complexity than other existing designs. But, in the proposed work, we have tried to show the design of the n-bit Ripple Carry Incrementer and Ripple Carry Decrementer circuit. So, for example, the design of 4-bit, 8-bit, and 16-bit ripple-carry incrementer and decrementer explored their design complexity.

The proposed ripple carry incrementer, and decrementer circuit is achieved by utilizing the XOR gate and Half-adder circuit. As a result, first, we have to design the XOR gate and Half-adder circuit, then it is used in designing incrementer and decrementer layouts. So, for better visualization, a comparison with other state-of-the-art designs is performed.

Table 3. The complexity of proposed QCA circuits

Proposed circuit	**Majority gate(MVs) and inverters(inv)**	**Cell count**	**Total area (um^2)**	**Cell Area(um^2)**
XOR	3Mvs and 1inv	25	0.0168	0.010
Half adder	3Mvs and 1inv	36	0.031	0.0144
4-bit incrementer	16Mvs 12 inv	**245**	**0.4536**	**0.07938**
8-bit incrementer	32Mvs 24inv	**557**	**0.7344**	**0.18046**
16-bit incrementer	64Mvs 48inv	**2168**	**2.3040**	**0.70243**
4-bit decrementer	16Mvs 8inv	**249**	**0.4536**	**0.08067**
8-bit decrementer	32Mvs 16inv	**565**	**0.7344**	**0.18306**
16-bit decrementer	64Mvs 32inv	**2184**	**2.4576**	**0.70761**

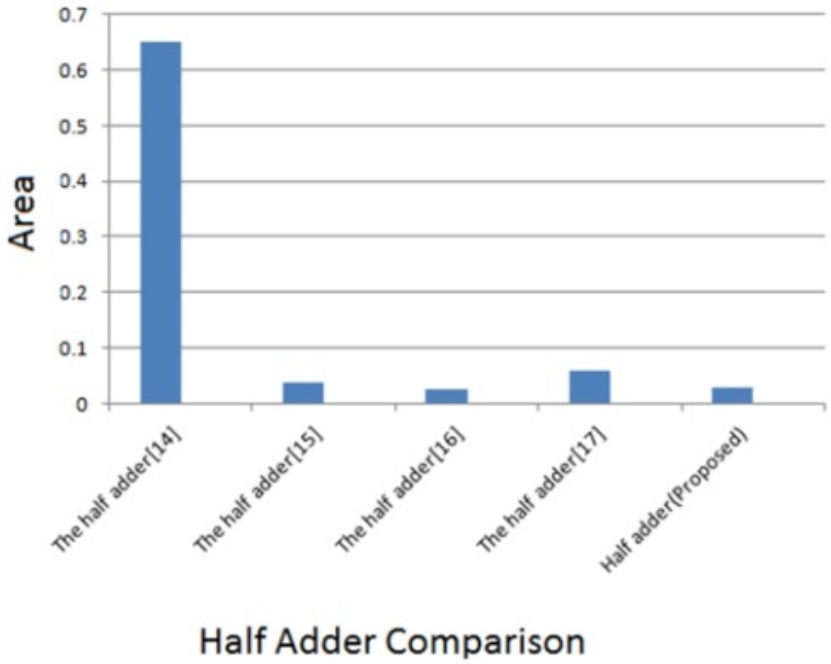
Table 4. Comparative Study of QCA of Half adder

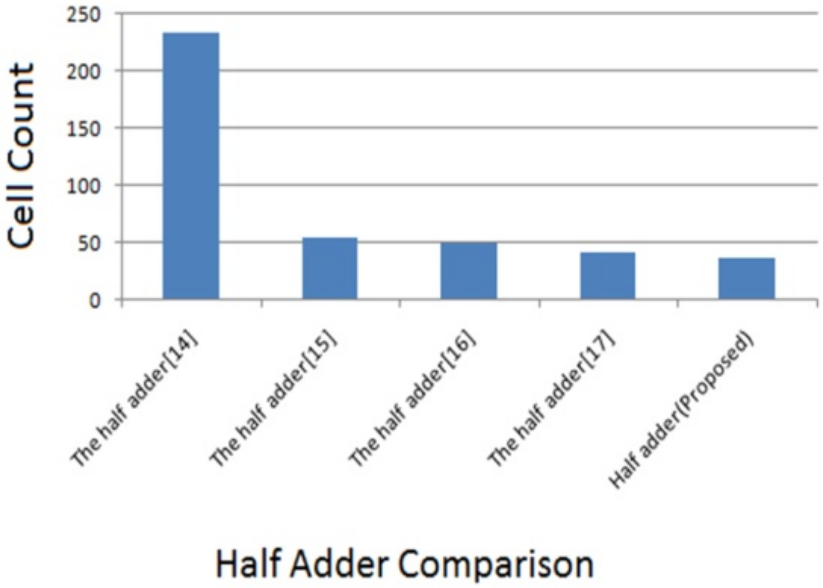
Circuit	**Cell count**	**Area**	**Clock cycle**	**Cross-over Type**	**Area-Delay Cost (Area × Latency^2)**
The half adder[14]	234	0.65	16	No wire crossing	166.4
The half adder[15]	53	0.04	3	Multilayer	0.36
The half adder[16]	48	0.027	2	No wire crossing	0.108
The half adder[17]	40	0.06	4	Multilayer	0.96
Half adder(Proposed)	36	0.03	3	Multilayer	0.27

Table 5. Comparative Study of XOR

Circuit	**Cell count**	**Area**	**Clock cycle**	**Cross-over Type**	**Area-Delay Cost (Area × Latncy^2)**
XOR [18]	39	0.03	3	Multilayer	0.27

Circuit	Cell count	Area	Clock cycle	Cross-over Type	Area-Delay Cost (Area × Latncy^2)
XOR [19]	28	0.02	3	No wire crossing	0.18
XOR [20]	27	0.02	3	No wire crossing	0.18
XOR [21]	25	0.01	2	No wire crossing	0.04
XOR [22]	10	0.01	2	No wire crossing	0.04
XOR [23]	9	0.009	1	No wire crossing	0.009
XOR [24]	14	0.01	2	No wire crossing	0.04
XOR [25]	14	0.02	2	No wire crossing	0.08
XOR(proposed)	24	0.01	3	Multilayer	0.09

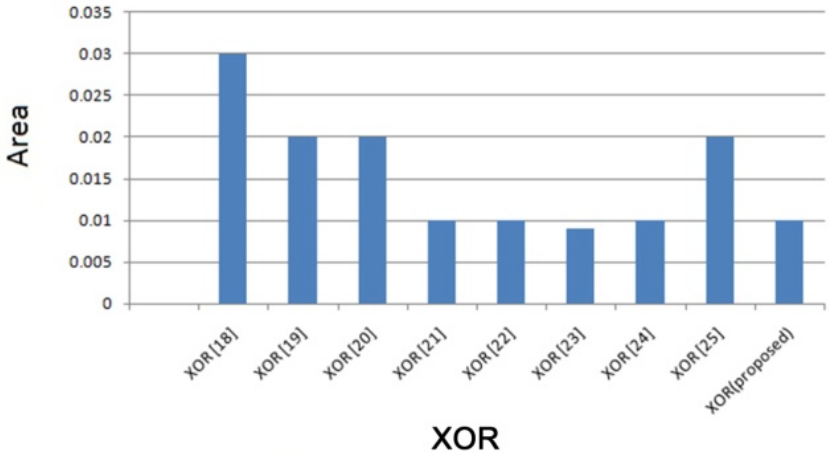


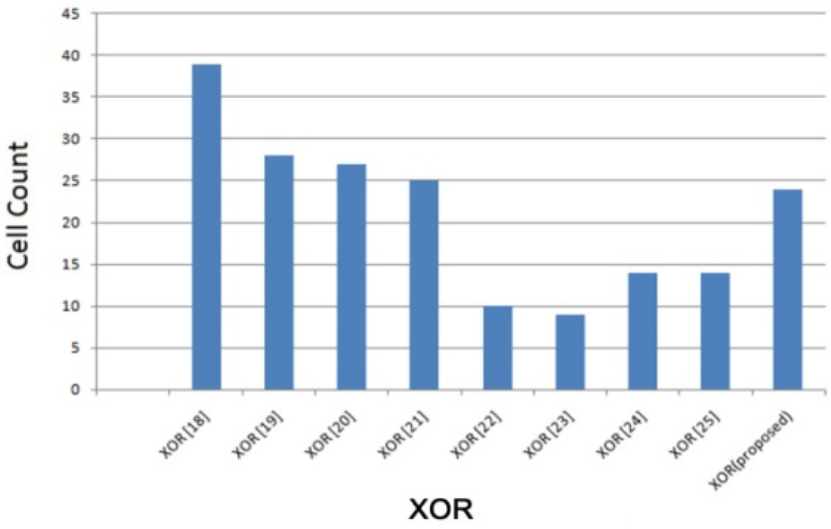


(a) (b)

(c) (d)

Fig.15 Graphical comparison study of half adder concerning (a) area (b) cell count (c)clock cycle (d) area-delay cost





(a) (b)

(c) (d)

Fig. 16 Graphical comparison of XOR concerning (a) area (b) clock cycle (c) cell count, (d)area-delay cost

6. Conclusion

A newly designed incrementer and decrementer circuit are studied in this paper using QCA. This paper shows the implementation of the newly designed half adder circuit cascading it 'n' times used to create 'n' bit ripple carry incrementer circuit. Less cell count, total area, and cell area are needed for this ripple carry incrementer and decrementer circuit. The minimum number of clock cycles used for constructing this incrementer/decrementer circuit is determined by calculating the circuit's complexity. The comparison with state-of-the-art designs is performed to explain the different design complexities. The simulation results of 4-bit, 8-bit, and 16-bit incrementer and decrementer circuits are per the theoretical results. This proposed circuit can be used for producing an address generation unit of microcontrollers and microprocessors.

References

1. C. S. Lent, P.D. Tougaw, W. Porod, G. H. Bernstein, Quantum Cellular Automata, Nanotechnology. 4 (1993) 49-57.
2. Y. Adelnia, A. Rezai, A novel adder circuit design in quantum-dot cellular automata technology, International Journal of Theoretical Physics, 58(2019) 184-200
3. H. Rashidi, A. Rezai, Design of novel multiplexer circuits in qcananocomputing. FactaUniversitatis, Series: Electronics and Energetics, 34 (2021) 105-114
4. M. N. Divshali, A. Rezai, S. S. F. Hamidpour, Design of novel coplanar counter circuit in quantum dot cellular automata technology, International Journal of Theoretical Physics, 58(2019), 2677-2691
5. I. E. Arani, A. Rezai, Novel circuit design of serial–parallel multiplier in quantum-dot cellular automata technology, Journal of Computational Electronics, 17(2018) 1771-1779.
6. M. G. Roshan, M. Gholami, 4-Bit serial shift register with reset ability and 4-bit LFSR in QCA technology using minimum number of cells and delay, Computers & Electrical Engineering, 78 (2019) 449-462.
7. M. Taheri,A. Arasteh, S. Mohammadyan, A. Panahi, K. Navi, A novel majority based imprecise 4: 2 compressor with respect to the current and future VLSI industry, Microprocessors and Microsystems, 73, (2020) 102962.

8. M. M. Abutaleb, Utilizing charge reconfigurations of quantum-dot cells in building blocks to design nanoelectronic adder circuits,Computers& Electrical Engineering, 86 (2020) 106712.doi: 10.1016/j.compeleceng.2020.106712.
9. S. Bi, W. Wang, A. Khalili,Multiplexer-based Binary Incrementer/decrementers. The 3rd IEEE-NEWCAS Conference. (2005) 219-222.
10. S.Sakthikumaran, S. Salivahanan, V. S.Kanchana, V.Bhaaskaran, B.B.Kavinilavu, A Very Fast and Low Power Incrementer and Decrementer Circuits, International Journal of Computer Communication and Information System (IJCCIS).2 (2010) 200-203.
11. C. Huang, J. Wang, Y. Hziang, A High-speed CMOS IncrementerDecrementer. In Circuits and Systems. 4 (2001) 88-91.
12. R. Hshemian, Highly parallel increment/decrement using CMOS technology. In Proceedings of the 33rd Midwest Symposium on Circuits and Systems, IEEE. (1990) 866-869.
13. J. C. Das, D. De, Novel low power reversible binary incrementer design using quantum-dot cellular automata, Microprocessors and Microsystems. 42 (2015) 10-23
14. A.Moustafa, Efficient Quantum-Dot Cellular Automata for Half Adder using Building Block, Quant. Inf. Rev. 7,(2019) 1-6
15. Y. Adelnia and A. Rezai, A Novel Adder Circuit Design in Quantum-Dot Cellular Automata Technology, International Journal of Theoretical Physics, 58 (2019) 184-200,
16. I. Gassoumi, L.Touil, B. Ouni, Design of efficient quantum Dot cellular automata (QCA) multiply accumulate (MAC) unit with power dissipation analysis. IET Circuits, Devices & Systems. 13(2019)534-543.
17. T. S. Rani, S. Karmakar, A. Metri , P. Sharan, Comparative Study of Half Adder and Subtractor Circuits based on Quantum-Dot Cellular Automata (QCA), International Journal of Innovative Research in Computer and Communication Engineering. 3 (2015) 61-67.
18. M. Mohammadi, S. Gorgin, and M. Mohammadi, Design of nonrestoring divider in quantum-dot cellular automata technology, IET Circuits, Devices Syst., vol. 11, no. 2, pp. 135–141, 2017
19. G. Singh, R.K. Sarin, B. Raj, A novel robust exclusive-OR function implementation in QCA nanotechnology with energy dissipation analysis. J. Comput. Electron. 15(2016) 455–465
20. J. C. Das, D. De, S. P. Mondal, A. Ahmadian, F. Ghaemi, and N. Senu, QCA Based Error Detection Circuit for Nano Communication Network, IEEE Access, 7(2019), 67355-67366, doi: 10.1109/ACCESS.20192918025.
21. H. R. Roshany, A. Rezai, Novel efficient circuit design for multilayer QCA RCA. International Journal of Theoretical Physics, 58(2019) 1745-1757
22. S. S. A. Ahmadpour, M. Mosleh, S. R. Heikalabad, A revolution in nanostructure designs by proposing a novel QCA full-adder based on optimized 3-input XOR, Physica B, Condensed Matter, 550(2018), 383-392.
23. H. B. Chen, H. Lv, Z. Zhang, X. Cheng, and G. Xie, Design and analysis of a novel low-power exclusive-OR gate based on quantum-dot cellular automata, Journal of Circuits, Systems and Computers, 28(2019), 1950141.
24. M. C. Chabi, A. Roohi, H. Khademolhosseini, S. Sheikhfaal, S. Angizi, K. Navi, and R. F. DeMara, Towards ultra-efficient QCA reversible circuits. Microprocessors and Microsystems, 49(2017), 127-138.
25. F. D. Ahmad, G. M. Bhat, H. Khademolhosseini, S. Azimi, S. Angizi and K. Navi, Towards single layer quantum-dot cellular automata adders based on explicit interaction of cells. Journal of Computational Science, 16(2016), 8-15.