Int. J. Nano Dimens., 9 (1): 68-78, Winter 2018

ORIGINAL ARTICLE

A fast wallace-based parallel multiplier in quantum-dot cellular automata

Hasan Faraji; Mohammad Mosleh*

Department of Computer Engineering, Dezfoul Branch, Islamic Azad University, Dezfoul, Iran

Received 24 July 2017; revised 24 October 2017; accepted 06 December 2017; available online 19 December 2017

Abstract

Physical limitations of Complementary Metal-Oxide-Semiconductors (CMOS) technology at nanoscale and high cost of lithography have provided the platform for creating Quantum-dot Cellular Automata (QCA)-based hardware. The QCA is a new technology that promises smaller, cheaper and faster electronic circuits, and has been regarded as an effective solution for scalability problems in CMOS technology. Therefore, it is possible to generalize QCA to all digital components. Multipliers are considered as one of the most important building blocks of computational circuits in digital processing systems. The traditional design of multipliers results in wasting the resources and increasing computational time. This paper presents an effective implementation of QCA parallel multiplier based on Wallace tree. It is able to significantly reduce the occupied area by reducing the number of QCA cells and therefore increases the speed of multiplying operation. The proposed QCA multiplier is simulated by QCADesigner2.0.3 software. The simulation results confirm that the proposed QCA multiplier works well and can be used in high performance circuits in QCA technology. Moreover, the proposed QCA multiplier has less complexity and occupied area compared to other QCA multiplier designs.

Keywords: *Full adder; Half adder; Multiplier; Nanotechnology; Quantum-dot cellular automata; Wallace tree.*

How to cite this article

Faraji H, Mosleh M. A fast wallace-based parallel multiplier in quantum-dot cellular automata. Int. J. Nano Dimens., 2018; 9 (1): 68-78.

INTRODUCTION

Because of future needs to design some transistors with dimensions less than 32 nanometers (nm), CMOS technology is expected to concern with challenges such as short-circuit channel, gate control reduction, exponential current leakage, heat and etc. Therefore, emerging technologies like Single Electron Transistor (SET), Resonant Tuning Diode (RTD), Carbon Nanotube Field Effect Transistor (CNTFET) and Quantum-dot Cellular Automata (QCA) have been able to overcome the mentioned problems. According to International Technology Roadmap for Semiconductor (ITRS), which provides a summary of future technologies, QCA is regarded as one of efficient solutions. QCA is a recently emerged technology that can be considered as a suitable replacement for transistor-based circuits with smaller size, higher speed and lower power consumption [1]. This technology operates by a certain physical phenomenon, called columbic repulsion, which uses the location of electron pairs instead of voltage levels for logic states [2].

There have been a lot of researches in circuit design field. Arithmetic elements such as sequential circuits, shifters, comparators, adders and array multipliers have been designed using QCA technology [3-4]. Among these, multipliers have elements with the highest area occupy, time and power consumption. So far, a variety of mechanisms for multiplying operation have been offered, all of which have common basis; they have formed by a series of partial products and their summations. It is worth mentioning that there are some techniques for reducing the number of these multiplications and summations and as a result, speeding up the multiplication operation. Designing multiplier circuits in QCA technology has less been considered by QCA designers; due to density of logic elements and highly complex

* Corresponding Author Email: *mosleh@iaud.ac.ir*

design. K. Walus *et al.* (2003) offered the first 2-bit serial multiplier in QCA technology [5]. S-W Kim and E-E. Swartzlander (2009) presented a quasi-modular approach to optimize parallel multipliers [6]. They used Wallace and Dadda trees to calculate the final sum. H. Cho and E-E Swartzlander (2009) presented QCA serial parallel multipliers using a new serial adder, called Carry Flow Adder (CFA) [7].

The optimum CFA had been designed based on unique features of QCA technology. P. Vijayalakshmi and K. Kirthika (2012) proposed a Wallace-based parallel multiplier using a full adder, consisted of three majority gates [8]. In order to calculate the final sum in Wallace tree, they used a Carry Select Adder (CSA) [8]. L. Lu *et al.* (2013) proposed a parallel multiplier based on two-dimensional systolic array [9]. In that systolic structure, each processing element consisted of one CFA adder, one multiplier and one multiplexer. S. Basu and A. Bal (2014) introduced a multiplier in which every building block was able to calculate one partial product [10]. Each building block consisted of one full adder and one AND gate implemented using three-input majority gates.

The studies show that one of most effective ways to reduce the complexity of multiplying includes using Wallace tree method. Due to its parallel nature and tree-like structure, this method is capable to reduce the number of multiplications and their summations and thus, increase the efficiency of the multiplier. This paper presents a Wallace tree-based parallel QCA multiplier which is able to optimize complexity, delay and occupied area, compared to the previous works using suggested full adder [11].

This paper is organized as follows. Introduction to QCA technology is provided by background concepts section. In the proposed method section, at first, Wallace-based parallel multiplier is introduced and then the proposed parallel QCA multiplier is presented. Simulation results and comparisons are given in a results and discussions section. Finally, this paper ends with a conclusion section.

EXPERIMENTAL

Background Concepts

- QCA Architecture

In this subsection, some basic concepts of QCA technology, such as cells, logic gates, wire and clock pulse are introduced.

- QCA Cell

The simplest element in QCA technology is a square cell which has four quantum-dots with minimum energy in four corners of the square. In this cell, two electrons are located in quantum-dots such that they constantly have minimum energies. A quantum-dot is a place in QCA cell in which an electron can be placed. These two electrons placed within a cell can tunnel between quantum-dots and determine the level of tunnel junctions, capacity of island capacitor and charging energy of dots. Tunneling everywhere, these electrons change their places between vertices of the square. Due to columbic repulsion force, these electrons are placed at the farthest distance from each other in diagonal direction of the square, and have two steady states.

Although the electrons within a cell can change their places by tunneling, they can't leave the cell and transfer to another cell. It is worth mentioning that in absence of external forces, the columbic repulsion force between electrons repels electrons to the corner of the cell so that they place at the farthest distance from each other. Accordingly, regarding the positions of electrons within a cell, two states occur as shown in Fig. 1. These two states for the positions of charges create two different polarizations (P= -1 and P=+1) for the cell, which can be regarded as two binary values; '0' and '1' [12-13].

- Wire Crossing in QCA

QCA cells can be placed close to each other to create a chain of cells, called QCA wire. Due to repulsion property of quantum forces, the polarity of a cell transfers to its neighbor cell when these cells are placed next to each other and then, a QCA wire is built. As can be seen in Fig. 2, in QCA technology, there are two types of wires: simple binary wire and inverter chain. The former can transfer input polarity to the output, without any change. On the other hand, inverter chain can generate input polarity or its inverse in the odd and even cells, respectively. Generally, there are two techniques for wire crossing in QCA technology; the first one is a coplanar technique in which crossing two wires is implemented in the same layer so that simple binary wire intersects an inverter chain. The other technique includes multilayer crossing in which each wire is located in a separated layer. Fig. 3 shows two techniques of coplanar and multilayer crossing [12, 14].

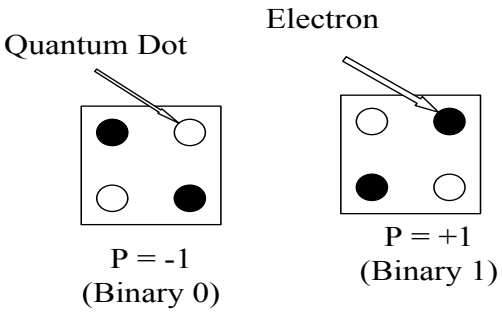


Fig. 1: QCA cells.

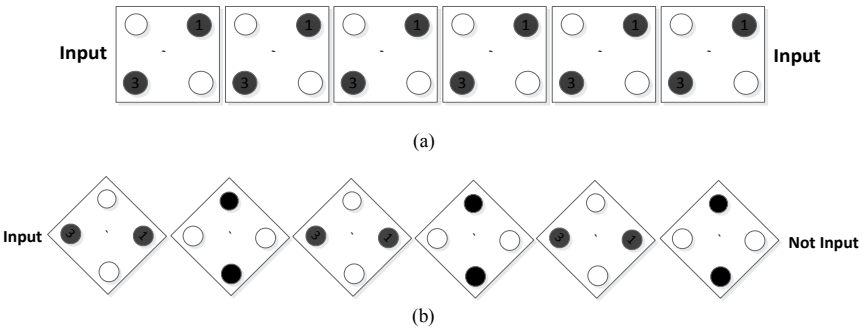


Fig. 2: (a) simple binary wire, (b) inverter chain.

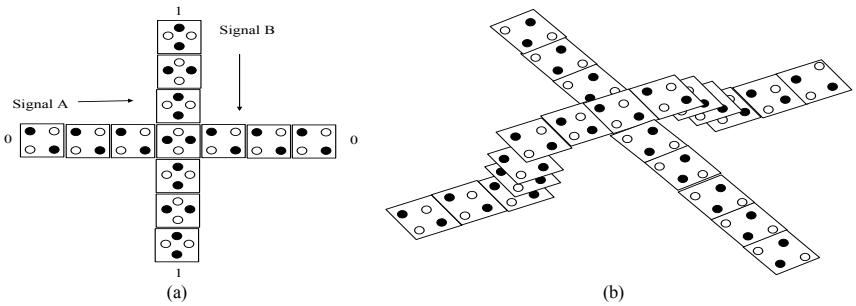


Fig. 3: (a) coplanar Wire, (b) multilayer Wire.

- Basic Gates in QCA

There are two conventional basic gates in QCA technology which are used in many designs: inverter (NOT) and three-input majority gates. A logical representation of inverter gate with two simple and resistant cellular designs has been shown in Fig. 4. As can be seen, the cellular design presented in Fig. 4(b) is resistant and has a high stability and reliability [15-16]. Another basic gate is three-input majority gate for which the logical function is given by Eq.1. Also, its logical and cellular designs have been shown in Fig. 5.

$$MV3(A, B, C) = AB + AC + BC \quad (1)$$

It should be noted that if we consider one of majority gate inputs to be equal to '0' or '1', we will have a two-input AND or a two-input OR gate, respectively.

- Clock in QCA

In QCA-based circuits, a clock signal generated by an electric field is applied to cells for either raising or downing the tunneling barrier between dots within a cell. One period of clock signal in QCA technology is divided into four phases respectively called switch, hold, release and relax. In the switch phase, tunneling barriers begins to increase

between the two dots of a quantum cell. In the second phase, namely hold, the barriers are held high and the cells of first area can act as stimulus for cells of the next area. In the third phase, release, tunneling barriers begin to decrease and cells are losing their polarities. In the last phase, relax, there are no barriers between the dots and cells are without polarity. Fig. 6 shows the clocking phases [17-18].

Proposed Method

- Wallace-based Parallel Multiplier

A multiplier is an electronic circuit applied in digital systems to multiply two binary numbers. It is built using binary adders. A variety of computer arithmetic techniques can be used to implement a digital multiplier. Most techniques involve computing a set of partial products, and then summing up them. There are some methods able to sum up the partial products in an effective way. One of efficient methods for accelerating the multiplication operation includes using Wallace tree technique. This technique is capable to create an effective hardware consisting of full adder and half adder circuits in three stages, which performs the multiplication operation in parallel. Wallace tree, presented by C. Wallace in 1964, has been considered as an effective hardware implementation for multiplying two numbers [19]. The Wallace-based multiplying operation includes three stages:

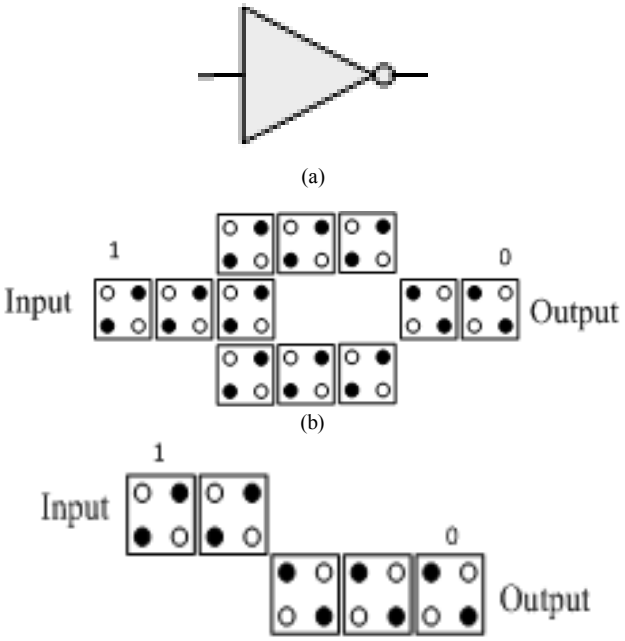


Fig. 4: Inverter gate,(a) Schematic representation, (b) simple layout, (c) resistant layout.

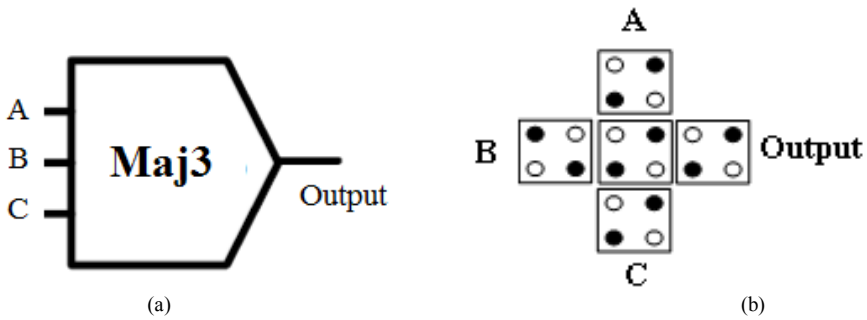


Fig. 5: Three-input majority gate, (a) logical gate, (b) layout.

Stage1. Multiply (or better expressing, AND) each bit of multiplicand by each bit of multiplier, yielding n^2 partial products.

Consider two n-bit numbers $A= (a_{n-1},..., a_1, a_0)$ and $B= (b_{n-1},..., b_1, b_0)$. Multiplying these two numbers results in a 2n-bit value as follows:

$$P = A \times B = \sum_{j=0}^{n-1} a_j 2^j \times \sum_{i=0}^{n-1} b_j 2^i = \sum_{k=0}^{2n-1} p_k 2^k \quad (2)$$

The final product, *P,* is computed through the partial products as shown in Fig. 7.

Stage2. Reduce the number of partial products by the layers of a Full Adder (FA) and a Half Adder (HA).

Stage3. Adding two n-sets resulted from the second stage to an n-bit adder. It should be noted that the second stage is carried out as follows. As long as there are three or more bits with the same value, add a following layer:

Three bits of the same value enter into FA and as a result, two bits with different values are produced (one bit with the same value and one bit with a higher value).

If two bits with the same value remain, put them into an HA.

If there is just one bit, transfer it to the next layer.

The proposed QCA Wallace-based multiplier

In this section, an effective implementation of 4-bit QCA multiplier based on Wallace tree, is proposed. Details of proposed QCA multiplier are

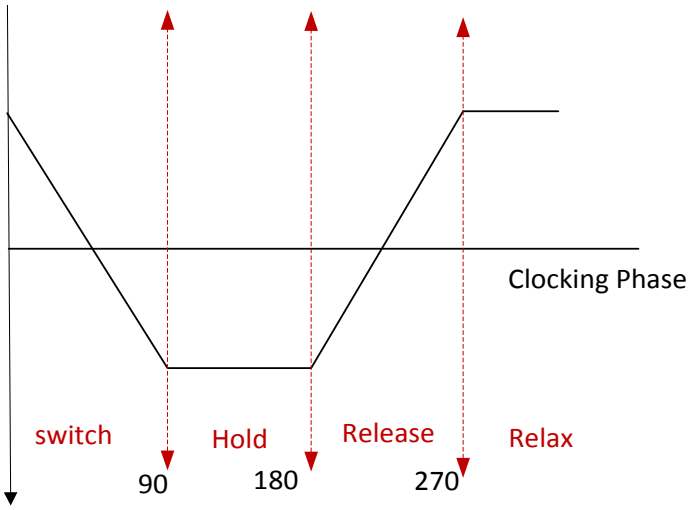


Fig. 6: Clock signal in a the QCA technology.

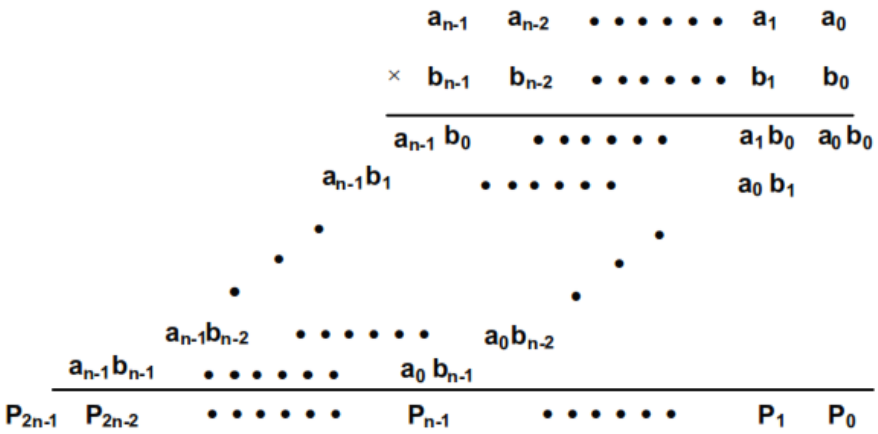


Fig. 7: Multiplying two n-bit numbers.

given as follows. As shown in Fig. 7, the product of two 4-bit numbers $A=(a_3 \ldots a_1, a_0)$ and $B=(b_3 \ldots b_1, b_0)$ results in 16 partial products. Each partial product is created by an AND gate which is implemented by a 3-input majority gate in which one of inputs is considered as '0'. Implementing partial products with 3-input majority gates is shown in Fig. 8 [20].

As mentioned in previous section, in the second stage of Wallace-based multiplier, all three partial products with the same value are categorized and fed into full adders. If two partial products with the same value remain, a half adder is used. In addition, if only one partial product remains, it will be transferred directly to the next layer.

Finally, in the last step, two bit-groups resulted from the second stage are added to a 4-bit carry ripple adder.

The process of multiplying two 4-bit numbers, based on Wallace technique has been illustrated in Fig. 9.

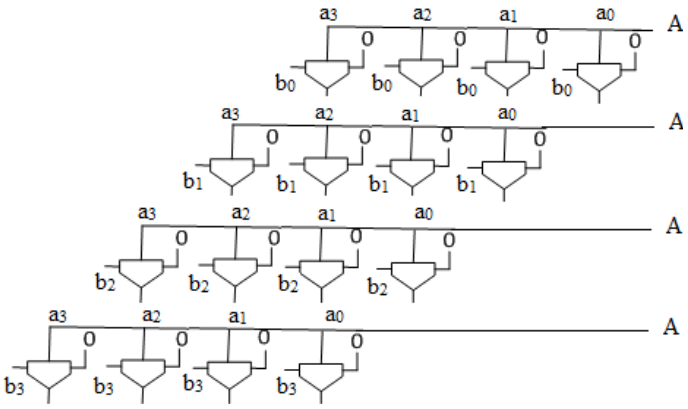


Fig. 8: Producing 16 partial products of a 4-bit multiplier using 3-input majority gates.

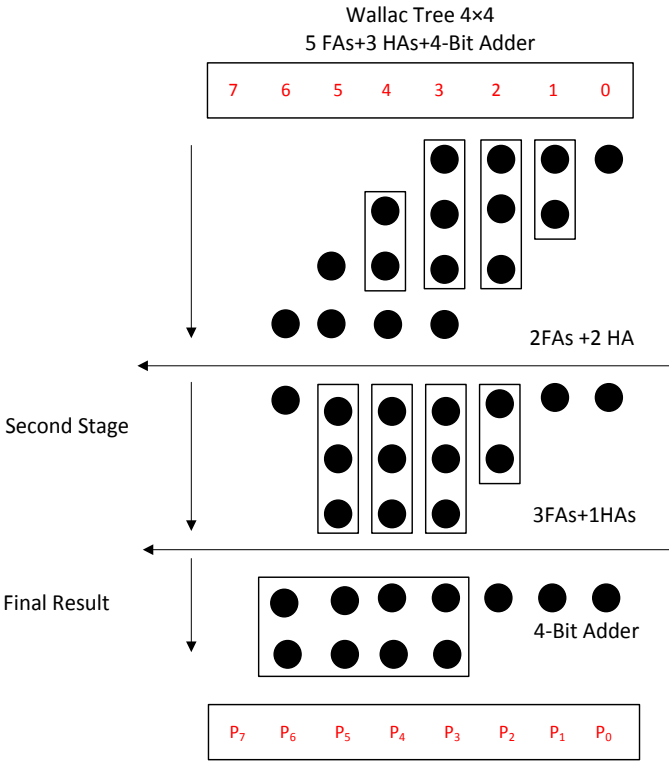


Fig. 9: Multiplying two 4-bit numbers using Wallace technique.

As seen in Fig. 9, Wallace-based multiplier needs five full adders, three half adders and one 4-bit adder. Since implementing a 4-bit ripple carry adder is possible by using one half adder and three full adders, we can implement a Wallace-based multiplier with eight full adders and four half adders. The proposed Wallace multiplier can be easily generalized to design 8-bit, 16-bit and n-bit multipliers. Applying an appropriate adder in QCA technology can help implementing the Wallace-multiplier, effectively. For this reason, it uses the adder cited in [11]. Low delay and regular structure are considered among the properties of this adder. So, applying this full adder in modular designs decreases total delay and occupied area.

The full adder is formed based on one 5-input majority gate and one new CMVMIN gate; their logical and layout designs are shown in Fig 10.

The CMVMIN gate has three inputs *A*, *B*, and *C* and two outputs *O1*, and *O2* which are defined as below:

$$O1 = A'B' + A'C' + B'C' \quad (3)$$

$$O2 = AB + AC + BC \quad (4)$$

The output *O2* in CMVMIN gate is equal to the carry output of full adder.

Moreover, the output of 5-input majority gate is defined as Eq.5.

$$\text{Maj-5 (A, B, C, D, E)} = ABC + ABD + ABE + ACD + ACE + ADE + BCD + BCE + BDE + CDE \quad (5)$$

Therefore, the sum output of full adder is also obtained through combination of Eq. (3) and Eq.(5):

$$\text{Maj-5 (A, B, C, O1, O1)} = ABC + AB'C' + A'BC' + A'B'C = \sum m(1, 2, 4, 7) \quad (6)$$

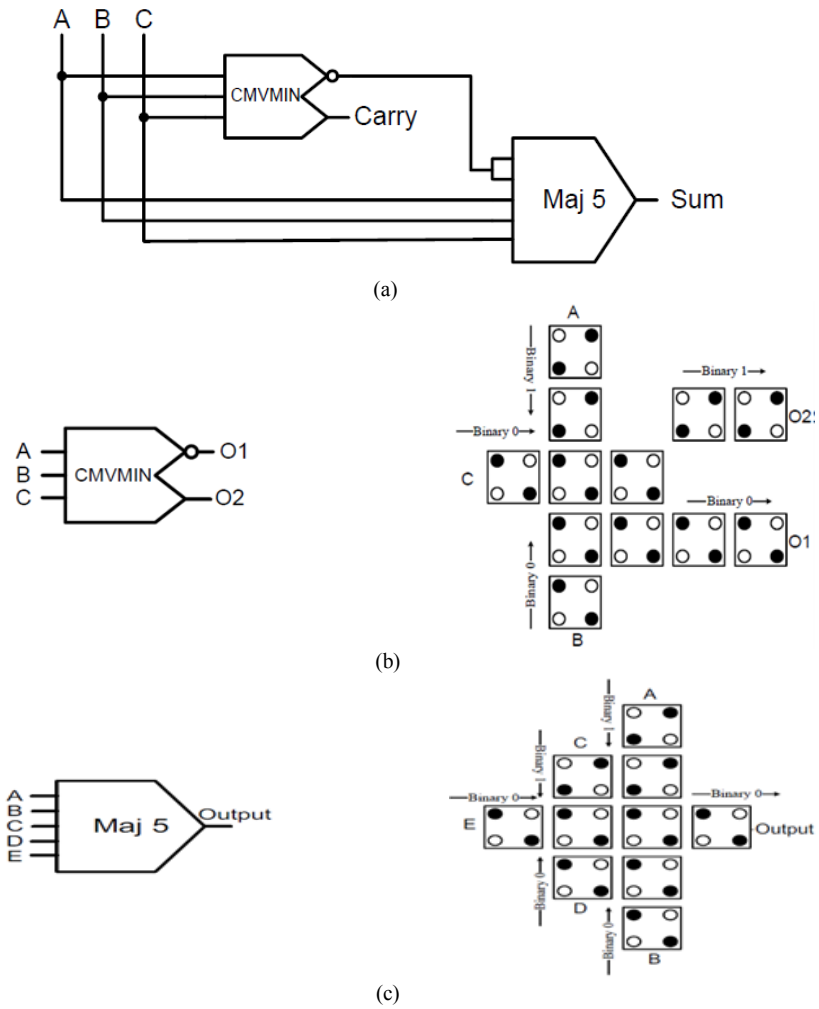


Fig. 10: (a) Schematic representation of applied full adder [11], (b) logical and layout of CMVMIN gate, (c) logical and layout of 5-input majority gate.

RESULTS AND DISCUSSIONS

In this section, 4-bit Wallace-based QCA multiplier, previously described in detail, is simulated using QCA Designer 2.0.3. All parameters and conditions for this simulation which are determined based on default values in QCA Designer include the size of quantum cells, dimension of quantum-dots and the center-to-center distance between neighbor cells and are considered to be 18 × 18, 5 nanometers and 20 nanometers, respectively. As shown in Fig. 11, the layout of applied full adder has only 48 cells and needs two clock phases to generate sum and carry outputs. This full adder is implemented in three layers [11]. Fig. 12 shows three layers of the applied full adder, respectively. It should be mentioned that there are six cells in layer 2 which connect layer 1 to layer 3. The mentioned full adder has a delay as 2 clock pulses and comprises of 48 cells. The results of simulation for all combinations of inputs, *A*, *B* and *C* are shown in Fig. 13. These results confirm that the output of this full adder which operates well, are obtained after two clock pulse phases. For instance, for inputs *A=0*, *B=1* and *C=0*, the true outputs *Sum=1* and *Carry=0* are obtained. Moreover, it occupies an area as 0.019 μm^2 [11]. Fig. 14 shows the layout of proposed 4-bit QCA multiplier. This design consists of 16 3-input majority gates, 12 CMVMIN gates and 12 5-input majority gates and has totally 2900 cells, occupying an area as 3.62. The delay of this multiplier is 14 clock phases. Simulation of the proposed 4-bit QCA multiplier is shown in Fig. 15.

A comparison among the proposed 4-bit QCA multiplier and all QCA multiplier designs is carried out in this paper to evaluate the features of proposed multiplier. The evaluation results are illustrated in Table 1.

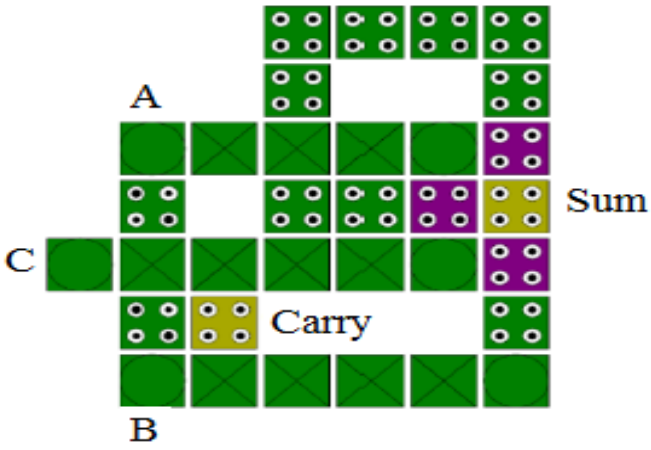


Fig. 11: Layout of the applied full adder [11].

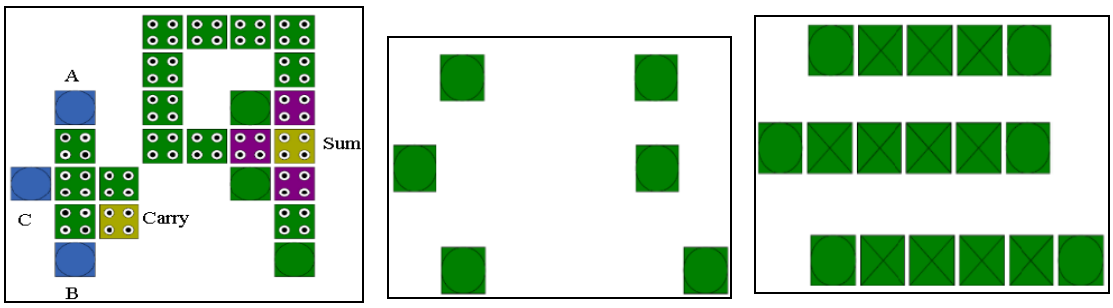


Fig. 12: Three different layers of the applied full adder [11].

It should be noted that the complexity of circuits is expressed based on the number of used cells, occupied area in the scale of μm^2 the delay in terms of number of clock cycle and the ratios of QCA multiplier architecture parameters proportion to those equivalent parameters extracted from each reference.

As can be clearly seen in Table 1, the most complex multiplier was already presented in L. Lu *et al.* [9] which has 36025 cells and occupies an area as 92.50 μm^2. The proposed multiplier has the lowest complexity with only 2900 cells and occupies an area as 3.69 μm^2. Although the delay of proposed multiplier is 4 clock phases more than the best delay and its corresponding ratio parameter is 1.36 better.

The percentage of cell number reduction by multipliers is compared in Fig. 16.

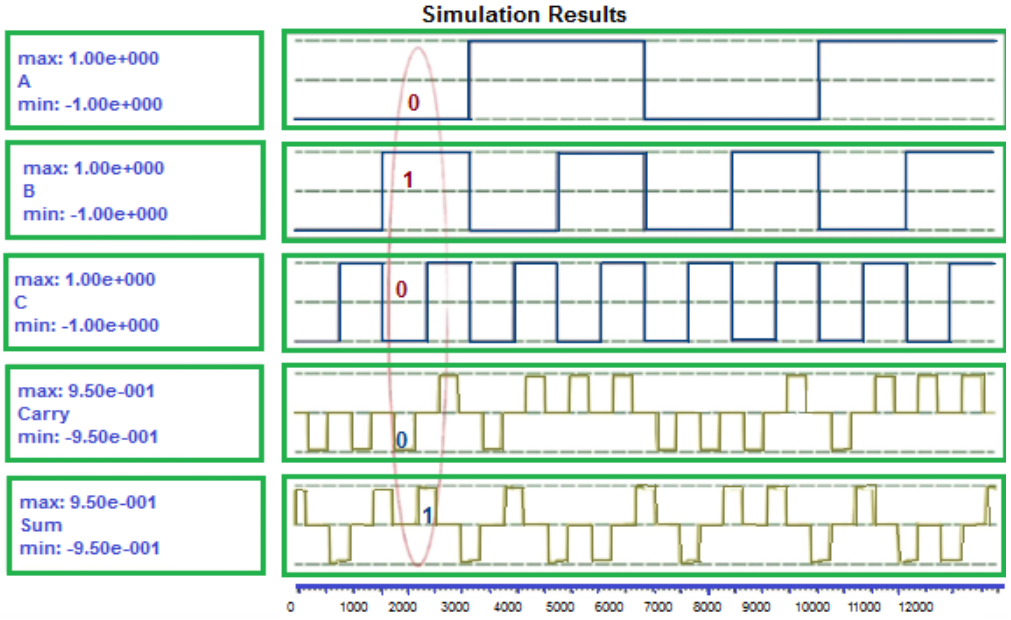


Fig. 13: The simulation results obtained for applied full adder.

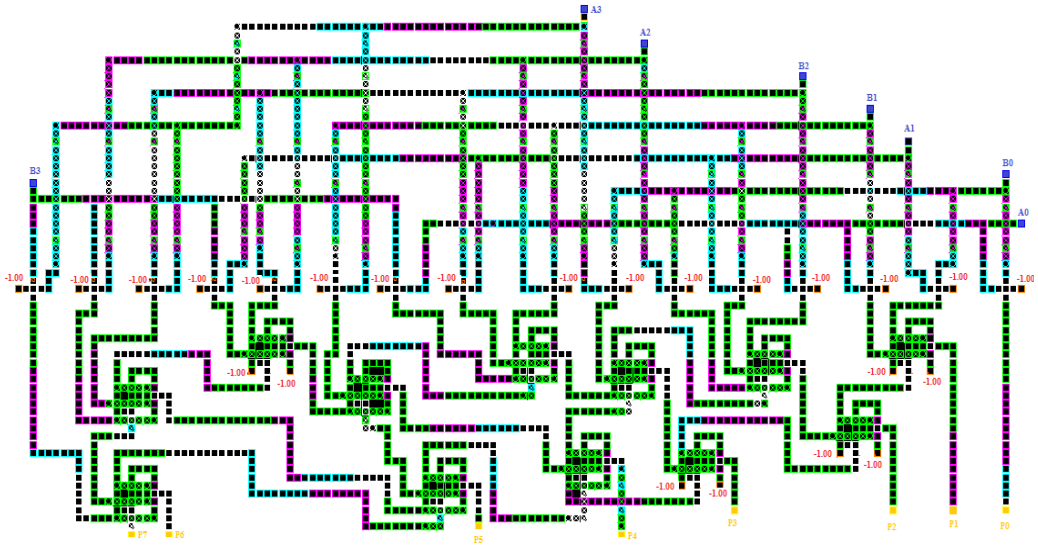
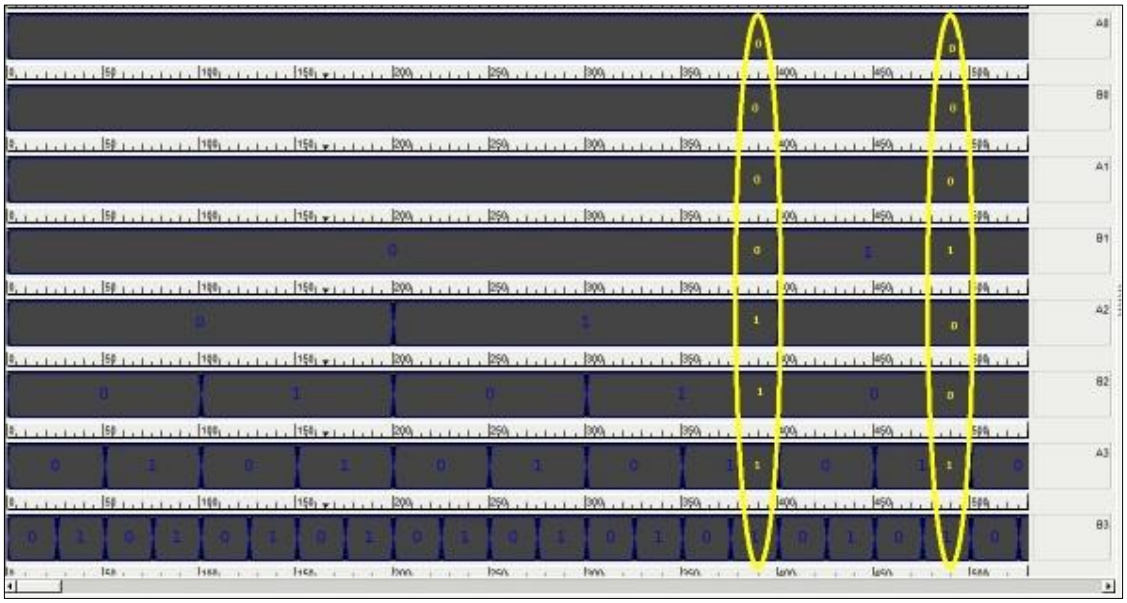
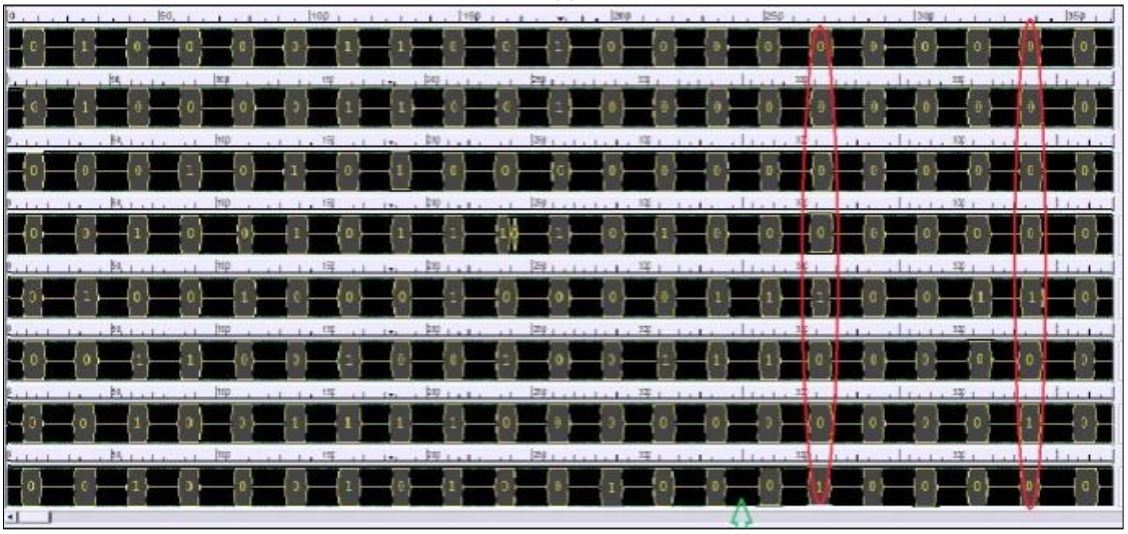


Fig. 14: Layout of the suggested 4-bit Wallace-based QCA multiplier.



(a)



(b)

Fig. 15: The simulation of proposed 4-bit QCA multiplier (a) inputs (b) outputs.

Table 1: Comparing proposed multiplier with other ones.

Multipliers	Complexity (#Cells)	Ratio	Area (μm^2)	Delay (Clock phases)
S-W. Kim and E. E.Swartzlander,2009 [6], (Wallace-based multiplier)	3295	1.136	7.39 μm^2	10
S-W. Kim and E. E.Swartzlander,2009 [6] (Array-based multiplier)	3738	1.29	6.02 μm^2	14
P. Vijayalakshmi and N. Kirthika ,2012[8]	-	-	155μm^2	24
L. Lu et al. 2013[9]	36025	12.42	92.50μm^2	33.5
Proposed multiplier	2900	1.00	3.69 μm^2	14

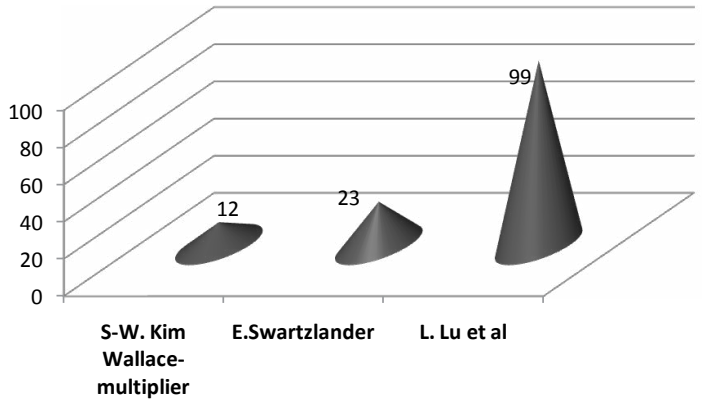


Fig. 16: Comparing the percentage of reduction in number of cells.

CONCLUSION

Multipliers play important roles in designing computational circuits. This paper has presented an efficient 4-bit Wallace-based QCA parallel multiplier using our proposed full adder [11]. The proposed QCA multiplier has 2900 cells and occupies an area of 3.96 μm^2. Moreover, its delay is 14 clock phases. The results of simulations show that the proposed QCA multiplier has better performance than the previous works; in terms of cell count and occupied area. In addition, the proposed multiplier has been evaluated and compared in terms of delay. The results show that the delay of proposed multiplexer is very close to the best available design.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interests regarding the publication of this manuscript.

REFERENCES

[1] Bourianoff G., (2003), The future of nanocomputing. *Computer*. 36: 44-53.

[2] Haron N. Z., Hamdioui S., Cotofana S., (2009), Emerging non-CMOS nanoelectronic devices-What are they?. *4th IEEE International Conference on Nano/Micro Engineered and Molecular Systems*. 63-68

[3] Vetteth A., Walus K., Dimitrov V. S., Jullien G. A., (2002), Quantum-dot cellular automata carry-look-ahead adder and barrel shifter. *IEEE Emerging Telecommunications Technologies Conference*. 2-4.

[4] Lakshmidevi K., Jordhana P. D., (2015), A novel full comparator design using quantum-dot cellular automata. *Int. J. VLSI system Design and Communic. Sys*. 3: 603-608

[5] Walus K., Jullien G., Dimitrov V., (2003), Computer arithmetic structures for quantum cellular automata. *Conf. Record of the Thirty-Seventh Asilomar Conf. on Signals. Sys. Comput.* 1435-1439.

[6] Kim S.-W., Swartzlander E. E., (2009), Parallel multipliers for quantum-dot cellular automata. *IEEE Nanotechnol. Mater. and Devices Conf.* 67-72

[7] Cho H., Swartzlander Jr E. E., (2009), Adder and multiplier design in quantum-dot cellular automata. *IEEE Transact. Comput*. 58: 721-727

[8] Vijayalakshmi P., Kirthika N., (2012), Design of hybrid adder using QCA with implementation of wallace tree multiplier. *Int. J. Adv. Eng. Technol.* 3: 202-215.

[9] Lu L., Liu W., O'Neill M., Swartzlander E. E., (2013), QCA systolic array design. *IEEE Transact. Comput.* 62: 548-560.

[10] Basu S., Bal A., (2014), Realization of combinational multiplier using quantum cellular automata. *Int. J. Comput. Applic.* 99: 1-6.

[11] Bandani-sousan H. A., Mosleh M., Setayeshi S., (2015), Designing and implementing a fast and robust full-adder in quantum-dot cellular automata (QCA) technology. *J. Adv. Comput. Res.* 6: 27-45.

[12] Lent C. S., Tougaw P. D., Porod W., Bernstein G. H., (1993), Quantum cellular automata. *Nanotechnol*. 4: 49-57.

[13] Lent C. S., Tougaw P. D., (1997), A device architecture for computing with quantum dots. *Proceed. IEEE*. 85: 541-557.

[14] Angizi S., Alkaldy E., Bagherzadeh N., Navi K., (2014), Novel robust single layer wire crossing approach for exclusive or sum of products logic design with quantum-dot cellular automata. *J. Low Power Electronics*. 10: 259-271.

[15] Bubna M., Mazumdar S., Roy S., Mall R., (2007), Designing cellular automata structures using quantum dot cellular automata. *14th Annual IEEE Int. Conf. High Perf. Computing*.

[16] Javid M., Mohamadi K., (2009), Characterization and tolerance of QCA full adder under missing cells defects. *Fifth Int. Conf. MEMS, NANO, and Smart Systems*. 85-88.

[17] Lakshmi S. K., (2010), Efficient design of logical structures and functions using nanotechnology based quantum dot cellular automata design. *Int. J. Comput. Applic*. 3: 35-42

[18] Lent C. S., Liu M., Lu Y., (2006), Bennett clocking of quantum-dot cellular automata and the limits to binary logic scaling. *Nanotechnol*. 17: 4240-4251.

[19] Wallace C. S., (1964), A suggestion for a fast multiplier. *IEEE Transact. Electronic Comput.* 1: 14-17.

[20] K'andrea C. B., Schulte M. J., Swartzlander E. E., (1995), Parallel reduced area multipliers. *J. VLSI Signal Process. Systems for Signal, Image and Video Technol*. 9: 181-191.