Quantum-Dot Cellular Automata Adders

Wei Wang
Department of Electrical & Computer Eng.
University of Western Ontario
London, ON, Canada

Konrad Walus and G. A. Jullien
Department of Electrical & Computer Eng.
University of Calgary
Calgary, AL, Canada

Abstract— **In this paper, a novel quantum-dot cellular automata (QCA) adder design is presented that reduces the number of QCA cells compared to previously reported designs. The proposed one-bit QCA adder structure is based on a new algorithm that requires only three majority gates and two inverters for the QCA addition. By connecting *n* one-bit QCA adders, we can obtain an *n*-bit carry look-ahead adder with the reduced hardware while retaining the simple clocking scheme and parallel structure of the original carry look-ahead approach. The proposed adder is designed and simulated using the QCADesigner tool for the four-bit adder case. The proposed design requires only about 70% of the hardware compared to previous designs with the same speed and clocking performance.**

Keywords- quantum-dot cellular; adder; carry look-ahead;

I.   INTRODUCTION

Quantum-dot cellular automata (QCA) has been recognized as one of the technologies that may replace field-effect transistor (FET)-based computing devices at the nano-scale level. In QCA, binary information is encoded in the charge configuration within quantum dot cells. Many studies have reported that QCA can achieve high device density, extremely low power consumption, and very high switching speed [1], [2]. First proposed in 1993 by Lent, et. al and fabricated in 1997, QCA is expected to play an important role in nanotechnology research [3], [4].

Recently, QCA adders have been studied intensively [5]-[7]. Initial adder designs were constructed with five majority gates (a fundamental QCA logic gate) and three inverters [5]. By connecting *n* such one-bit QCA full adders, a carry look-ahead (CLA) adder can be obtained, since the carry is generated before the sum in the QCA adder [6]. A new bit-serial QCA adder has also been proposed [7] that uses carry feedback and only requires three majority gates and two inverters. However, this bit-serial approach requires a complicated clocking scheme and feedback control.

In this paper, a novel QCA adder design is presented that reduces the number of QCA cells when compared to previously reported designs. We demonstrate that it is possible to design a CLA QCA one-bit adder, with the same reduced hardware as the bit-serial adder, while retaining the simpler clocking scheme and parallel structure of the original CLA approach. The proposed structure is based on a new algorithm that requires only three majority gates and two inverters for the QCA addition. It is noted that the bit-serial QCA adder uses a variant of the proposed one-bit QCA adder. By connecting *n*

proposed one-bit QCA adders, we can obtain an efficient *n*-bit QCA adder with CLA.

The rest of the paper is organized as follows. In Section II, we introduce some background material on QCA technology. In Section III we propose a new QCA addition algorithm and the corresponding one-bit QCA adder structure that reduces the number of majority gates and inverters required by existing designs [5],[6]. Then we demonstrate that, using this structure, we can obtain efficient *n*-bit CLA QCA adders. In Section IV we use simulation results obtained from QCADesigner [6] to compare our new adder design with previously published designs [5]-[7]. Finally, we conclude the paper in Section V.

II.   BACKGROUND MATERIALS

A.   QCA Basics

QCA is based on the interaction of bi-stable QCA cells constructed from four quantum-dots. A high-level diagram of two polarized QCA cells is shown in Fig. 1. Each cell is constructed from four quantum dots arranged in a square pattern. The cell is charged with two electrons, which are free to tunnel between adjacent dots. These electrons tend to occupy antipodal sites as a result of their mutual electrostatic repulsion. Thus, there exist two equivalent energetically minimal arrangements of the two electrons in the QCA cell as shown in Fig. 1. These two arrangements are denoted as cell polarization $P = +1$ and $P = -1$ respectively. By using cell polarization $P = +1$ to represent logic "1" and $P = -1$ to represent logic "0", binary information can be encoded.
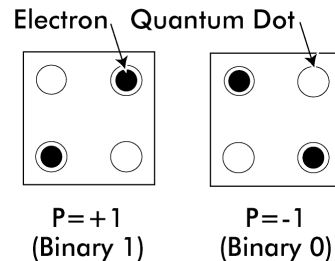

Figure 1.   QCA cells

Arrays of QCA cells can be arranged to perform all logic functions. This is due to the Coulombic interactions, which influences the polarization of neighbouring cells. QCA architectures have been proposed with potential barriers between the dots that can be controlled and used to clock QCA circuits [2]-[7].

B. QCA Logical Devices

The fundamental QCA logic devices are the QCA wire, majority gate and inverter.

QCA wire: In a QCA wire, the binary signal propagates from input to output because of the Coulombic interactions between cells. This is a result of the system attempting to settle to a ground state. Any cells along the wire that are anti-polarized to the input would be at a higher energy level, and would soon settle to the correct ground state. The propagation in a 90-degree QCA wire is shown in Fig. 2. Other than the 90-degree QCA wire, a 45-dgree QCA wire can also be used. In this case, the propagation of the binary signal alternates between the two polarizations. Further, there exists a so-called non-linear QCA wire, in which cells with 90-degree orientation can be placed next to one another, but off center.
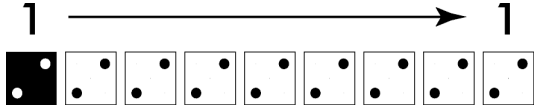

Figure 2. A QCA wire (90-degree)

Majority gate and inverter: The majority gate and inverter are shown in Fig. 3 and Fig. 4 respectively. The majority gate performs a three-input logic function. Assuming the inputs are A, B and C, the logic function of the majority gate is

$$m(A,B,C) = A \cdot B + B \cdot C + A \cdot C \qquad (1)$$

By fixing the polarization of one input as logic "1" or "0", we can obtain an OR gate and an AND gate respectively. More complex logic circuits can then be constructed from OR and AND gates.
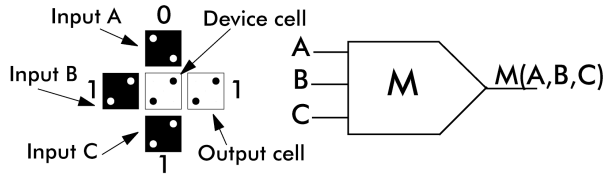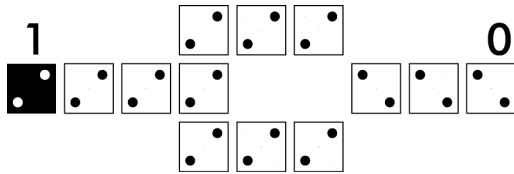

Figure 3. A QCA majority gate

Figure 4. A QCA inverter

C. QCA Adders

The University of Notre Dame first proposed the design of a one-bit full adder. As shown in Fig. 5, it consists of five majority gates and three inverters [5], [6]. By connecting n such one-bit QCA full adders, a n-bit CLA adder can be obtained, since the carry is generated before the sum in the QCA adder. Recently, a new bit-serial QCA adder has also been proposed [7] that uses carry feedback and requires three

majority gates, two inverters and feedback control. However, this bit-serial approach requires a complicated clocking scheme. In this paper we demonstrate that it is possible to design a QCA one-bit adder, with the same reduced hardware as the bit-serial adder, while retaining the simpler clocking scheme and parallel structure of the original carry look-ahead approach.
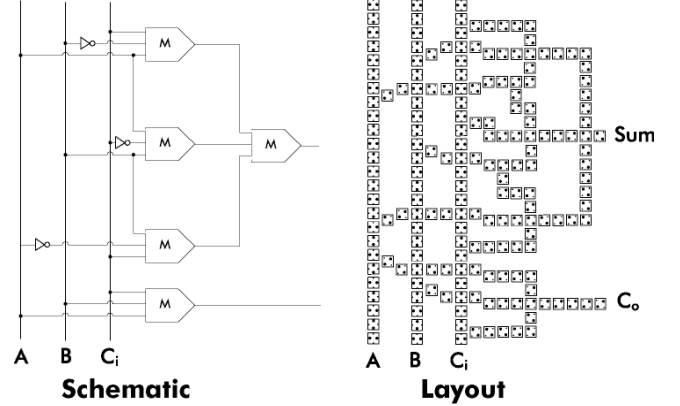

Figure 5. One-bit QCA full adder [5], [6]

III. PROPOSED QCA ADDERS

In this section, we first propose a new QCA addition algorithm and the corresponding one-bit QCA adder structure that reduces the number of the majority gates and inverters required by existing designs [5]-[6]. Then, we demonstrate that, using this structure, we can obtain efficient carry look-ahead n-bit QCA adders.

A. New QCA Addition Algorithm

A one-bit full adder is defined as follows:

Inputs: Operand bits a, b and carry-in c_{in}

Outputs: Sum bit s and carry-out c_{out}

$$s = a \cdot b \cdot c_{in} + \bar{a} \cdot \bar{b} \cdot c_{in} + \bar{a} \cdot b \cdot \bar{c}_{in} + a \cdot \bar{b} \cdot \bar{c}_{in} \qquad (2a)$$

$$c_{out} = a \cdot b + a \cdot c_{in} + b \cdot c_{in} \qquad (2b)$$

Proposition 1: By using the majority function (1), we get the QCA addition algorithm as

$$c_{out} = m(a,b,c_{in}) \qquad (3a)$$

$$\overline{c_{out}} = m(\bar{a},\bar{b},\bar{c}_{in}) \qquad (3b)$$

$$s = m(\overline{c_{out}},c_{in},m(a,b,\bar{c}_{in})) \qquad (3c)$$

Proof:

By using the majority function (1) and the equation (2b), we get

$$c_{out} = a \cdot b + a \cdot c_{in} + b \cdot c_{in} = m(a,b,c_{in})$$

$$\overline{c_{out}} = \bar{a} \cdot \bar{b} + \bar{a} \cdot \bar{c}_{in} + \bar{b} \cdot \bar{c}_{in} = m(\bar{a},\bar{b},\bar{c}_{in})$$

Then, the equation (2a) can be rewritten as

$$s = (a \cdot b + \bar{a} \cdot \bar{b})c_{in} + (\bar{a} \cdot b \cdot \overline{c_{in}} + a \cdot \bar{b} \cdot \overline{c_{in}})$$

$$= [(\bar{a} \cdot \bar{b} + \bar{a} \cdot \overline{c_{in}} + \bar{b} \cdot \overline{c_{in}}) + (a \cdot b + a \cdot \overline{c_{in}} + b \cdot \overline{c_{in}})]c_{in} + (\bar{a} \cdot b \cdot \overline{c_{in}} + a \cdot \bar{b} \cdot \overline{c_{in}})$$

$$= (\bar{a} \cdot \bar{b} + \bar{a} \cdot \overline{c_{in}} + \bar{b} \cdot \overline{c_{in}})c_{in} + (a \cdot b + a \cdot \overline{c_{in}} + b \cdot \overline{c_{in}})c_{in} + (\bar{a} \cdot b \cdot \overline{c_{in}} + a \cdot \bar{b} \cdot \overline{c_{in}})$$

$$= (\bar{a} \cdot \bar{b} + \bar{a} \cdot \overline{c_{in}} + \bar{b} \cdot \overline{c_{in}})c_{in} + (a \cdot b + a \cdot \overline{c_{in}} + b \cdot \overline{c_{in}})c_{in} + (\bar{a} \cdot \overline{c_{in}} + \bar{b} \cdot \overline{c_{in}})(a \cdot \overline{c_{in}} + b \cdot \overline{c_{in}})$$

$$= (\bar{a} \cdot \bar{b} + \bar{a} \cdot \overline{c_{in}} + \bar{b} \cdot \overline{c_{in}})c_{in} + (a \cdot b + a \cdot \overline{c_{in}} + b \cdot \overline{c_{in}})c_{in} + (\bar{a} \cdot \bar{b} + \bar{a} \cdot \overline{c_{in}} + \bar{b} \cdot \overline{c_{in}})(a \cdot b + a \cdot \overline{c_{in}} + b \cdot \overline{c_{in}})$$

$$= m(\bar{a}, \bar{b}, c) \cdot c_{in} + m(a, b, \overline{c_{in}}) \cdot c_{in} + m(\bar{a}, \bar{b}, c) \cdot m(a, b, \overline{c_{in}})$$

$$= m(m(\bar{a}, \bar{b}, c), c_{in}, m(a, b, \overline{c_{in}}))$$

$$= m(\overline{c_{out}}, c_{in}, m(a, b, \overline{c_{in}}))$$

Q.E.D.

It is noteworthy that using a similar procedure, we can get two other equations to calculate s.

$$s = m(\overline{c_{out}}, a, m(\bar{a}, b, c_{in})) \qquad (4a)$$

$$s = m(\overline{c_{out}}, b, m(a, \bar{b}, c_{in})) \qquad (4b)$$

The equation (4a) is just a variant of (3c) by switching the signals a and c_{in}, while (4b) switches b with c_{in}. These two equations are the same as (3c) in terms of designing the QCA adder.

The advantage of the proposed algorithm is that it simplifies the QCA addition. Based on the proposed algorithm, the calculation of c_{out} involves one majority gate and the calculation of s involves two majority gates and two inversion operations $\overline{c_{out}}$ and $\overline{c_{in}}$. In contrast, in the original QCA addition algorithm [5], [6], the calculation of c_{out} involves one majority gate and the calculation of s is

$$s = m(m(\bar{a}, b, c_{in}), m(a, \bar{b}, c_{in}), m(a, b, \overline{c_{in}})) \qquad (5)$$

that requires four majority gates and three inversion operations $\bar{a}$, $\bar{b}$ and $\overline{c_{in}}$. Thus, the original one-bit adder requires five majority gates and three inverters, while based on the proposed algorithm, it can be implemented by using only three majority gates and two inverters.

B. Proposed QCA Adders

We now introduce a new design of one-bit QCA adder based on the proposed algorithm. The proposed one-bit QCA adder consists of three majority gates and two inverters as shown in

Fig. 6. It results in reduced hardware compared to the original full adder [5], [6] and retains the simple clocking scheme. It is noted that the bit-serial QCA adder [7] uses a variant of the proposed one-bit QCA adder.
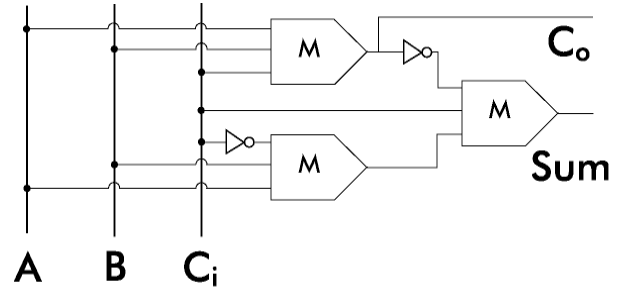

Figure 6. Proposed one-bit QCA full adder

To create an n-bit adder, we arrange n proposed one-bit adders vertically in a column. The clocking of the cells within the n-bit adder is designed such that the carry will propagate down to the last bit before the sum is calculated, thereby implementing a CLA adder. The proposed QCA adder design requires fewer majority gates and inverters while maintaining the same clocking scheme and speed in comparison with existing QCA adders. Compared to the bit-serial QCA adder, the proposed design is n times faster, at the expense of being approximately n times larger in area. Furthermore, the proposed design has a much simpler clocking scheme and does not require complicated feedback control.

The performance data of the proposed QCA adder and of those considered in [5], [6] and [7], are given in Table I. It is seen from Table I that the proposed design requires less hardware while retaining the simple clocking scheme and parallel structure of the original carry look-ahead approach.

TABLE I.   COMPARISON OF QCA ADDERS

QCA adders	Complexity of one-bit	Clocking	Speed
Proposed CLA	Three majority gates and two inverters	Simple	Fast with CLA
Previous CLA [5], [6]	Five majority gates and three inverters	Simple	Fast with CLA
Bit-serial [7]	Three majority gates, two inverters and feedback control	Complicated	Slow

IV.   SIMULATION RESULTS

The proposed QCA adders are designed and simulated by using the QCADesigner tool for the four-bit case. QCADesigner is a QCA layout and simulation tool developed at the University of Calgary. The design and simulation procedure is as follows. First, we generate the layout of the proposed one-bit adder. Then, we design a four-bit adder using the one-bit layout. Next, we setup the circuit clocking. Finally, we setup the vector table simulation to simulate the adder.

A. Four-Bit Adder Design and Simulation

The layout of the proposed one-bit adder is shown in Fig. 7. Using the layout of the one-bit adder, we design a four-bit QCA adder as shown in Fig. 8.
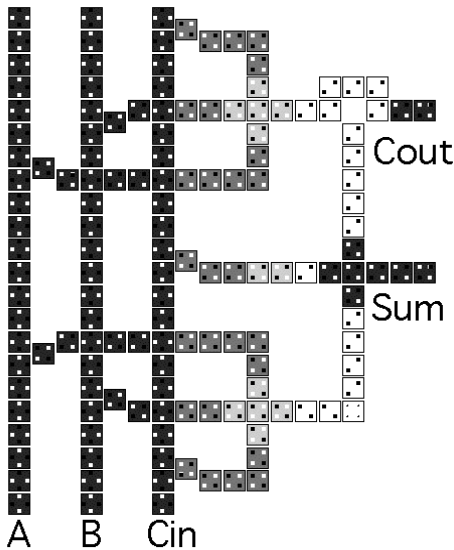

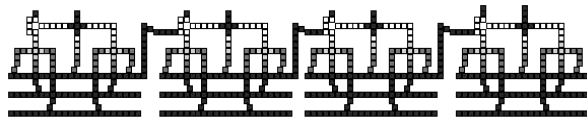Figure 7.   Layout of the proposed one-bit QCA adder

Figure 8.   Layout of the proposed four-bit QCA adder

B. Performance Comparison

To maintain consistency with size measurements in previous publications [5], [6], we assume that the QCA cells are made of 2nm quantum dots. The cells are separated by 10nm. Thus, the area of the proposed 4-bit adder is 0.220μm*0.959μm. The speed and clocking of the proposed design is comparable to the original design [5],[6].

Table II lists the area, speed and clocking of the proposed adder along with those of the existing one [5], [6]. It is seen from Table II that the proposed design requires only about 36% of the hardware compared to the existing one with the same speed and clocking performance. Thus, the simulation results are consistent with our theoretic results.

No performance comparison has been carried out with the bit-serial adder in [7], since no simulation data is available in [7].

TABLE II.   SIMULATION RESULTS OF QCA ADDERS

QCA adders (4-bit)	Area	Speed	Clocking
Proposed CLA	0.2109 μm²	Same	Same
Previous CLA [6]	0.598 μm²	Same	Same

V.   CONCLUSION

In this paper, a novel QCA adder design has been presented that reduces the number of QCA cells in comparison to previously reported designs. The proposed one-bit QCA adder structure is based on a new algorithm that requires only three majority gates and two inverters for QCA addition. By connecting *n* proposed one-bit QCA adders, we can obtain an *n*-bit carry look-ahead adder with the reduced hardware while retaining the simpler clocking scheme and parallel structure of the original approach. The proposed adder has been design and simulated using the QCADesigner tool for the four-bit adder case. The proposed design requires only 36% of the hardware compared to the existing one with the same speed and clocking performance.

REFERENCES

[1]   M.Wilson, K. Kannagara, G. Smith, M. Simmons, and B. Raguse, Nanotechnology: Basic science and emerging technologies, Champman & Hall/CRC, 2002.

[2]   C. S.Lent, P. Tougaw, and W. Porod, "bistable saturation in coupled quantum dots for quantum cellular automata," Appl. Phys. Lett., vol. 62, pp. 714, 1993.

[3]   C. S. Lent, P. Tougaw, W. Porod, and G. Bernstein, "Quatum cellular automata,," Nanotechnology, vol. 4, pp. 49-57, 1993.

[4]   J. Timer and C. S. Lent, "Power gain and dissipation in quantum-dot cellular automata," *J. Appl. Phys.*, American Institute of Physics, vol. 91, no. 2, pp. 823-831, 2002.

[5]   P.D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata," *J. Appl. Phys.*, American Institute of Physics, vol. 75, pp. 1818-1824, 1994.

[6]   A. Vetteth, K. Walus, V. S. Dimitrov, and G. A. Jullien, "Quatum-dot cellular automata carry-look-ahead adder and barrel shifter," *in Proc.* IEEE Emerging Telecommunications Technologies Conference, Dallas, TX, Sept. 2002.

[7]   Fijany, N. Toomarian, K. Modarress, and M. Spotnitz, "Bit-serial adder based on quantum dots," NASA technical report, Jan. 2003.