Implementation of 4x4 Vedic Multiplier using Carry Save Adder in Quantum-Dot Cellular Automata

Ashvin Chudasama, Trailokya Nath Sasamal

***Abstract*—A multiplier is the basic structural unit of many arithmetic logical units(ALU), digital signal processing(DSP) and communication system. So the area, speed and power consumption are the prime factors for the designing of multiplier circuits. QCA (Quantum dot-Cellular Automata) is one of the alternatives, which yields small size and low power consumption. In this paper, we proposed a 4-bit Vedic multiplier (using Urdhva Tiryagbhyam sutra) in QCA Technology. The generated partial product addition in Vedic multiplier is realized using carry-save technique. The design is simulated on QCA Designer tool and it confirms the efficiency of the design. The simulation result shows that, the proposed design has reduced 30% cell count, 60% reduction in area and 50% delay as compared to the 4x4 Wallace and Dadda multiplier.**

***Index Terms*—Carry Save Adder, Full Adder (FA), Majority gate, Clocking, Quantum dot Cellular Automata (QCA), 4x4 Vedic Multiplier.**

I. INTRODUCTION

SINCE moore said that the number of transistors on a single IC doubles approximately every 18 months. So reduced size, increase in operating speed and low power are successfully achieved by VLSI technology. But now a days the devices are exponentially scaled down due to this CMOS technology facing the problems like Gate leakage current, Interconnection noise and Stray capacitances, which degrades the circuit performance [1]. The International Technology Roadmap for Semiconductors (ITRS) found many alternatives to overcome these problems. Quantum dot-Cellular Automata (QCA) which can work on nano-scale is one of the solutions. This new technology gives new techniques for the data transmission and data computation [2].

Theoretically the concept of cellular automata was introduced by Von Neumann and Ulam in 1940 [3]. Later in 1993 Lent et al. first experimentally demonstrated the possibility of Quantum dot-Cellular Automata, with Aluminum Island acting as Quantum dots [4]. In QCA the cell is used for logic computation and also used for interconnection i.e., for data transmission [5]. The Majority Gate Voter and Inverter are the basic logic gates in the QCA architecture designing.

Recently, highly resourceful design of arithmetic operation in QCA has been followed, it includes the design of Adders [6] and multipliers [7]. The speed of adder mainly depends on the carry propagation delay, if we reduce this delay, it will enhance the multiplier efficiency. This paper mainly concentrates on a Vedic multiplier using Carry Save Adder, which utilized the efficient full adder and half adder designs in QCA Designer. The design has been simulated using QCA Designer [8] and the simulation result proves that the multiplier has high speed and low area compared to previous multipliers [5, 9].

The QCA terminology is explained in the section-II and the design of 4-bit Vedic mathematics has been presented in section III. Proposed 4x4 vedic multiplier in section IV Simulation results and comparisons are presented in section V. Conclusions are presented in section VI.

II. TERMINOLOGY

The basic component of QCA logic gates and circuits is the QCA Cell. A QCA cell is square like structure consisting of four Quantum dots, which are placed in four corners of the cell. The Quantum dot is a conducting material in nanometer sizes, which are surrounded by an insulating material. So this structure has the ability to trap the electron in a three dimensional space and if an electron comes into a quantum dot, it cannot escape without enough electrical potential. All this four quantum-dots are connected by four tunneling

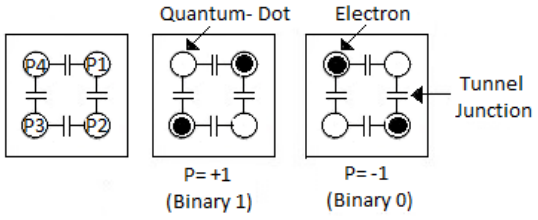


Fig. 1. Basic QCA cell with two possible polarizations.

Ashvin Chudasama is with the School of VLSI Design and Embedded Systems Department, National Institute of Technology, Kurukshetra, Hayana, India. (e-mail: chudasamaashvin093@gmail.com).

Trailokya Nath Sasamal is with Electronics and Communications Engineering Department, National Institute of Technology, Kurukshetra, Haryana, India. (e-mail: tnsasamal.ece@nitkkr.ac.in).

978-1-5090-0396-9/16/$31.00 ©2016 IEEE

junctions. In QCA cell, two extra electrons are intruded, which have the ability to tunnel between quantum-dots [3-10].

These intruded extra electrons have Coulombic interaction together, due to the Coulombic repulsion they have only two possible diagonal arrangements according to their polarities as shown in Fig. 1. These two arrangements have negative or positive polarities, which represents logic "0" or logic "1" binary information in the cell. If the polarization of all four quantum-dots are as p1, p2, p3, and p4, then the polarization of the QCA Cell can be calculated By using eq.1.

$$P = \frac{[(P2 + P4) - (P1 + P3)]}{P1 + P2 + P3 + P4} \quad (1)$$

A. Basic QCA Gates

You In quantum cellular automata, the primary Boolean logic function can be realized by the physical interaction between cells and additive nature of the coulombic force [11]. The Basic QCA logic device includes QCA wire, QCA Majority Gate and QCA Inverter as explained below.

1) QCA Wire: The basic practical cell arrangement is obtained by placing QCA cells in series, to the side of each other. Then the next cell occupies the polarization from the neighbor cell polarity because of physical interaction between the cells and its polarity depends on the position of the cell. Fig. 2 (a) shows 90 degrees wire, but when dots in the wire are rotated by 45 degrees, then this arrangement is called the 45-degree wire as shown in Fig. 2 (b) [10].

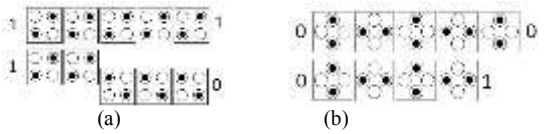


Fig. 2. (a) 90 degree (b) 45 degree wire according to their location of cell

2) QCA Majority Gate: Fig. 3 shows a QCA Majority Gate build by the 5 QCA cells in which 3 are input cells, one intermediate cell and one output cell. A 2 input AND gate can be implemented by the 3 input majority gate by fixing one input to logic zero out of the 3 inputs. Similarly, if the input is fixed to a logic one, then it functions like a 2 input OR gate [12]. The mathematical equation for the majority gate is as follows.

$$M(A, B, C) = AB + BC + AC \quad (2)$$

3) QCA Inverter: The QCA inverter gate cannot be constructed by the majority gate. The QCA inverter gate has one input and it simply returns the opposite of input logic as output. A standard implementation of the NOT gate is shown in Fig. 4.

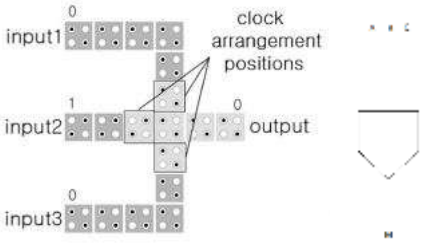


Fig. 3. Majority gate with its symbol and clock arrangement [12].

Fig. 4. QCA inverter [9].

B. Wire Crossing

The QCA technology uses two types of wire crossing, 1) coplanar wire crossing and 2) multilayer wire crossing. Here we discuss, only about Multilayer crossings. Multilayer layer crossing use three different layers of 90° wire to form a bridge as shown in Fig.5 [7]. The lower most layer is generally considered as the main cell layer, second layer is for vertical interconnect known as Middle layer (layer1) and the top most layer is known as upper layer (layer2). In comparison with coplanar wire crossings, the design of the multilayer wire crossings has less complexity and wire connection is more reliable [13]. In this paper, we used Multilayer layer crossing for designing the multiplier.

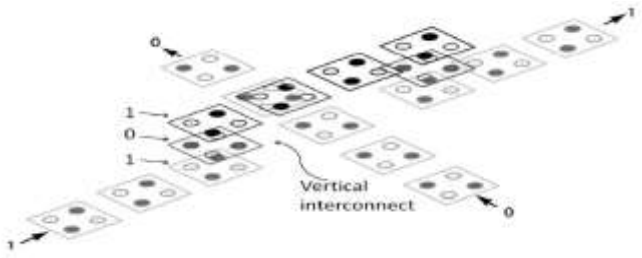


Fig. 5. Multilayer Wire Crossing [13].

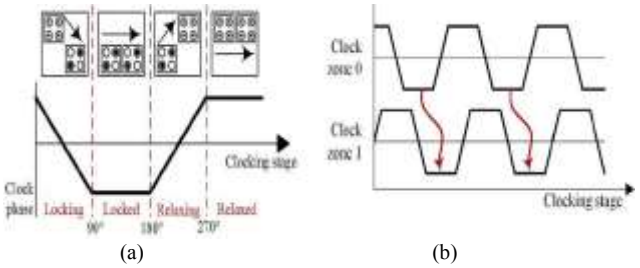
C. Signal and Control

QCA clocking is the tricky challenge for the QCA circuit implementation. They avoid the random adjustments of QCA cells due to the noise and guide the data propagation though QCA circuits.

In QCA cell all four quantum-dot are connected by the four tunneling junctions. The raising or lowering phenomena of tunneling barriers are used in the clocking technique. Typically the tunneling barrier is modulated by the four distinct stages as shown in Fig. 6 (a). First, the locking stage is the raising of the tunneling barriers. Next, in locked stage tunneling barriers are fully raised and the electrons are localized in quantum-dot according to the neighbour cell polarization. Then in the relaxing stage, the tunneling barriers are lowering and in the relaxed stage tunneling barriers are

completely suppressed, allowing the electrons to move freely in the cell [10].

Data transmission, through two clock zone is performed as shown in Fig. 6 (b). When clock zone 0 is in locked stage and clock zone 1 is in the locking stage at the same time, then there exists, the data propagation from clock zone 0 to clock zone 1 respectively.



(a) (b)

Fig. 6. (a) Four phase of operation of QCA cells during a clock period. (b) QCA clocking scheme [11].

III. VEDIC MATHEMATICS

Vedic mathematics has 16 different sutras to perform multiplication. Here we discussed only Urdhwa and Tiryakbhyam sutras, which increase the computation speed by performing concurrent addition and generating partial product. "Urdhwa and Tiryakbhyam " is a Sanskrit word, which means vertically and crosswise respectively. This technique is generally used for the decimal multiplication. The main advantage of this method is that it generates partial products in parallel and adds them concurrently [14]. Thus, this method takes the benefit of the parallelism in generation of partial product which increase the speed of multiplication without increasing the clock frequency. This technique can be used for 2x2, 4x4, 8x8,…..NxN bit multiplication. Here we exploit the advantages of the above technique to build a 4x4 multiplier.

A. 2×2 Vedic Multiplier Block

The algorithm can be easily explained by using a two 2-bit binary numbers A (A1A0) and B (B1B0) as shown in the line diagram (Fig.7). As a start LSB of B (B0) is multiplied with LSB of A (A0), the generated partial product is considered as an LSB of final product (vertically). The next step is that the LSB of B (B0) is multiplied with MSB of A (A1) and MSB of B (B1) is multiplied with LSB of A (A0) (crosswise). To add the generated partial products (B0xB1+ A0xB1) a half adder is required, which generates a two bit result (C1 S1) of which the LSB (S1) is considered as the second bit of the final product and MSB (C1) is saved as pre carry for the next step. Finally MSB of B (B1) is multiplied with MSB of A (A1), the generated partial product (A1 x B1) and saved pre carry are added using another half adder. The resultant is again a two bit (C2 S2), considered as third and fourth bit of the final product (C2 S2 S1 S0). The diagram of 2x2 Vedic multiplier is shown in Fig. 8 [15].

$S_0 = A_0B_0$;

$C_1\ S_1 = A_1B_0 + B_1A_0$;

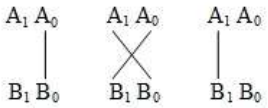


Fig. 7. 2x2 bit multiplication by Vedic mathematics.

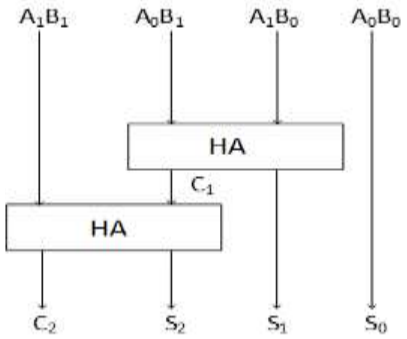


Fig. 8. Block Diagram of 2x2 bit Vedic Multiplier.

$S_2 = C_1 + A_1B_1$;

IV. PROPOSED 4 X 4 VEDIC MULTIPLIER

The proposed 4 x 4 Vedic multiplier operation is explained using the line diagram as shown in the Fig. 4. Consider two 4-bit binary numbers A (A3A2A1A0) and B (B3B2B1B0) respectively. As a first step the LSB (A0 x B0) of both numbers are multiplied to generate the LSB of the final product. Further the Same procedure of 2x2 multiplier is carried out to generate the remaining bits of the final product. The carry save adder technique is used to add the partial products to reduce the computation time. The carry save adder is used as shown in Fig.6. Thus, this algorithm mainly depends on the design of an efficient half-adder and full-adder which directly contributes to the efficiency of the Vedic multiplier in QCA. A half- adder and full-adder design in QCA can be achieve by using three majority gates and one inverter [13].

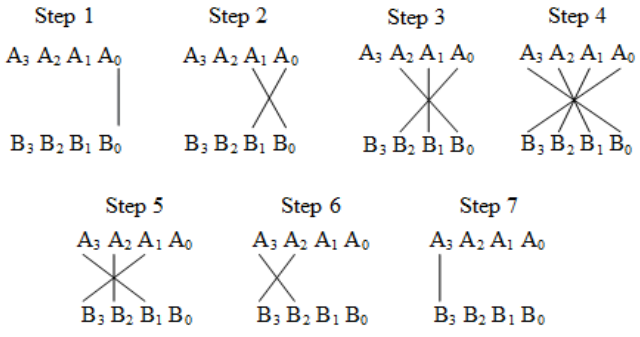


Fig. 9. Line Diagram of 4 x 4 Vedic Multiplier.

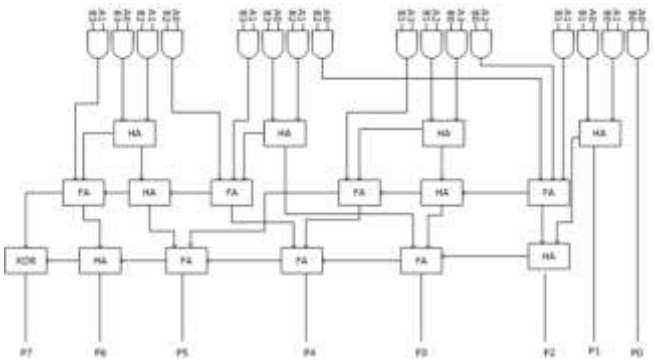


Fig. 10. Schematic Block diagram of 4 x 4 Vedic Multiplier.

A. Half Adder Realization

Let A and B are two binary inputs. The sum and carry can be calculated by using equation 3 and 4 respectively.

$$Carry = AB \tag{3}$$

$$Sum = \overline{Carry} + (A + B) \tag{4}$$

Proof :

$$\begin{aligned} Sum &= A \oplus B \\ &= \overline{(A\overline{B} + \overline{A}B)} \\ &= (\overline{AB}) + (A + B) \\ &= M(\overline{M(A,B,0)}, M(A,B,1),1) \end{aligned}$$

A one-bit Half adder circuit (with one inverter and three majority gates) as well as the QCA Designer layout are shown in Fig.11.

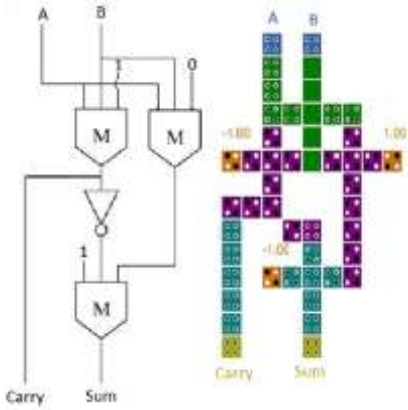


Fig. 11. Half Adder Circuit and Layout.

B. Full Adder Realization

Let A, B and C are three binary inputs. Full adder circuit realization is done By using the eq. (5) [13].

$$Sum = M(\overline{M(A,B,C)}, M(\overline{M(A,B,C)}, B, C), A) \tag{5}$$

So, by using the equation (5), full adder can be implemented by using three majority gate and one inverter only.

A 1-bit full adder (with one inverter and three majority gates) as well as the QCA Designer layout are shown in Fig. 12.

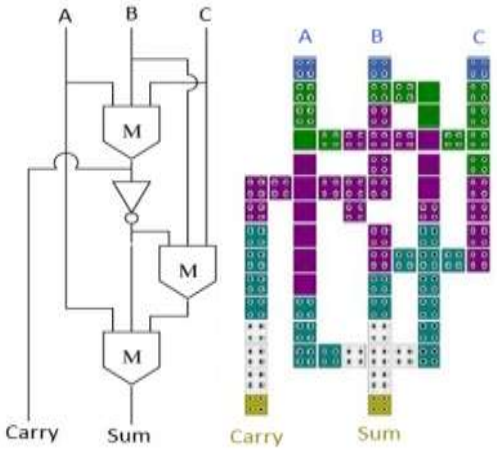


Fig. 12. Full Adder Circuit and Layout.

V. SIMULATION RESULT AND COMPARISIONS

The simulations were carried out using QCA Designer tool [8]. In this proposed design, multilayer wire crossing and a maximum number of 15 cells per clock zone are used. Also the default value of cell dimension is chosen (18nm x 18nm with 5 nm diameter and center to center distance is 20nm) and the Coherence vector engine (using Euler method and Randomize simulation order) is used for the simulation. The layout of the 4-bit multiplier based on Vedic mathematics is shown in Fig. 14 and the output result of the simulation is shown in Fig. 13. The simulation results are compared with the previous multiplier with respect to cell count, delay, throughput and area are summarized in Table I. The proposed 4x4 Vedic Multiplier has 4.25 clock cycle delay, 1955 cell count and Area-Delay product is 8.96, which are less in comparison with the previous multipliers [4, 9].

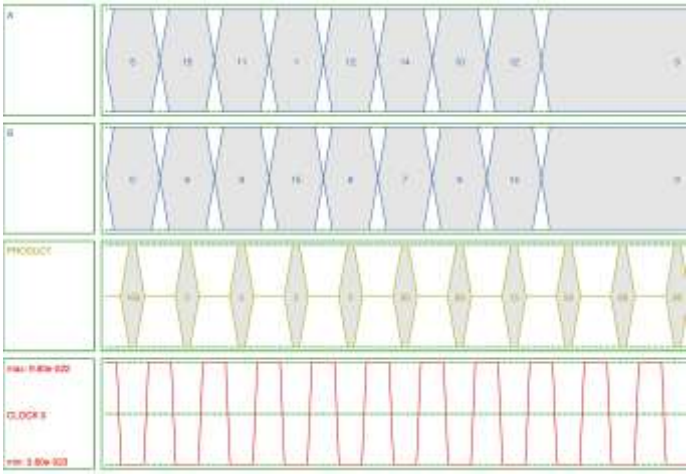


Fig. 13. Resultant Waveform of 4-bit Vedic Multiplier.

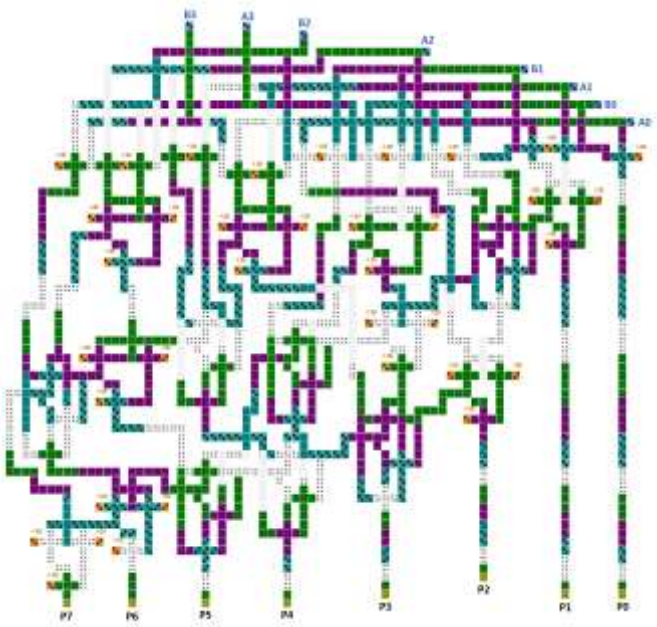


Fig. 14. Layout of 4x4 Vedic Multiplier.

TABLE I
COMPARISON OF DIFFERENT QCA MULTIPLIER DESIGNS

Approch	Cell Count	Area (μm^2)	Dealy (Clock Cycle)	Throughput	Area-Delay Product
4x4 Wallace	3295	7.39	10	1	73.9
4x4 Dadda	3384	7.51	12	1	90.12
4x4Serial -Parallel [4]	406	0.493	1	1/8	0.4935
N x N Array multiplier	-	$0.44N^2$	4N-1	1	-
Proposed 4x4 Vedic Multiplier	1955	2.25	4.25	1	8.96

VI. CONCLUSION

In this paper, proposed 4x4 Vedic multiplier architecture is designed by using carry save adder. This design is simulated on QCA Designer. The simulation shows 30% reduction in cell count, 60% reduction in area and 50% reduction in delay as compared to the 4x4 Wallace and Dadda multiplier. Thus, the reduced number of cell count, directly contributes to low power application.

REFERENCES

[1] Rairigh, Daniel. "Limits of CMOS technology scaling and technologies beyond-CMOS." Institute of Electrical and Electronics Engineers, Inc , 2005.

[2] Kummamuru, Ravi K., et al. "Operation of a quantum-dot cellular automata (QCA) shift register and analysis of errors." *Electron Devices, IEEE Transactions on* 50.9, pp. 1906-1913, 2003.

[3] Von Neumann, John, and Arthur W. Burks. "Theory of self-reproducing automata." *IEEE Transactions on Neural Networks* 5.1, pp. 3-14, 1966.

[4] Lent, Craig S., et al. "Quantum cellular automata." Nanotechnology 4.1, pp. 49, 1993.

[5] Porod, Wolfgang. "Quantum-dot devices and quantum-dot cellular automata."International Journal of Bifurcation and Chaos 7.10, pp. 2199-2218, 1997.

[6] Cho, Heumpil, and Earl E. Swartzlander. "Adder designs and analyses for quantum-dot cellular automata." *Nanotechnology, IEEE Transactions on* 6.3, pp. 374-383, 2007.

[7] Cho, Heumpil, and Earl E. Swartzlander Jr. "Adder and multiplier design in quantum-dot cellular automata." *Computers, IEEE Transactions on* 58.6, pp. 721-727, 2009.

[8] Walus, Konrad, et al. "QCADesigner: A rapid design and simulation tool for quantum-dot cellular automata." *Nanotechnology, IEEE Transactions on* 3.1, pp. 26-31, 2004.

[9] Kim, Seong-Wan, and Earl E. Swartzlander Jr. "Parallel multipliers for quantum-dot cellular automata." *Nanotechnology Materials and Devices Conference, 2009. NMDC'09. IEEE*. IEEE, 2009.

[10] Kim, Kyosun, Kaijie Wu, and Ramesh Karri. "The robust QCA adder designs using composable QCA building blocks." *IEEE transactions on computer-aided design of integrated circuits and systems* 1.26, pp. 176-183, 2007.

[11] Tougaw, P. Douglas, and Craig S. Lent. "Logical devices implemented using quantum cellular automata." Journal of Applied physics 75.3, pp. 1818-1825, 1994.

[12] Tougaw, P. Douglas, and Craig S. Lent. "Dynamic behavior of quantum cellular automata." Journal of Applied Physics 80.8, pp. 4722-4736, 1996.

[13] Pudi, Vikramkumar, and K. Sridharan. "Efficient design of Baugh-Wooley multiplier in quantum-dot cellular automata." *Nanotechnology (IEEE-NANO), 2013 13th IEEE Conference on*. IEEE, 2013.

[14] Thapliyal, Himanshu, and Hamid R. Arabnia. "A Time-Area-Power Efficient Multiplier and Square Architecture Based on Ancient Indian Vedic Mathematics." *ESA/VLSI*. 2004.

[15] Tiwari, Honey Durga, et al. "Multiplier design based on ancient Indian Vedic Mathematics." *SoC Design Conference, 2008. ISOCC'08. International*. Vol. 2. IEEE, 2008.