

Scientific Journal of Impact Factor(SJIF): 3.134

e-ISSN(O): 2348-4470 p-ISSN(P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 2, Issue 9, September-2015

Design of Efficient Binary Comparatorsin Quantum-Dot Cellular Automata

M.HARI CHANDRA REDDY⁽¹⁾, L.S.DEVARAJ⁽²⁾

(1)PG Scholar, VLSI System Design, Intellectual Institute of Technology, AP, India (2)Assistant Professor, Intellectual Institute of Technology, AP, India

Abstract *Quantum-dot cellular automata (QCA) are an attractive emerging technology suitable for the development of ultradense low-power high-performance digital circuits. Efficient solutions have recently been proposed for several arithmetic circuits, such as adders, multipliers, and comparators. Nevertheless, since the design of digital circuits in QCA still poses several challenges, novel implementation strategies and methodologies are highly desirable. This paper proposes a new design approach oriented to the implementation of binary comparators in QCA. New formulations of basic logic equations required to perform the comparison function are proposed. The new strategy has been exploited in the design of two different comparator architectures and for several operands word lengths. With respect to existing counterparts, the comparators proposed here exhibit significantly higher speed and reduced overall area.*

Index Terms—Binary comparators, majority gates, quantum- dot cellular automata (QCA).

I.INTRODUCTION

Quantum Cellular Automata (QCA) refers to models of quantum computation, which have been devised in analogy to usual models of cellular automata introduced by von Neumann. It may also refer to quantum dot cellular automata, which is a planned physical implementation of "classical" cellular automata by exploiting quantum mechanical phenomena. QCA has attracted a lot of interest as a result of its really small feature size and its ultra-low power consumption, building it one candidate for replacing CMOS technology. Complementary metal-oxide semiconductor (CMOS) technology has been the industry typical for implementing Very Large Scale Integrated (VLSI) devices for the last 2 decades, mostly due to the penalty of miniaturization of such devices QCA is only one of the many alternative technologies proposed as a substitute solution to the basic limits CMOS technology will impress in the years to come.

Even though QCA solves most of the restrictions of CMOS technology, it also brings its own. The intrinsic switching time of a QCA cell is at best in the order of terahertz. Though, the real speed may be much lesser, in the order of megahertz for solid state QCA and gigahertz for molecular QCA, due to the good quasi-adiabatic clock switching frequency setting. Moreover, solid-state QCA devices cannot work at room temperature. The only choice to this temperature limitation is the newly proposed "Molecular QCA" which theoretically has an inter-dot distance of 2 nm and an inter-cell distance of 6 nm.

II. BACKGROUND AND RELATED WORKS

The basic element of a nanostructure based on QCA is a square cell with four quantum dots and two free electrons. The latter can tunnel through the dots within the cell, but, owing to Columbic repulsion, they will always reside in opposite corners thus leading to only two possible stable states, also named polarizations. Locations of the electrons in the cell are associated with the binary states 1 and 0.

Adjacent cells interact through electrostatic forces and tend to align their polarizations. However, QCA cells do not have intrinsic data flow directionality. Therefore, to achieve controllable data directions, the cells within a QCA design are partitioned into the so-called clock zones that are progressively associated with four clock signals, each phase shifted by 90°. Thisclock scheme, named the zone clocking scheme, makes the QCAdesigns intrinsically pipelined, since each clock zone behaves like a D-latch

QCA cells are used for both logic structures and interconnections that can exploit either the coplanar cross or the bridge technique the fundamental logic gates inherently available within the QCA technology are the inverter and the majority gate (MG). Given three inputs a, b and c, the MG performs the logic function reported in (1) provided that all input cells are associated with the same clock signal clkx (with x ranging from 0 to 3), whereas the remaining cells of the MG are associated with the clock signal clkx +1

 $M(a, b, c) = a \cdot b + a \cdot c + b \cdot c \tag{1}$

There are several QCA designs of comparators in the A 1-bit binary comparator receives two bits a and bas inputs and establishes whether they are equal, less than or greater than each other. These possible states are represented through three output signals, here named Aeq B, A big B, Bbig A, that are asserted, respectively, when a = b, a > b, and a < b. Full comparators are those that can separately identify all the above cases, whereas non-full comparators recognize just one or two @IJAERD-2015, All rights Reserved 45

of them. As an example, the comparator designed in and depicted in Fig. 1(a) can verify only whether a = b. Conversely, the circuits shown in Fig. 1(b) and (c), proposed in and are full comparators. The latter also exploits two 1-bit registers D to process n-bit operands serially from the least significant bit to the most significant one.

With the main objective of reducing the number of wire crossings, which is still a big challenge of QCA designs in the universal logic gate (ULG) f (y1, y2, y3) = M (M (y1, y2, 0), M (y1, y3, 1), 1) was proposed and then used to implement the comparator illustrated in Fig. 1(d). It is worth noting that, two n-bit numbers A (n –1:0) =an –1 ... a0 and B (n –1:0) = bn -1 ... b0 can be processed by cascading n in- stances of the 1-bit comparator. Each instance receives as inputs the ith bits ai and bi (with i = n – 1... 0) of the operands and the signals A bigB (i–1:0) and Bbig A (i–1:0). The former is asserted when the sub word A (i–1:0) = ai-1 ... a0 represents a binary number greater than B (i–1:0) = bi-1 ... b0. In a similar way, Bbig A (i–1:0) is set to 1 when A (i–1:0) <B (i–1:0). The outputs A bigB (i: 0) and Bbig A (i: 0) directly feed the next stage. It can be seen that this circuit does not identify the case in which A = B, therefore it cannot be classified as a full-comparator.

The design described in exploits a tree-based (TB) architecture and exhibits a delay that in theory logarithmically increases with n. The 2-bit version of such designed comparator is illustrated in Fig. 1(e). Also the full comparator proposed in exploits a TB architecture to achieve high speed. As shown in Fig. 1(f), where4-bit operands are assumed, one instance of the 1-bit comparator presented in is used for each bit position. The intermediate results obtained in this way are then further processed through a proper number of cascaded 2-input OR and AND gates implemented by means of MGs having one input permanently set to1 and 0, respectively



Fig.1. QCA-based comparators

III. DESIGNING BINARY COMPARATORS EXPLOITING THE NEW THEOREMS

The circuits illustrated in Fig. 2 were designed to implement in QCA the novel equations demonstrated in the previous Section. The generic module Ti, with i ranging between 1 and 4, implements the equations enunciated in the ith theorem,

@IJAERD-2015, All rights Reserved

whereas C1 and C2 compute the signals AbigB(k-1:0), BbigA(k-1:0), and AeqB(k-1:0) as shown above in Corollaries 1 and 2, respectively. As examples of application, the above QCA modules havebeen used to design two different structures of full comparators here named cascade-based and TB architectures. However, many other structures can be designed by combining the basic modules in different manners.

A. Novel QCA Comparators

The first proposed comparator exploits a cascade-based (CB) architecture. To explain better how the overall computation is performed, the schematic diagram illustrated in Fig. 3 is provided. It shows a possible implementation of a 32-bit comparator based on the proposed theory. Following the criterion illustrated



Fig. 2. QCA modules: (a) T1; (b) T2; (c) T3; (d) T4; (e) C1; and (f) C2.



Fig. 3. Novel 32-bit CB full comparator.

in Fig. 3, an n-bit CB full comparator designed as proposed here uses: n/3 instances of T1 and/or T2; n/3 cascaded instances of T4 through which the signals A big B(n -1:0) and Bbig A(n -1:0) are computed; and one instance of C2, needed to compute also Aeq B(n -1:0). Circles visible in Fig. 3 indicate the additional clock phases that have to be inserted on wires to guarantee the correct synchronization of the overall design.

The CB full comparator was designed for operands word lengths ranging from 2 to 32 and using, for n > 2, the split criterion summarized in Table I. Obviously, alternative splits could be used.

TABLEISPLITTING CRITERIONADOPTEDINTHECBCOMPARATORS

п	Splitting of the operands
4	$A_{(3:2)}A_{(1:0)}$ $B_{(3:2)}B_{(1:0)}$
8	$A_{(7:5)}A_{(4:2)}A_{(1:0)} = B_{(7:5)}B_{(4:2)}B_{(1:0)}$
16	$A_{(15:14)}A_{(13:11)}A_{(10:8)}A_{(7:5)}A_{(4:2)}A_{(1:0)}$
	$B_{(15:14)}B_{(13:11)}B_{(10:8)}B_{(7:5)}B_{(4:2)}B_{(1:0)}$
32	A _(31:29) A _(28:26) A _(25:23) A _(22:20) A _(19:17) A _(16:14) A _(13:11) A _(10:8) A _(7:5) A _(4:2) A _(1:0)
	$B_{(31:29)}B_{(28:26)}B_{(25:23)}B_{(22:20)}B_{(19:17)}B_{(16:14)}B_{(13:11)}B_{(10:8)}B_{(7:5)}B_{(4:2)}B_{(1:0)}$

As it is well known, the number of cascaded MGs within the worst computational path of a QCA design directly affects the delay achieved. In fact, each MG introduces one clock phase in the overall delay. From Fig. 2, it can be seen that the modules T1 and T2 contribute to the computational path with one inverter and two MGs. Each instance of T4 introduces one more MG, whereas C2 is responsible for one MG and one inverter. As a consequence, the critical computational path of the novel n-bit CB full comparator consists of n/3+ 3 MGs and 2 inverters. As an example, the 32-bit implementation depicted in Fig. 3 has the worst-case path made up of 13 MGs and 2 inverters.

As always happens in CB computational architectures, the number of MGs within the computational path of the abovedescribed comparator linearly increases with n. An alternative solution presented here adopts a TB architecture to achieve shorter computational paths. When this approach is exploited, several implementations of an n-bit full comparator can be designed differently combining the novel theorems and corollaries, as well as their QCA implementations depicted in Fig. 2. The TB comparators implement the comparison function recursively. The operands A and B are preliminarily partitioned as A = AMS B ALS B and B = BMS B BLS B. The portions AMS B and BMS B are compared independently of the portions ALS B and BLS B. The depth of the recursion directly impacts the whole architecture. Examples of TB structures designed for 16- and32-bit comparators are illustrated in Fig. 4. In Fig. 4(b) and (d), the recursion with its minimum depth is adopted. The portions AMS B and BMS B, as well as the portions ALS B and BLS B, are separately compared trough two independent CB architectures. The overall result is finally built with the modules C1 and C2. Fig. 4(a) and (c) shows comparators designed adopting deeper recursions.

In the following of the paper, the 16- and 32-bit TB implementation's illustrated in Fig. 4(b) and (d) are deeply analysed. Referring to the QCA modules depicted in Fig. 2, it can be easily verified that the former uses 35 MGs and 17 inverters and its critical computational path consists of 7MGs and 2 inverters, whereas the latter utilizes 83 MGs and 33 inverters and it has a worst-case path composed by 9 MGs and 2 inverters.

B. Results

Preliminary results obtained for the novel comparators at several operands word lengths are reported in Table II and compared to the pre-implementation characteristics furnished in the original papers for the parallel comparators described in [26] and [27] for operands wider than 1 bit. The design complexity of the comparators examined is reported in terms of number of MGs and inverters required in the overall designs, and number of MGs within the worst computational paths. The computational capability of each architecture is also specified.



Fig. 4. Examples of novel TB comparators with: (a) and (b) 16-bit operands; (c) and (d) 32-bit inputs.

		-				1,000.000 ns	
Name	Value	0 ns	200 ns	400 ns	600 ns	800 ns	
16 x	1						
16 у	0						
16 z	0						
▶ 📷 a[31:0]	00000100000	(1000000X		00000 1000000000 100000	10000000001		
🕨 🏹 b[31:0]	00100000000	(10000000000000000000000000000000000000	×	0010000000000000000	0010000000000010		
		X1: 1,000.000 ns					

Fig5:Simulation Results of TB

						1,000.000 ns
Name	Value	0 ns	200 ns	400 ns	600 ns	800 ns
l₀ x	1					
Ц у	0					
🗓 z	0					
🕨 📷 a[31:0]	00000100000	1000000		00000 1000000000 100000	10000000001	
▶ ■ b[31:0]	00100000000	100000000000000000000000000000000000000		00100000000000000000	0010000000000010	
		X1: 1,000.000 ns				

Fig6: Simulation Result of CB

IV. CONCLUSION

A new methodology useful to design binary comparators in QCA has been presented. It is based on innovative formulations that allow increased speed performances and reduced overall sizes to be achieved with respect to the existing competitors. The novel comparators split the received n-bit inputs into a proper number of 2- and 3-bit sub words that are processed in parallel through 2- and 3-bit comparators designed by applying theorems demonstrated here Thanks to the basic logic and layout strategies adopted, a 32-bit CB full comparator designed as described in this paper exhibits a delay of only 3+(3/4) clock cycles, occupies an active area of 2.66 μ m2, and achieves an area-delay product less than 10. When the alternative TB architecture presented here is exploited, the delay is further reduced to 2+(3/4) clock cycles; the active area is ~2.9 μ m2, whereas the area-delay product is less than 8.

V. FUTURE SCOPE

Currently, studies are being performed to extend this design in a form of tree based architecture to bring more features of comparator. We can try to analyse the area and delay of existing system and proposed system. The accomplishment of two key issues like area and complexity, provide an excellent starting point for future QCA designs. The QCA circuits can function logically and can be used to implement equivalent versions of CMOS circuits. The resulting designs can be used to build an Arithmetic and Logical Unit (ALU) which can be enhanced to design a Nano processor.

REFERENCES

[1] C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernstein, "Quantum cellular automata," Nanotechnology, vol. 4, no. 1, pp. 49–57, 1993.

[2] M. T. Niemer and P. M. Kogge, "Problems in designing with QCAs: Lay- out = timing," Int. J. Circuit Theory Appl., vol. 29, pp. 49–62, 2001.

[3] G. H. Bernstein, A. Imre, V. Metlushko, A. Orlov, L. Zhou, L. Ji, G. Csaba, and W. Porod, "Magnetic QCA systems," Micro electron. J., vol. 36, pp. 619–624, 2005.

[4] J. Huang and F. Lombardi, Design and Test of Digital Circuits by Quantum- Dot Cellular Automata. Norwood, MA, USA: Artech House, 2007.

[5] W. Liu, L. Lu, M. O'Neill, and E. E. Swartzlander Jr., "Design rules for quantum-dot cellular automata," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), Rio De Janeiro, Brazil, May 2011, pp. 2361–2364.

[6] K. Kim, K. Wu, and R. Karri, "Towards designing robust QCA architectures in the presence of sneak noise paths," in Proc. IEEE Design, Automation Test Eur. Conf. Exhib. (DATE), Munich, Germany, Mar. 2005, pp. 1214–1219.

@IJAERD-2015, All rights Reserved

[7] K. Navi, M. H. Moaiyeri, R. F. Mirzaee, O. Hashem pour, and

B. M. Nezhad, "Two new low-power full adders based on majority-not gates," Micro electron. J., vol. 40, pp. 126–130, 2009.
[8] H. Cho and E. E. Swartzlander Jr., "Adder design and analyses for quantum-dot cellular automata," IEEE Trans. Nanotechnology., vol. 6, no. 3, pp. 374–383, May 2007

ACKNOWLEDGMENT

FIRST AUTHOR: **Hari Chandra Reddy** received the B.Tech degree in Electronics and Communication Engineering in the year 2006 and pursuing M.Tech degree in VLSI System design from Intellectual institute of technology. Area of interests includes Communication and VLSI Design

SECOND AUTHOR: L.S.Devaraj received the B.Tech degree in Electronics and Communication Engineering and M.Tech degree in VLSI. Currently, She is working as Assistant Professor in the Department of Electronics and Communication Engineering, Intellectual institute of technology, Ananthapuramu.Having three years of teaching experience. Area of interest includes VLSI design, vhdl and Verilog.