International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

Area-Delay Dynamic Binary Adders in QCA

[1] V R K Chaitanya Uppuluri, [2] Ch.Lavanya , [3]S.Neelima

[1]M.Tech (VLSI & Embedded System),
[2]Asst.Professor,
[3]HOD, Assoc.Proffesor,
Gandhiji Institute of Science & Technology (Gist) – JNTU Kakinada Affiliated,
Gattubhimavaram (Village),Jaggayyapet (Mandal),Krishna (District),Andhra Pradesh, India.

Abstract-

In this paper, a novel quantum-dot cellular automata (QCA) adder design is presented that decrease the number of QCA cells compared to previously report designs. A novel 128-bit adder designed in QCA was implemented. It achieved speed performances higher than all the existing. QCA adders, with an area requirement comparable with the cheap RCA and CFA established. The area necessity of the QCA adders is comparable cheap with the RCA and CFA established. The novel adder operated in the RCA fashion, but it could propagate a carry signal through a number of cascade MGs significally lower than conventional RCA adders. In addition, because of the adopted basic logic and layout approach, the number of clock cycles required of completing the explanation was limited. As transistor decrease in size more and more of them can be accommodated in on its own die, thus increasing the chip computational capabilities. On the other hand, transistors cannot find much lesser than their existing size. The QCA approach represents one of the probable solutions in overcome this physical limit, even though the design of logic modules in QCA is not forever uncomplicated. The proposed one-bit QCA adder design is based on a new algorithm that requires only three majority gates and two inverters for the QCA addition. we propose a new adder that outperforms all state-of-the art competitors and achieves the best area-delay tradeoff.

Index Terms-- quantum-dot cellular automata (QCA); Adders; nanocomputing.

1. INTRODUCTION

A quantum-dot cellular automaton (QCA) is an attractive emerging technology suitable for the development of ultra dense low-power higher-performance digital circuits. For this cause in the last few years, the design of proficient logic circuits in QCA has received a great deal of attention. Special efforts are directed to arithmetic circuits, with the major interest focused on the binary addition that is the basic operation of any digital system. Of course, the designs commonly employed in traditional CMOS designs are considered a first reference for the new design environment. Ripple-carry[1], carry look-ahead (CLA), and conditional sum adders were implemented in. The carry-flow adder shown in was mainly an improved RCA in which detrimental wires effects were mitigated. Recently, further efficient designs were proposed for the CLA and the BKA, and for the CLA and the CFA. In this brief, an innovative technique is presented to implement high-speed low-area adders into QCA. Theoretical formulations established for CLA and parallel-prefix adders are here exploited for the realization of a novel 2-bit addition slice. The latter allows the carry to be propagated through two subsequent bit-positions with the delay of just one majority gate (MG). In addition, the clever top level architecture leads to very compact layouts, thus avoiding unnecessary clock phases due to lengthy interconnections. QCA is based on the interface of bi-stable QCA cells constructed from four quantum-dots. In a QCA wire, the binary signal propagates from

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

input to output because of the Columbic connections 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 high energy level, and would soon settle to the correct ground state.

2. QCA TECHNIQUE

Quantum – Dot Cellular Automata (sometimes referred as quantum cellular automata or QCA) are future models of quantum computation, which have been devised for analogy to conventional models of cellular automata introduced by Von Neumann. QCA[2] consists of four quantum dots in which two quantum dots are engaged by free electrons. Thus each cell consist two electrons. Electrons are arranged opposite to each other due to columbic repulsion [3]. The locations of the electrons establish the binary states. QCA is based on the interface of bi-stable QCA cells constructed from four quantum-dots. A high-level design 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 of charge to tunnel between adjacent dots. These electrons tend t take up antipodal sites as a result of their mutual electrostatic repulsion.

2.1 QCA CELL DIAGRAM

The following figure shows the simplified diagram of a QCA cell.

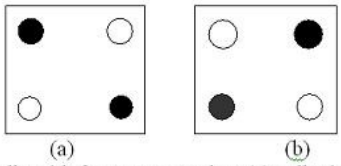

Fig.1. QCA cells with four quantum dots (a) cell with polarization p= -1(Logic '0') (b) cell with polarization p = 1 (Logic '1')

2.2 MAJORITY GATE AND INVERTER

The majority gate and inverter are shown in figure 2 and figure 3 respectively. The majority gate performs a three-input logic function. Assuming the inputs A, B and C, the logic function of the majority gate is m(A,B,C) = A|B +B|C+A|C By fixing the polarization of one input as logic "1" or "0", we can get an OR gate and an AND gate respectively. More complex logic circuits can be designed from OR and AND gates. Fig 1: Simplified diagram of a QCA cell.

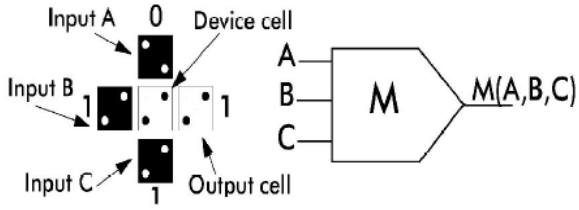

Fig 2: QCA Majority Gate

QCA is based on the interface of bi-stable QCA cells constructed from four quantum-dots. A high-level design 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 of charge to tunnel between adjacent dots. These electrons tend to take up antipodal sites as a result of their mutual electro static repulsion [3]. Thus, there exist two equal energetically minimal arrangements of the two electrons in the QCA cell as shown in Fig. 2. These two arrangements are denoted as cell polarization P = +1 and P = -1 correspondingly. By using cell polarization P = +1 to represent logic "1" and P = -1 to represent logic "0", binary information can be encoded. Arrays of QCA cells can be set to perform all logic functions. This is owed to the Columbic interactions, which influences the polarization of neighboring cells. QCA designs have been proposed with potential barriers between the dots that can be controlled and used to clock QCA designs.

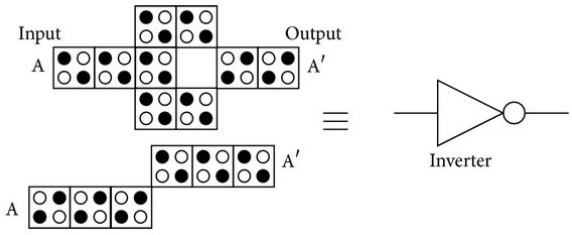

Fig. 3 A QCA inverter

3. ADDERS

In electronics, an adder or summer is a digital circuit that performs addition of numbers. In several computers and other forms of processors, adders are

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

used not solely within the arithmetic logic unit(s)[4], however additionally in alternative components of the processor, wherever they're accustomed calculate addresses, table indices, and similar operations. Though adders are made for several numerical representations, like binary-coded decimal or excess-3, the foremost common adders care for binary numbers. In cases wherever two's complement or ones' complement is getting used to represent negative numbers, it's trivial to switch associate adder into associate adder subtracted. Alternative signed range representations need a additional advanced adder.

3.1 Half adder

The half adder adds two single binary digits A and B. It has two outputs, sum (S) and carry (C). The carry signal represents an overflow into the next digit of a multi-digit addition. The value of the sum is 2C + S. The simplest half-adder design, pictured on the right, incorporates an XOR gate for S and an AND gate for C. With the addition of an OR gate to combine their carry outputs, two half adders can be combined to make a full adder

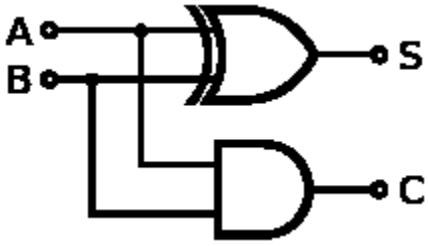

Fig4. Half adder logic diagram

3.2 Full adder

A full adder adds binary numbers and accounts for values conceded in as well as out. A one-bit full adder adds three one bit numbers, often written as A, B, and Cin,A and B are the operands, and Cin is a bit carried in from the next less significant stage. The full-adder is typically a component in a cascade of adders, which add 8, 16, 32, etc.

3.3 Ripple-carry adder

It is possible to create a logical circuit using multiple full adders to add N-bit numbers. Each full adder inputs a Cin, which is the Cout of the previous adder. This kind of adder is called a ripple-carry adder, since each carry bit "ripples" to the next full adder. Note that the first (and only the first) full adder may be replaced by a half adder (under the assumption that Cin = 0).

The layout of a ripple-carry adder is simple, which allows for fast design time.however, the ripple-carry adder is relatively slow, since each full adder must wait for the carry bit to be calculated from the previous full adder. The gate delay can easily be calculated by inspection of the full adder circuit. Each full adder requires three levels of logic. In a 32-bit ripple-carry adder, there are 32 full adders, so the critical path (worst case) delay is 2 (from input to carry in first adder) + 31 * 3 (for carry propagation in later adders) = 95 gate delays. The general equation for the worst-case delay for a n-bit carry-ripple adder is

$$T_{CRA}(n) = T_{HA} + (n-1)\cdot T_c + T_s = T_{FA} + (n-1)\cdot T_c = 6D + (n-1)\cdot 2D = (n+2)\cdot 2D$$

The delay from bit position 0 to the carry-out is a little different:

$$T_{CRA_{[0:c_{out}]}} = T_{HA} + n \cdot T_c = 3D + n \cdot 2D$$

The carry-in must travel through n carry-generator blocks to have an effect on the carry-out

$$T_{CRA_{[c0:cn]}}(n) = n \cdot T_c = n \cdot 2D$$

A design with alternating carry polarities and optimized AND-OR-Invert gates can be about twice as fast.[3]

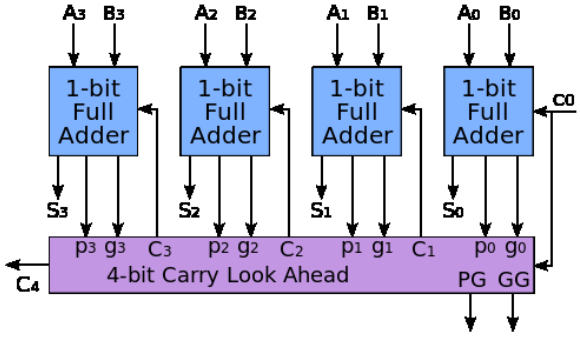

Fig 5. Ripple-carry adder

3.4 Carry-look ahead adders

To reduce the computation time, engineers devised faster ways to add two binary numbers by using carry-look ahead adders. They work by creating two signals (P and G) for each bit position, based on whether a carry is propagated through from a less significant bit position (at least one input is a '1'), generated in that bit position (both inputs are '1'), or killed in that bit position (both inputs are '0'). In most cases, P is simply the sum output of a half-adder and G is the carry output of the same adder. After P and G are generated the carries for every bit position are created. Some

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

advanced carry-look ahead architectures are the Manchester carry chain[6], Brent–Kung adder, and the Kogge–Stone adder .Some other multi-bit adder architectures break the adder into blocks. It is possible to vary the length of these blocks based on the propagation delay of the circuits to optimize computation time. These block based adders include the carry-skip (or carry-bypass) adder which will determine P and G values for each block rather than each bit, and the carry select adder which pre-generates the sum and carry values for either possible carry input (0 or 1) to the block, using multiplexers to select the appropriate result when the carry bit is known.

4. PROPOSED METHOD

4.1 Description

Quantum Dot Cellular Automata (sometimes referred to simply as quantum cellular automata, or QCA) are proposed models of quantum computation, which have been devised in analogy to conventional models of cellular automata introduced by von Neumann. Standard solid state QCA cell design considers the distance between quantum dots to be about 20 nm, and a distance between cells of about 60 nm. Just like any CA, Quantum (-dot) Cellular Automata[7] are based on the simple interaction rules between cells placed on a grid. A QCA cell is constructed from four quantum dots arranged in a square pattern. These quantum dots are sites electrons can occupy by tunneling to them.

QCA Device Background

Quantum Dot cellular Automata (QCA) don't use transistors.QCA design addresses the issue of device density and interconnection. The basic element of QCA is a quantum cell. Each quantum cell has electrons in them. Where electron transmissions occur on the columbic interaction of the electrons.QCA is an advanced research program and efforts are made to reduce the complexity of the circuits. When featured size is reduced to nanometers than Quantum effects such as Tunneling takes place[1].QCA circuits can be directly obtained from conventional designs with addition of special clocking system. This provides very easy transmission of conventional circuits to get transformed into QCA structures. QCA structures are designed as an array of quantum cells. Where every cell has electrons in them where electrostatic interaction with its neighboring cells takes place.QCA uses a new technique for computation. It uses polarization effect rather than conventional current for the transmission of information which contains the digital information. Thus a cell is responsible for the transfer of information throughout the circuit. The basic operators used in QCA are three input majority gates and invertor. This study proposes the design of different types of adders. Hence the proposed design is used to minimize the area and complexity.

B. QCA Design Architecture 1. Basics of QCA The basic elements of QCA are QCA cell, Majority gate and Invertor. These is important elements. In QCA cell each cell is having four quantum dots and is having two free electrons. The locations of the electrons determine the binary states.Fig.6 shows the QCA cell diagram.

QCA Design Architecture

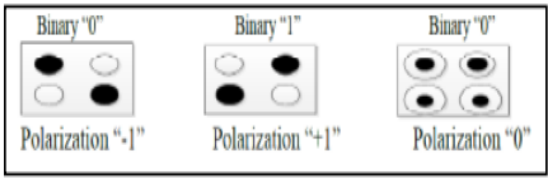

Fig.6 Simplified Diagram of QCA Cell Polarization

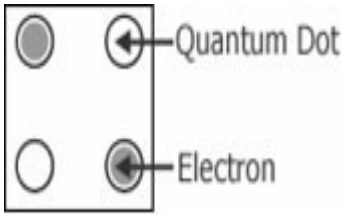

Fig .7Four Dot Quantum Cell

QCA Cell:

A quantum-dot cellular automata (QCA) is a square nanostructure of electron wells having free electrons. Each cell has four quantum dots [2].The four dots are located in the four corners .The cell can be charged with two free electrons. By using the clocking mechanism, the electrons tunnel to proper location during the clock transition. Thus there exist two equivalent energetically arrangements of the two electrons in the QCA cell as shown in Fig. 6.These two arrangements can represent logic 1 and logic 0 respectively so that binary information can be

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

encoded. Invertors is represented in Fig. 8 and Majority gate in Fig. 9.

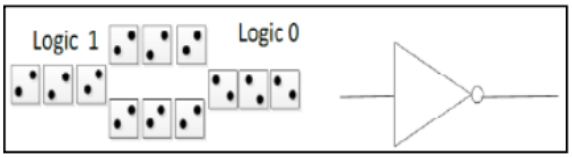

Fig 8. Invertor

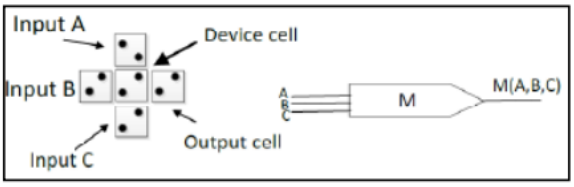

Fig 9.Majority gate

4.3 Architecture of Basic Novel 2-bit adder

To introduce the novel architecture proposed for implementing ripple adders in QCA, let consider two n-bit addends A = an−1, . . . , a0 and B = bn−1, . . . , b0 and suppose that for the I th bit position (with i = n − 1, . . . , 0) the auxiliary propagate and generate signals, namely pi = ai + bi and gi = ai · bi , are computed. ci being the carry produced at the generic (i−1) bit position, the carry signal ci+2, furnished at the (i+1)th bit position, can be computed using the conventional CLA logic reported in (2). The latter can be rewritten as given in (3), by exploiting Theorems 1 and 2 demonstrated in [15]. In this way, the RCA action, needed to propagate the carry ci through the two subsequent bit positions, requires only one MG. Conversely, conventional circuits operating in the RCA fashion, namely the RCA and the CFA, require two cascaded MGs to perform the same operation.

In other words, an RCA adder designed as proposed has a worst case path almost halved with respect to the conventional RCA and CFA is exploited in the design of the novel 2-bit module shown in Fig. 1 that also shows the computation of the carry ci+1 = M(pi gici). The proposed n-bit adder is then implemented by cascading n/2 2-bit modules. Having assumed that the carry-in of the adder iscin = 0, the signal p0 is not required and the 2-bit module used at the least significant bit position is simplified.
The sum bits are finally computed. It must be noted that the time critical addition is performed when a carry is generated at the least significant bit position (i.e., g0 = 1) and then it is propagated through the subsequent bit positions to the most significant one. In this case, the first 2-bit module computes c2, contributing to the worst case computational path with two cascaded MGs. The subsequent 2-bit modules contribute with only one MG each, thus introducing a total number of cascaded MGs equal to (n − 2)/2. Considering that further two MGs and one inverter are required to compute the sum bits, the worst case path of the novel adder consists of (n/2) + 3 MGs and one inverter.

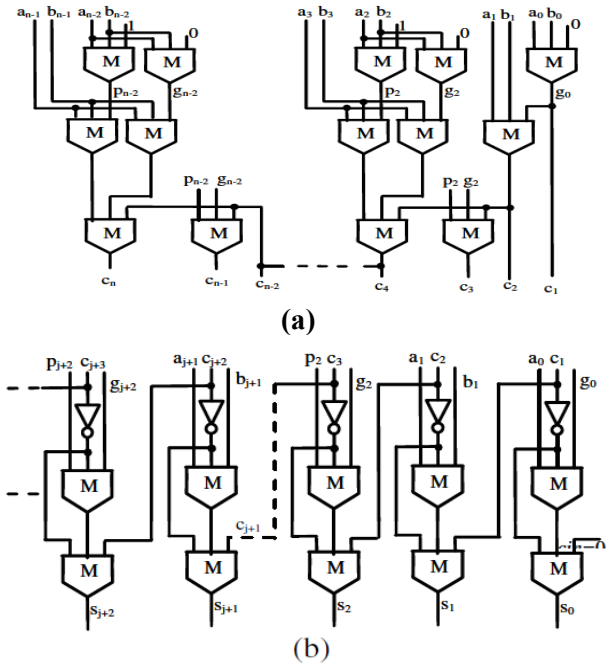

Fig.10 Novel n-bit adder (a) carry chain and (b) sum block.

5. RESULT

The proposed addition architecture is implemented for several operands word lengths using the QCA. Designer tool adopting the same rules and simulation settings. The QCA cells are 18 nm wide and 18 nm high, the cells are placed on a grid with a cell center to center distance of 20 nm; there is at least one cell spacing between adjacent wires; the quantum-dot diameter is 5 nm; the multilayer wire crossing structure is exploited; a maximum of 16 cascaded cells and a minimum of two cascaded cells per clock zone are assumed. The coherence vector engine is used for simulations with the options. Layouts for the 16, 32 and 64-bit versions of the novel adder. Simulation results for the 64-bit adder. There, the carry out bit is included in the output sum bus. Because of the limited

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

QCA[8] Designer graphical capability, input and output busses are split into two separate more significant and less significant busses. The polarization values of few single output signals.

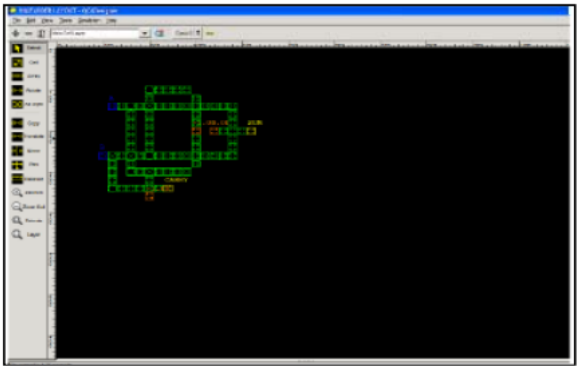

Fig11. Design of Half Adder using QCA

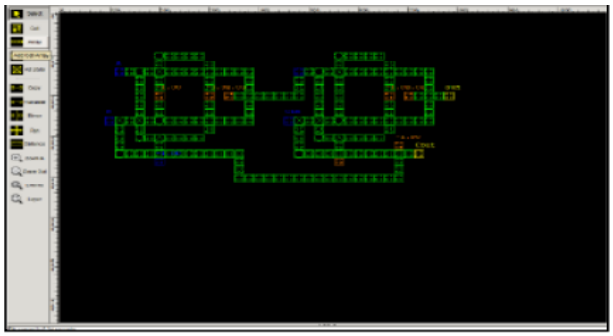

Fig 12.Design of Full adder using two half adder in QCA

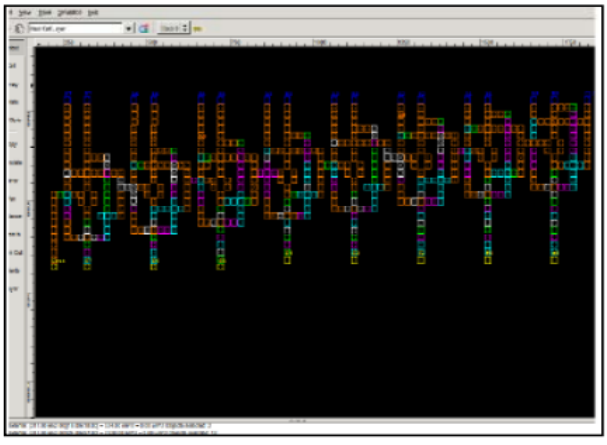

Fig13. Design of 8-bit Ripple Carry Adder using QCA.

Simulations performed on 32- and 64-bit adders have shown that the first valid result is outputted after five and nine latency clock cycles, respectively. As an example, the 20 clock phases (or five cycles delay) of the 32-bit adder are as follows: one clock phase is needed for inputs acquisition; the carry c2 related to the least significant bit positions is then computed within the two subsequent clock phases; 15 phases are required for the carry propagation through the remaining bit positions; finally, two more phases are needed for the sum computation. The number of cascaded MGs within the worst case computational path directly impacts on the achieved speed performances as an MG always adds one more clock phase.

The layout strategy is also quite important. In fact, designs using the same basic logic with the same worst case theoretical path can have different number of clock phases due to differently compact layouts. It should also be noted that the critical path of the HYBA contains the fewest MGs, while the novel adder, the RCA and the CFA require less additional clock phases exceeding the number of cascaded MGs. This means that their layouts do not contain overlong wires, thus demonstrating that the novel architecture inherits logic advantages of CLA and parallel-prefix adders (i.e., the short computational path), but limiting, as happens in the RCA and CFA structures, detrimental layout effects[9]. Comparison results for operands word lengths ranging from 8- to 64-bit also show that the novel adder achieves the lowest delay and spans over an area similar to that occupied by the cheaper designs known in literature. Therefore, our design approach allows the best area-delay tradeoff to be achieved. The numbers of wire crossovers for some competitors. As it is well known, wire crossovers are sensitive to fabrication imperfections and consequently they can represent a reliability issue. It can be seen that the number of crossovers used in the novel 8- and 16- bit adder was 33.6% and 37%.The robustness[10] of the proposed circuit was evaluated as a function of temperature using a simulation set-up identical to that. It must be noted that the time critical addition is performed when a carry is generated at the least significant bit position and then it is propagated through the subsequent bit positions to the most significant one. In this case, the first 2-bit module computes causal to the worst case computational path with two cascaded MGs. The subsequent 2-bit modules contribute with only one MG each, thus introducing a total number of cascaded MGs equal to $(n - 2)/2$. Considering that additional two MGs and one inverter are required to compute the sum bits, the

worst case path of the novel adder consists of (n/2) + 3 MGs and one inverter.

6. CONCLUSION

We have implemented a new 128 bit adder designed in QCA. It achieved performance of high speed when compared to all the existing QCA adders, with an area needed is cheap comparable with the RCA and CFA. Also the cell count required is less when compared to 64 bit adder. The unnecessary clock phases are avoided due the adopted basic logic and layout strategy. Hence, the QCA architecture is therefore, low area, low delay, simple and efficient for VLSI hardware implementation. The reduced number of gates of this work offers the great advantage in the reduction of area and also the total delay. The QCA architecture is therefore, low area, low delay, simple and efficient for VLSI hardware implementation. It would be interesting to test the design of the modified 128-bit Novel adders. In addition, because of the adopted basic logic and layout strategy, the number of clock cycles required for completing the elaboration was limited.

REFERENCES

[1] C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernestein, "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: Layout Timing," Int. J. Circuit Theory Appl., vol. 29, no. 1, pp. 49–62.

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

[4] W. Liu, L. Lu, M. O'Neill, and E. E. Swartzlander, Jr., "Design rules for quantum-dot cellular automata," in Proc. IEEE Int. Symp. CircuitsSyst., May 2011, pp. 2361–2364.

[5] K. Kim, K. Wu, and R. Karri, "Toward designing robust QCA architectures in the presence of sneak noise paths," in Proc. IEEE Design ,Autom. Test Eur. Conf. Exhibit., Mar. 2005, pp. 1214–1219.

[6] K. Kong, Y. Shang, and R. Lu, "An optimized majority logic synthesis methology for quantum-dot cellular automata," IEEE Trans. Nanotechnol., vol. 9, no. 2, pp. 170–183, Mar. 2010.

[7] K. Walus, G. A. Jullien, and V. S. Dimitrov, "Computer arithmetic structures for quantum cellular automata," in Proc. AsilomaConf.Sygnals, Syst. Comput., Nov. 2003, pp. 1435–1439.

[8] J. D. Wood and D. Tougaw, "Matrix multiplication using quantumdot cellular automata to implement conventional microelectronics," IEEE Trans. Nanotechnol., vol. 10, no. 5, pp. 1036–1042, Sep. 2011.

[9] K. Navi, M. H. Moaiyeri, R. F. Mirzaee, O. Hashemipour, and B. M. Nezhad, "Two new low-power full adders based on majority-not gates," Microelectron. J., vol. 40, pp. 126–130, Jan. 2009.

[10] L. Lu, W. Liu, M. O'Neill, and E. E. Swartzlander, Jr., "QCA systolic array design," IEEE Trans. Comput., vol. 62, no. 3, pp. 548–560, Mar. 2013.