Design and Characterization of An And-Or-Inverter (AOI) Gate for QCA Implementation

Jing Huang, Mariam Momenzadeh, Mehdi B. Tahoori and Fabrizio Lombardi
Dept of Electrical and Computer Engineering
Northeastern University
Boston, MA 02115
{hjing,mmomenza,mtahoori,lombardi}@ece.neu.edu

ABSTRACT

Quantum-dot Cellular Automata (QCA) offers a new computing paradigm in nanotechnology. The basic logic elements of this technology are the inverter and the majority voter. In this paper, we propose a novel complex and universal QCA gate: the And-Or-Inverter (AOI) gate, which is a 5 input gate consisting of 7 cells. This paper presents a detailed simulation-based analysis of the AOI gate as well as the characterization of QCA defects and study of their effects at logic level. Design implementations using the AOI gate are compared with the conventional CMOS and the majority voter-based QCA methodology. Testing of the AOI gate at logic level is also addressed, unique testing features of designs based on this complex gate have been investigated.

Categories and Subject Descriptors: J.6.1 [Computer-Aided Design]
General Terms: Design
Keywords: QCA, Test, Defect Characterization

1. INTRODUCTION

As the CMOS technology is approaching it's fundamental physical limits, there has been extensive research in nano-regimes for the future generation ICs. QCA is one of the promising new technologies that not only gives a solution at nano scale, but also it offers a new method of computation and information transformation [5]. It is predicted that QCA cells of few nanometer size can be fabricated through molecular implementation and operate at THZ frequencies.

For QCA the cells must be precisely aligned at nano scale to provide correct functionality, therefore proper testing of these devices for manufacturing defects and misalignment is essential. Additionally, the QCA devices are quite different from the conventional CMOS designs, so different test schemes are needed.

The basic logic elements in QCA are the *majority voter* (MV) and the inverter. Although MV can be easily programmed into either AND or OR gates, it suffers from the fact that the majority function is not universal and it is not favorable in terms of synthesis using existing tools. In this paper, we present a novel, complex and universal QCA gate: the so-called AOI (And-Or-Inverter) gate, which is a 5 input gate consisting of 7 cells. A detailed simulation-based characterization of the AOI gate is presented. Various fabrication defects and their effects on the AOI gates are also addressed. Testing of the AOI gate at logic level is investigated and unique features of design based on this complex gate is identified.

The rest of this paper is organized as follows. In Sec 2, a review of QCA is presented. The QCA implementation of the AOI gate is presented in Sec 3 while the defect characterization for the QCA AOI gate is in Sec 4. Test sets, defect, and fault coverage are discussed in Sec 5. Finally, Sec 6 concludes the paper.

2. REVIEW OF QCA

QCA is a novel nano device that stores logic states not as voltage levels but rather based on the position of individual electrons. A quantum cell can be viewed as a set of four charge containers or dots, positioned at the corners of a square. The cell contains two extra mobile electrons which can quantum mechanically tunnel between dots, but not cells. The electrons are forced to the corner positions by Coulomb repulsion. The two possible polarization states represent logic "0" and logic "1", as shown in Figure 1. Unlike conventional logic, information is transferred with Coulomb interaction in QCA which connects the state of one cell to the state of its neighbors. This results in a technology in which information transfer (interconnection) is the same as information transformation (logic manipulation). Power dissipation in QCA circuits is ultra low compared with conventional CMOS circuits [5][8][9].

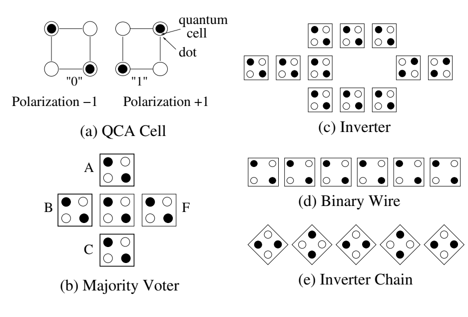
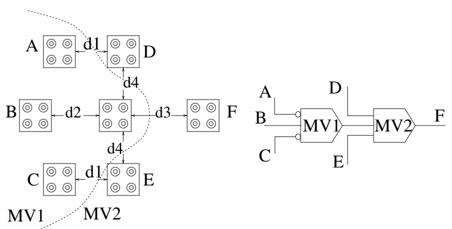


Figure 1: QCA devices

The basic logic gate in QCA is the 5-cell MV with logic function $F = AB + AC + BC$, as shown in Figure 1(b). Logic AND and OR functions can be implemented with MV by setting one input permanently to "0" or "1", respectively. The QCA Inverter is shown in Figure 1(c), which consumes considerable area in QCA. The binary wire and the inverter chain are used as interconnects in QCA circuits, as shown in Figure 1(d)(e). Various high-level designs with QCA technology including microprocessors, FPGA and memories have been proposed [4][10][13]. Micro-sized QCA devices have been fabricated with metal cells which operate at 50mK [5]. Current research in manufacturing has been focused on molecular realizations of QCA that operates at at room-temperature[3].

3. AOI GATE CHARACTERIZATION



(a) AOI Gate Layout (b) AOI Gate Schematic

Figure 2: The AOI Gate

Although MV can be easily adapted to realize AND or OR, it suffers from the disadvantage that it's not an universal gate and can not offer the inverting function. Since inverting is expensive in QCA (unlike conventional CMOS), built-in inversion is desirable. Moreover, in our experiments we found that the MV is not favorable in terms of synthesis. This motivates us to build a complex QCA gate with embedded AND, OR and INV functions, and with better usability for synthesis.

Also the device has to be stable, which means that: (1) the output must exhibit a definite polarization; (2) small displacements of individual cells should not change the logic function of the device, i.e. the device should provide certain degree of tolerance to faults and defects; (3) wiring of the device should not change the logic function.

Here we present a complex universal gate: the And-Or-Inv gate (AOI gate); the layout and corresponding logic schematic is illustrated in Figure 2. This is a 7 cell gate with 5 input cells, one device cell and one output cell. We build the gate from the original 5-cell MV by adding two extra inputs (cells A and C); these two inputs have an inverting effect on the center cell as it can be seen from the layout of the inverter in Figure 1 that cells in a diagonal orientation tend to exhibit an inverting function. The logic function realized by the AOI gate is:

$$\begin{aligned} F &= DE + (D+E)(\overline{A}\,\overline{C} + \overline{A}B + B\overline{C}) \\ &= Maj(D, E, Maj(\overline{A}, B, \overline{C})) \qquad (1) \end{aligned}$$

where $Maj()$ is the 3-input majority function. By simulation we find that this AOI gate is relatively stable, i.e. moving marginally the cells doesn't change its function. We have found that the placement in Table 1 (see Figure 2) yields an AOI gate performing the function described above. We use a semi-symmetric and stable configuration with $d1 = d3 = d4 = 25nm$, $d2$=$35nm$.

$d1$ nm	$d2$ nm	$d3$ nm	$d4$ nm
20	30-40	20-40	20
25	30-40	20-40	20
30	35-40	20-40	20
25	35-40	25-40	25
30	35-40	25-40	25

Table 1: cell placement in the AOI(not wired) gate

The AOI gate is logically equivalent to a concatenation of two MVs with 2 complemented inputs (A and C). The layout of the AOI gate consists of two nested MVs. $MV1$ performs the function $MV1 = Maj(\overline{A}, B, \overline{C})$. It has been shown in our previous work [11] that the horizontal input (i.e. B) has the strongest influence on the center cell in a MV. Therefore in the AOI gate, cell B is placed further away than A and C (see Table 1). Since A and C tend to have an inverted effect on the center cell, $MV1$ is the majority of $\overline{A}$,B and $\overline{C}$. The second majority voter is $MV2 = Maj(D, E, MV1)$.

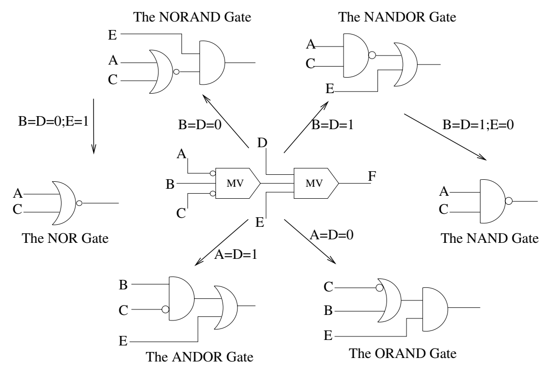


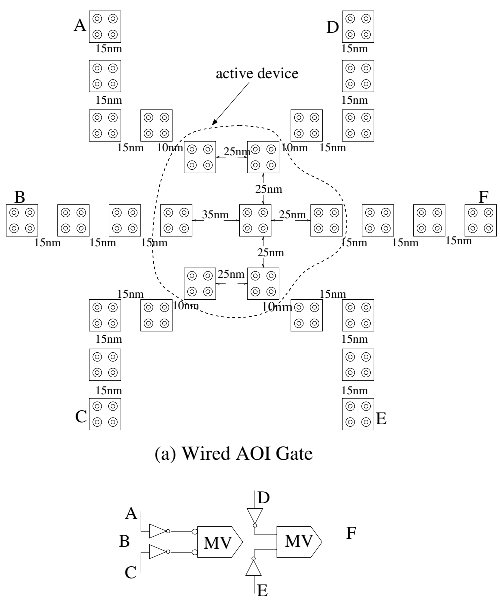
Figure 3: Various Logic Function Realized by an AOI Gate

Our experiments on synthesis show that existing tools such as Synopsis Design Compiler [7] do not make efficient use of MVs. Even for arithmetic circuits, the tool does not map the carry function in a full adder to MV (which is an exact match). With the existing tools, MVs can only be utilized as 2-input AND or 2-input OR gates. The AOI gate can be configured to perform a variety of logic functions, some of these functions are shown in Figure 3. Experiments show that the synthesis tool makes better use of the gates derived from the AOI gate. The synthesis for ISCAS85 benchmark circuits using 13 different gates implemented with the AOI gate results in up to 43% improvement compared to the synthesis results using MV and inverter in terms of cell count in active devices [1]. The AOI gate is universal and offers more flexibility for design implementation.

The proposed wiring scheme for the AOI gate is shown in Figure 4; the active AOI gate has $d1 = d3 = d4 = 25nm$, $d2$=$35nm$. As our previous research in [11] indicates, when two binary wires are placed close enough to each other, they tend to interfere with each other, similar to the crosstalk in conventional CMOS circuits. The main challenge in wiring the AOI gate is to separate the input/output binary wires such that they do not interfere with each other while still preserving the original logic function. By simulation we find that the wire for cell A and D (also C and E) must have a distance of more than $25nm$, and wiring for inputs A,B,C and D has inverting effect (shown as extra inverters for these inputs). Hence the device is wired as shown in Figure

4. Based on simulation, the AOI gate with the above wiring scheme has been found to be quite stable.

The AOI gate consists of only 7 cells. If we were to implement the same logic with MV and inverters in QCA, 26 cells are needed. Alternatively we can implement the same logic in CMOS with 4 2-input NAND gates, 2 2-input NOR gates and 1 inverter, altogether 26 transistors. Next consider the full adder. [12] proposed a full adder built with 3 MVs and 1 INV (25 cells). Here we show a much simpler implementation with only 2 AOI gates (14 cells), as shown in Figure 5.



(a) Wired AOI Gate

(b) Schematic of Wired AOI Gate

Figure 4: The Wired AOI Gate

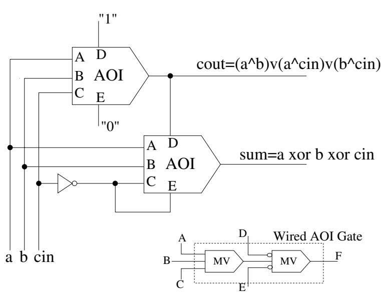


Figure 5: Full adder implemented with the AOI gate

4. DEFECT CHARACTERIZATION OF THE AOI GATE

In this section, the robustness of the AOI gate is investigated. The basic functionality of a QCA device is based on the Coulombic interactions among neighboring QCA cells, which depends on the accuracy and geometry of their implementation. Various configurations of the AOI gate have been studied using the QCADesigner[1] v1.20 simulation tool. The bistable model is used in the simulation engine.

[1]QCADesigner is the product of an ongoing collaboration between the ATIPS laboratory and University of Notre Dame.

d1	*d4*	*d2*	*d3*	Output Function F
25	25	5-10	5-40	
		15	15-40	
		20	20-40	
		25	25	B
		15	5-10	
		20	5-15	$\overline{D}\ \overline{E} + B\overline{D} + B\overline{E} = Maj(B, \overline{E}, \overline{D})$
		25-30	5-20	
		25	30-40	
		30	25-40	$DE + BE + BD = Maj(B, D, E)$
		35-40	5-20	$\overline{D}\ \overline{E} + (\overline{D} + \overline{E})(\overline{A}B + B\overline{C} + \overline{A}\ \overline{B}\ \overline{C})$
		35-40	25-40	Normal Operation
		≥45		For some input combinations F=Z(no polarization)
			≥45	For all input combination F=Z(no polarization)

Table 2: Defect Characterization of the AOI gate

For defect characterization of QCA devices and circuits and to study their effects at the logic level, appropriate defect mechanisms and models must be considered which 1) can be simulated using available tools; 2) are realistic to model manufacturing and fabrication defects. According to [2], in the present stage of QCA manufacturing, defects are possible in both *synthesis phase*, in which the individual cells (molecules) are manufactured, and *deposition phase* in which the cells are placed in a specific location in the surface. Manufacturing defects during synthesis may cause a cell to have missing or extra dots or/and electrons. However, defects are much more likely to occur in the deposition part than in the synthesis part which will result in cell misplacement. A *cell displacement* is a defect in which the defective cell is misplaced from its original direction. In this work, we assume each individual cell functions correctly and simulate cell displacements with respect to the central cell under different distance conditions. Studying the behavior of the AOI gate in the presence of cell displacements not only helps to establish its fault tolerance property but also it gives an insight into cell interactions within the AOI gate structure.

In our simulation the fault free AOI gate (as in Figure 2) has $d1 = d3 = d4 = 25nm$, $d2$=$35nm$. We then move the input and output cells of the AOI gate with respect to the central cell and record the logic function performed by the AOI gate. Part of the simulation results with $d1 = d4 = 25nm$ are shown in Table 2. We find in our experiments that similar faulty patterns occur with $d1 = d4 = 20nm$ or $d1 = 25nm$,$d4 = 20nm$ or $d1 = 30nm$,$d4 = 20nm$ or $d1 = 30nm$,$d4 = 25nm$.

An important result observed from Table 2 is that the horizontal input (cell B) has greater influence on the central device cell than the other inputs, which confirms our results of [11]. If Cell B is placed close enough to the central cell, the output follows cell B, the whole AOI gate acts as a binary wire with input B. Two other interesting fault patterns can be observed where the AOI gate behaves as a MV with $F = DE + BE + BD = Maj(B, D, E)$ or a MV with inversions at some inputs with $F = \overline{D}\ \overline{E} + B\overline{D} + B\overline{E} = Maj(B, \overline{E}, \overline{D})$. In these two cases B is closer to the central cell than in the fault free case and cancels out the effect of A and C. When output F is placed sufficiently far from the central cell, no polarization can be observed at the output (indicated by $F = Z$ in the table). Also when B is placed far away from the central cell, some input combinations cause the output to show no polarization at all. These results are consistent because in QCA, information is transmitted via Coulomb interactions, the larger the distance between two cells, weaker the interactions. It can also be concluded that the AOI gate is reasonably robust as small displacement does not change the functionality.

5. LOGIC LEVEL TESTING OF THE AOI GATE

The overall structure of the QCA implementation of (combinational) logic designs is shown in Figure 6. The block consists of an interconnection of AOI gates. There are two system-level control lines, U_0 and U_1, which are connected to AOI gates in the design. U_0 and U_1 are connected to logic "0" and "1", respectively.

Although in the current CMOS process only a small portion of the actual defects behaves like stuck-at faults, the stuck-at fault model is still widely used as the test sets generated based on this model are quite acceptable. So it is important to investigate the effectiveness of stuck-at test sets for QCA defects even though QCA defect mechanisms cannot be modeled by the stuck-at model. The logic-level testing properties of logic networks consists of MVs and inverters have been investigated in [11]. If all bits of the primary inputs V are flipped, $V \rightarrow \overline{V}$, then all nodes in the network will also be flipped. Moreover, for any node n in the network, n stuck-at-u is detected by V, if and only if n stuck-at-$\overline{u}$ is detected by $\overline{V}$. Since the gate-level representation of an AOI gate consists of only MVs and inverters, all properties for a logic network of MVs and inverters hold for a network of AOI gates.

Property 1. *If all inputs of an AOI gate are flipped, the output will be also flipped.*

Property 2. *Consider an arbitrary network of AOI gates with primary input vector V. If all bits of V are flipped, $V \rightarrow \overline{V}$, all nodes in the network will be flipped.*

Corollary 1 *The test vector pair $(V, \overline{V})$, where V is any arbitrary vector, causes a transition on all nodes of the network. Also, the three vectors $(V, \overline{V}, V)$ cause both fall and rise transitions on all nodes in the network, a 100% toggle fault coverage test.*

Property 3. *Consider an arbitrary network of AOI gates with primary input vector V. For any node n in the network, n stuck-at-u is detected by V, if and only if n stuck-at-$\overline{u}$ is detected by $\overline{V}$.*

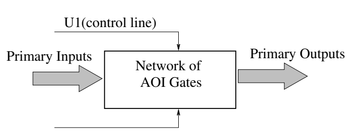


Figure 6: The QCA Implementation of Logic Networks Using the AOI Gate

Note that Property 3 does not hold for other logic functions. For example, test vector $V = 11$ detects stuck-at-0 at the two inputs of an AND gate, but $\overline{V} = 00$ does not detect any single stuck-at-1 on the inputs.

Corollary 2 *The fault list for any network of AOI gates consists of only one fault per node (either stuck-at-1 or stuck-at-0), because if a vector V detects one stuck-at fault on that node, $\overline{V}$ will detect the other stuck-at fault on that node.*

We have developed a test set with 100% coverage to detect all cell displacement defects presented in Sec 4, which consists of only two vectors: $ABCDE$ ={ 00000,00001}. We have also considered single stuck-at fault test sets for the AOI gate and found that at least 4 vectors are needed for 100% PIN stuck-at fault coverage; for example the set {01110, 00101, 00000, 00001}. It is interesting to observe that in this case, the 100% PIN fault test set can detect all single defects discussed in Sec 4. This is a very useful feature because we only need to consider the PIN faults to achieve 100% coverage of the internal defects.

6. CONCLUSION AND FUTURE WORK

Quantum-dot cellular automata (QCA) are novel devices which are promising in the era of nano scale computing. In this paper, we have proposed a novel complex QCA logic gate: the AOI gate. The AOI gate forms universal logic; it offers not only AND,OR but also the INV function. A detailed simulation-based analysis and a characterization of QCA defects are presented. AOI gate is more favorable than MV in terms of logic synthesis. This gate can be efficiently used by existing synthesis tools. Testing at the logic level is also addressed and appropriate test sets are developed. Future research will also include bridging faults among the wiring of the AOI gate.

Acknowledgment

The authors would like to thank Prof. Marya Lieberman at Department of Chemistry and Biochemistry, University of Notre Dame and Konrad Walus at ATIPS Lab, University of Calgary for helpful discussions and feedbacks.

7. REFERENCES

[1] M.B.Tahoori, J.Huang, M.Momenzadeh, F.Lombardi, "Design and Synthesis of Digital Circuits Using Nano-Scale Quantum Dot Cellular Automata", *Internal Report, ECE Dept. Northeastern University*, 2004.

[2] Personal communication with Professor Marya Lieberman, Department of Chemistry and Biochemistry, University of Notre Dame, IN, USA.

[3] M.Lieberman, S.Chellamma, B.Varughese, Y.Wang, C.S.Lent, G.H.Bernstein, G.Snider, F.Peiris, "Quantum-Dot Cellular Automata at a Molecular Scale", *Annals of the New York Academy of Sciences*, vol 960, pp. 225-239, 2002.

[4] M.T. Niemier, P.M.Kogge, "Logic-in-Wire: Using Quantum Dots to Implement a Microprocessor", *International Conference on Electronics, Circuits, and Systems (ICECS '99)*,1999.

[5] A.O.Orlov, I.Amlani, G.H.Bernstein, C.S.Lent, G.L.Snider, "Realization of a Functional Cell for Quantum-Dot Cellular Automata", *Science*, vol 277, pp 928-930, 1997.

[6] QCADesigner Home Page: www.atips.ca/projects/qcadesigner/

[7] Synopsys Inc.,"Synopsys Online Documentation", v2001.08

[8] P.D.Tougaw and C.S. Lent, "Logical Devices Implemented Using Quantum Cellular Automata", *Journal of Applied Physics*, Vol 75(3), pp. 1818-1825, 1994.

[9] P.D.Tougaw, C.S.Lent, "Dynamic Behavior of Quantum Cellular Automata", *Journal of Applied Physics*, vol 80(8), pp. 4722-4736, 1996.

[10] V. S. Dimitrov, G. A. Jullien, K. Walus, "Quantum-Dot Cellular Automata Carry-Look-Ahead Adder and Barrel Shifter", *IEEE Emerging Telecommunications Technologies Conference*, 2002.

[11] M.B.Tahoori, M.Momenzadeh, J.Huang, F.Lombardi, "Defects and Faults in Quantum Cellular Automata at Nano Scale", *VLSI Test Symposium*, 2004.

[12] W.Wang, K.Walus, G.A.Jullien, "Quantum-Dot Cellular Automata Adders", *IEEE Nano 2003 Conference*, 2003.

[13] K.Walus, A.Vetteth, G.A.Jullien, V.S.Dimitrov, "RAM Design Using Quantum-Dot Cellular Automata", *NanoTechnology Conference*,vol 2, pp. 160-163, 2003.