

PLAs in Quantum-dot Cellular Automata

Xiaobo Sharon Hu Michael Crocker
Dept. of Comp. Sci. & Eng.
University of Notre Dame
Notre Dame, IN 46545, USA
Email: {shu, mcrocker}@nd.edu

Michael Niemier
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332, USA
Email: mniemier@cc.gatech.edu

Minjun Yan Gary Bernstein
Dept. of Elec. Eng.
University of Notre Dame
Notre Dame, IN 46545, USA
Email: {myan, gbernste}@nd.edu

Abstract— Research in the fields of physics, chemistry and electronics has demonstrated that Quantum-dot Cellular Automata (QCA) is a viable alternative for nano-scale computing. However, little work on QCA has studied designing implementation-friendly programmable QCA circuits. This paper fills this gap by presenting a novel QCA-based Programmable Logic Array (PLA) structure. In addition to being compact, the proposed PLA structure exploits some unique properties of QCA cells to achieve ease of implementation, programming and defect detection. These features are indispensable to the successful adoption of any nano-scale circuits.

I. INTRODUCTION

In order to sustain the remarkable growth rate in computing performance, device research is clearly needed. However, equally important are research efforts in computational models, circuits, and architectures for the devices. Not surprisingly, advances in one aspect do not necessarily correlate to advances in others. For example, quantum computing is a potentially powerful computational model that could easily solve several notoriously difficult problems (such as factoring large numbers) [36]. [2] shows that it is possible to create architectures that can handle the immense overhead of required error correction. However, to date, the largest number factored is 15 [21]. On the other hand, advances in nano-scale technologies and devices based on traditional diode or transistor models (i.e., using nanowires or carbon nanotubes) seem to indicate that fabricating these nano-scale circuits is markedly more plausible in the nearer-term [7], [35].

This paper focuses on another computational model, Quantum-dot Cellular Automata (QCA). First proposed in the early 1990s [32], QCA accomplishes logical operations and data movement via Coulombic interaction rather than electric current flow. It has been shown that QCA-based circuits could be clocked at an extremely high frequency (adiabatically at 1 THz [29] with cell switching times on the order of 10^{-12} to 10^{-15} seconds [30], [3]), potentially leading to circuits with densities that are multiple orders of magnitude beyond what end-of-the-curve CMOS can provide [22], and dissipate very little power [29]. Simple QCA circuits based on metal-dots have already been demonstrated [1]. More importantly, recent advances in DNA tiling and molecular self-assembly show great promise in implementing molecular QCA circuits [26], [18], [17]. As QCA-based systems could provide various “wins” over end-of-the-curve CMOS, an existing body of research on device physics has been joined by efforts that

explored the construction of traditional computer architectures [22], [24], provided simulators [33], considered how to test circuits [28]. Recent research has begun to examine the implications of physically implementing QCA-based circuits [5].

It is well recognized that realized molecular electronic circuits must be able to tolerate higher percentages of defects than current CMOS circuits. The task of ensuring that a system is still functional post-fabrication falls largely to the circuit designer or computer architect who, in many instances, have leveraged reconfigurable/reprogrammable structures. Largely for this reason, Programmable Logic Arrays (PLAs) have been studied by a number of research groups for different nanoscale technologies. For example, DeHon and Wilson proposed a nano-wire-based sublithographic PLA design [8]. Likharev and Strukov presented CMOL FPGA [27]. Hogg and Snider showed interesting algorithmic approaches to map logic functions to PLAs with defective crossbar switches. Though many papers have been published on designing QCA-based circuits, only a few have examined design issues with programmable structures that use QCA cells [10], [23], [14]. However, these designs either are difficult to implement or sacrifice performance.

In this paper, we present a novel PLA structure based on the QCA device architecture. It is fully re-programmable and this programmability is achieved by exploiting the fact that QCA logic can be bidirectional. Furthermore, the PLA structure allows defects to be readily detected and isolated, again by exploiting logic bidirectionality. The design is compact and easily extensible. Even with the consideration of near- to mid-term implementation constraints, the size of a QCA PLA is still very small. We have verified the functionality of our proposed design through simulations based on statistical mechanics for a 4-dot molecular QCA cell. Since correction operation of a QCA circuit is also dependent on the clocking fields controlling the QCA circuit, we have investigated the impact of utilizing CMOS technology to generate the necessary electric field distribution for our PLA structure. Several representative clocking wire layouts were studied and their resulting electric fields were simulated with a commercial Finite-Element based software [9].

II. BACKGROUND AND RELATED WORK

QCA represents information by encoding binary numbers into cells having a bi-stable charge configuration. A QCA cell

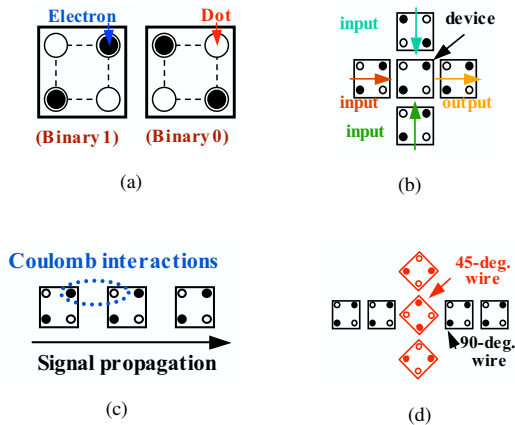


Fig. 1. (a) Two 4-dot QCA cells. (b) A majority gate made of 4 QCA cells. (c) A QCA wire. (d) Physically crossing two QCA wires.

can consist of 2 or 4 “charge containers” (i.e., quantum dots) and 1 or 2 excess charges, respectively. One configuration of charge represents a binary ‘1’, while the other a binary ‘0’ (Fig. 1(a)) [20]. Logical operations and data movement are accomplished via Coulomb interactions. QCA cells interact because the charge configuration of one cell alters the charge configuration of the next cell. Additionally, information transmission and processing is carried out by the same entities – QCA cells – rather than by separate devices and wires. As QCA cells are made smaller, interconnects shrink too.

Fig. 1(b)–Fig. 1(d) illustrate several basic QCA circuit elements [32], [22]. The majority gate (Fig. 1(b)) implements logic function $AB + BC + AC$. The output cell assumes the polarization of the majority of the 3 input cells [20]. By setting one of the majority gate inputs to logic ‘0’ or ‘1’, the gate reduces to an AND or OR gate, respectively. A QCA wire (Fig. 1(c)) is just a line of QCA cells; at the input end, one cell must be polarized to act as the driver for the wire. The cells do not need to be spaced exactly the same distance apart. QCA wires with different orientations (Fig. 1(d)) can cross in the plane without destructing the binary value on either wire. An inverter can also be built easily with QCA cells [32].

A “clock” structure is required to provide signal gain for a QCA circuit. The clock structure supplies a necessary electric field for groups of QCA cells to transition from a null state to a bistable, active state, and then back to the null state. A QCA cell can only change its charge configuration (from a logic ‘0’ to ‘1’ or vice versa) by going through a transition from the null state to the active state. Specifically, a negative electric field first pulls the charges in QCA cells “down” so that the cells are put in the null state, then a positive electric field pushes charge “up” to the active state, while a driver (a cell already in the active state) determines if a cell just turned on will be a 1 or 0. If the positive electric field is maintained, QCA cells will keep their original charge configurations.

For systems of molecular QCA cells (the target of this paper), the clock structure, i.e., the electrical field, can be generated using wires formed by conventional lithography [11], [29] on a silicon substrate to which QCA molecules can be attached [13]. The phases of a clock signal could take the form

of time-varying, repetitious voltages applied to such CMOS wires. Multiple wave fronts could exist simultaneously – even along the same QCA wire – resulting in inherent pipelining determined by the granularity of the clock structure [20], [24].

In a **molecular** implementation, a QCA device could be made from a single chemical molecule. A recent experiment demonstrates that applying reasonable electric fields can move a charge between two sites of a molecule engineered to function as a two-dot QCA cell [25]. The significance of this experiment is that it shows a self-assembled monolayer (or plane) of these molecules switching between a chemical representation of a binary 0 and 1. Four-dot QCA molecules have also been made [16] and promising I/O methodologies exist. Also, active research in DNA-scaffolding can provide viable substrates for the molecules. These and other advances in QCA implementation propel research in QCA-based circuits and architecture to an important position.

The work discussed here presents a novel programmable structure for QCA-based circuits. The design principle revealed here could be useful for developing other QCA-based programmable devices. The PLA structure is essentially implementation *independent* despite the possibilities of multiple QCA implementations (e.g., metal-QCA [1], [19] and magnetic-QCA [6], [4] have also been proposed).

III. PLA STRUCTURE

A. Re-programmable PLA Cells

We use a combination of AND and OR planes made of QCA cells to implement a PLA. AND or OR gates can be implemented by using a single majority gate. Our challenge is to make the PLA cells programmable so that they may function either as a logic gate or as a wire. Each PLA cell, no matter which plane it is in, is made to contain a programmable select bit, denoted as S . In the AND plane, each PLA cell consists of two majority gates: one behaves as an AND gate, while the other as an OR gate to aid programming. A QCA cell schematic of the AND PLA cell is shown in Fig. 2(a) and its equivalent logic representation is given in Fig. 2(b). The OR PLA cell is constructed by simply switching the positions of the AND and OR gates in Fig. 2(a) and Fig. 2(b).

We will use the AND cell as an example to illustrate how a PLA cell can be programmed to function as either a logic gate or a wire. The AND cell has the following functionality.

$$\text{MintermOut} = (\text{LiteralIn}) \cdot (\text{MintermIn}), \text{ if } S = 0.$$

$$\text{MintermOut} = (\text{MintermIn}), \text{ if } S = 1.$$

We denote the mode of the PLA cell as “**logic**” if $S = 0$ and as “**wire**” otherwise. Thus during normal operation, the QCA cell corresponding to the select bit is used as an input to the OR gate. Initially, the mode of a PLA cell (i.e., the value of S) is undefined. To see how programming is achieved, it is interesting to note that a QCA-based majority gate is logically *bidirectional*, i.e., any one of the four boundary QCA cells can be treated as the output while the rest of the three as inputs.

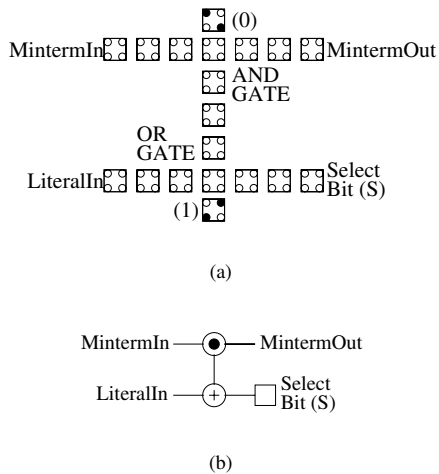


Fig. 2. (a) QCA representation of PLA cells in the AND plane. (b) Logic representation of PLA cells in the AND plane.

To program a cell, we treat the QCA cells corresponding to the select bit as the output of the OR gate. To drive a logical '1' or '0' to the select bit, we set MintermIn to '0' and LiteralIn to '1' or '0', respectively. The functionality and programming of OR cells are very similar except that '1' corresponds to **logic** mode and '0' corresponds to **wire** mode.

Readers may wonder how to differentiate QCA cells whose values need to be held (such as the QCA cells representing the select bit) from those whose values change when driver cells change their values. This can be achieved by judiciously controlling the clocking structure for the QCA cells. Recall from Section 2 that QCA cells can be in one of two states: a null state or a bistable, active state, depending on the specific electric field they are in. To allow a QCA-based logic gate to re-evaluate upon arrival of new input values, the electric field of the QCA cells associated with the logic gate needs to be altered (i.e., going from negative to positive). On the other hand, to hold the value of a set of QCA cells, simply supply a constant positive electric field. Therefore, to ensure that the PLA cell works properly, we must be able to individually control the electric field, i.e., having two separate clocking regions for the QCA cells of the select bit and for those of the logic gates. We will discuss this point further in Section 3.2.

B. The array structure

Based on the AND and OR cells introduced in the previous subsection, it is straightforward to construct a PLA. Fig. 3 depicts a logic representation of a PLA. The T_{XX} 's represent the terminal QCA cells at the PLA boundary. During normal operation, the inputs come from below, the minterm signals move from left to right, and the sum-of-minterm signals move from top to bottom. By setting $T_{L1} = T_{L2} = T_{L3} = 1$, $T_{T5} = T_{T6} = 0$, $T_{B1} = X$, $T_{B2} = X'$, $T_{B3} = Y$, $T_{B4} = Y'$, and leaving the rest of the terminals unset, it is easy to verify that the PLA performs the following two logic functions: $T_{B5} = X \text{ XOR } Y$, and $T_{B6} = X' \text{ OR } Y'$.

We have made a conscious effort to keep the number of wire crossings in a PLA to a minimum. This is due to the fact that implementing a wire crossing as shown in Fig. 1(d)

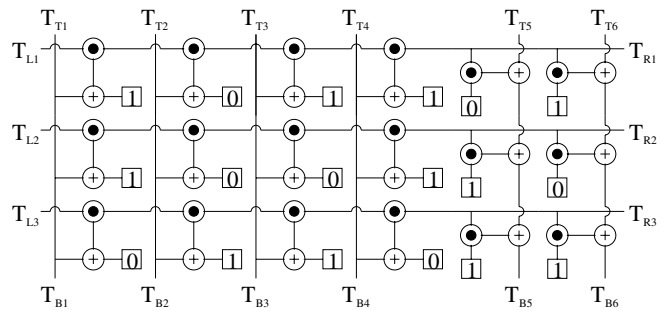


Fig. 3. A PLA array that implements a XOR and a OR function.

can be extremely challenging with near- to mid-term chemical synthesis processes envisioned by chemists [5]. One possible approach is to adopt the logic crossing idea proposed in [22]. The downside of employing the logic crossing idea is that it increases the number of QCA cells since as many as 12 majority gates are needed to achieve one logic crossing. Because our PLA structure only requires one crossing per cell, this increase is tolerable. (Additionally, our ongoing – and currently unpublished work – shows that a logical crossing could be mapped to a DNA substrate and would require less area than two metal wires would need to cross – at the 22 nm ITRS node [15]).

In order for a PLA array to operate properly, a carefully designed underlying clocking structure is required. For each PLA cell, two separate clocking regions are needed; one for the QCA cells representing the select bit and the other for the rest of the QCA cells. Let us consider the implications of this requirement. It is certainly possible to simply keep two clocking regions for every PLA cell. However, such a design may make the circuitry for generating the electric fields unnecessarily complicated. The reason is that each clocking region would need its own clock wire, which increases the demands on signal routing. Furthermore, since a PLA cell tends to be much smaller than the minimum metal wire in the end-of-the curve CMOS technology (if a crossing could be implemented with the structure shown in Fig. 1(d)), having a separate clocking region for the logic gates in each PLA cell would be somewhat wasteful. Individually controlled clocking regions would also increase the programming time and latency.

A much better approach is to treat the AND and OR planes as two large, alternate clocking regions. The entire AND plane, except the clocking regions for the select bits, is clocked by a larger clock wire. The same is true for the OR plane. The terminal QCA cells are grouped as follows: all T_{Li} 's in one group, all T_{Ri} 's in another group, all AND plane T_{Ti} 's and T_{Bi} 's in a group, and all OR plane T_{Ti} 's and T_{Bi} 's in a group. Such partitioning allows both efficient normal operation and programming. In normal operation, the clocking regions for the inputs (AND plane T_{Bi} 's) and outputs (OR plane T_{Bi} 's), together with those for the AND plane and OR plane can be driven by four 4-phase clock signals, each assuming a distinctive phase at any give time. We refer to this approach as the **global clocking** approach while the one in the previous paragraph as the local clocking approach. The

normal operation for the global clocking approach proceeds as follows. In the first quarter of a cycle, the inputs are updated. Then the minterms in the AND plane are evaluated during the second quarter of the cycle. In the third quarter, the OR plane cells are switched based on the results of the AND plane. Finally, in the last quarter, the outputs are made available.

The global clocking approach is fast and easy to implement. Furthermore, the structure immediately leads to a pipelined PLA and can be readily used to implement finite state machines without requiring any additional registers. However, there is a drawback to this clocking approach. The number of QCA cells that can be driven reliably by a single clocking region is finite and depends on the number of QCA cells along the longest path in the region as well as the clock frequency [20], [22]. Based on the current estimate, one clocking region can easily drive anywhere between 10^3 to 10^5 QCA cells along the longest path. We feel that this upper bound on the PLA size is sufficient for many realistic designs, especially considering the impact of defects. If a PLA array is too large for the global clocking approach, we can partition the PLA cells into smaller regions so that the number of QCA cells in each clocking region is within the acceptable range. How to partition the PLA cells can greatly impact the complexity of the clocking wire layout (as we discussed for the case of the local clocking approach—an extreme partitioning). Furthermore, different partitioning may require different clock signal patterns. We plan to investigate various clocking region partitions in our future work if the needs become evident.

C. Programming and defect detection

The ability to reprogram a PLA is very powerful. It makes the PLA more versatile and it also helps defect tolerance. Unlike mask-programmable CMOS PLAs which can be programmed once at fabrication time, our QCA-based PLA can be reprogrammed over and over again for different applications. Furthermore, by judiciously programming the PLA cells into **logic** or **wire** mode, an entire PLA can be tested for defects after fabrication. Defective columns, rows, and cells can be avoided during the programming stage, allowing PLAs with defective QCA cells to remain useful.

Since putting a cell into either **logic** or **wire** mode can be done by setting the select bit of the cell to '0' or '1' (see Section 3.1), programming a PLA can thus be achieved by driving the desired select bit values to corresponding PLA cells. The fact that a number of PLA cells, such as those on the same column of the AND plane, share the same input requires that a proper programming sequence be followed.

A simple and efficient programming sequence is to program row by row for the AND plane first and then column by column for the OR plane. To program one row, say row i of the AND plane, the desired select bit value for each PLA cell j on this row is applied to the corresponding terminal T_{Bj} (or equivalently T_{Tj}). Terminal T_{Li} is set to '0' to ensure that the programmed bit makes it to S unchanged. After a row is programmed, the electric field for the select bits on this row must be kept adequately positive such that these values

would not change even when different input signals appear at the OR gates connected to the select bits. A column can be programmed in a similar way except that T_{Ri} is used to supply the needed S values and T_{Tj} (or equivalently T_{Bj}) is set to '1'.

Defect detection is a multi-step process that requires analysis of both the QCA wires and the PLA cells. Testing the QCA wires is straightforward for the AND plane columns, as no logic gates interfere with signal propagation from bottom to top. We can simply apply a logic '0' and '1' to each T_{Bi} and test if the same signal appears at T_{Ti} . To test the AND plane rows, the OR plane columns and rows, the logical gates must first be placed into **wire** mode in order for the input signal to pass through. Once in **wire** mode, the same test as used for the AND plane rows can be performed. Testing the rows and columns is a quick process as it can be done in parallel.

In order to test the PLA cells, only a single AND plane column or OR plane row can be tested at a time. Each PLA cell in that column (resp. row) is placed into **logic** mode while the rest of the cells are all put into **wire** mode. Then, we set T_{Li} (resp. T_{Ti}) to '0' (resp. '1'), drive a '0' and '1' alternately to T_{Bj} (resp. T_{Rj}), and monitor the values at terminal T_{Rj} (resp. T_{Bj}) in the AND (resp. OR) plane. By repeating this process, all cells in the PLA can be tested. After the testing procedure is completed, methods for mapping a specific function to a PLA with known defects can be used to determine the mode in which each functioning PLA cell should be in [12]. We summarize the entire testing procedure of our QCA-based PLA circuit below.

1. Check AND plane columns
2. Program *all* cells to **wire** mode
3. Check all rows and OR plane columns
4. Repeat the following steps for all AND plane columns and OR plane rows
5. Program *all* cells to **wire** mode
6. Program a single AND (resp. OR) plane column (resp. row) of cells to **logic** mode
7. Check row (resp. column) outputs for defects in individual cells

Special care must be taken when programming *all* cells into **wire** mode (step 2 and 5 above) in order to avoid being overly pessimistic. Suppose a column, say column 2 in Fig. 3, is found to be defective after step 1 above. Then it seems that we may not be able to program some (or even all) of the PLA cells along column 2 since we could not drive a logic '1' from T_{B2} all the way up. To alleviate this problem, we again make use of the logic bidirectionality property of QCA cells. Note that our goal is to put as many PLA cells in **wire** mode as possible. With the original programming method discussed at the beginning of this section, the select bit in an AND plane cell can only be set to '1' by a terminal at the bottom of the AND plane. To increase the probability of select bits in the AND plane receiving a logic '1', we propose to set *all* the terminals of the PLA to logic '1'. By doing so, we have introduced two additional routes for a logical '1' to reach a select bit, i.e., from the terminal at the top of the PLA array and from the AND gate of the corresponding PLA cell. This greatly reduces

the impact of defective QCA cells and hence increases the probability of discovering operational QCA cells. The bottom line is that the yield of QCA-based PLAs could be increased. The detailed yield study will be presented in our future work.

IV. EXPERIMENTAL RESULTS

We have conducted several simulation-based experimental studies to validate the PLA design presented in the previous section. These experiments both verify the functionality of our PLA design through simulations based on a statistical mechanics model, and examine the electric field distribution resulting from the global clocking approach.

Based on the analytical model provided by [34], we have developed a statistical mechanics simulation tool for QCA circuits built with 4-dot QCA cells (Fig. 1(a)). This tool takes a pattern of QCA cells and determines the electron potential energy for all states. It then reports the lowest twenty energy states, including the global “ground” (i.e., the lowest energy) state. If the electron distribution in the global ground state matches the desired behavior of the QCA circuit, we can be confident that the design will work. However, if it does not match the global energy minimum, but is still one of the lower energy states, it may be a local minimum, and other simulation can determine if the design will still work in practice.

In order to verify the operation of our PLA design, we have tested our PLA cells using the statistical mechanical model. These tests considered the full truth table of possible inputs to the cells. Through careful layout and extension of the majority-gate intersections, we have found a PLA cell design that produces a global energy minimum for the desired behavior in each truth table combination, indicating that our PLA should function as intended.

Clocking region design plays a key role in ensuring the correct operation of the PLA. In our global clocking approach, one unique requirement is that the regions corresponding to the select bits must be able to supply a constant positive electric field even when the other regions’ electric fields are changing. Hennessy and Lent have studied electric fields introduced by evenly spaced point charges [11]. Their main results demonstrated that desirable electric field distributions for a pipelined QCA circuit can be obtained by applying proper charges on infinitesimally small wires underneath the QCA circuit. The results should also apply to wires with finite sizes. However, to our best knowledge, no experimental results have been published regarding this, and there is no experimental study on the possibility of supplying the more demanding electric field distribution for our proposed PLA structure.

To validate that our global clocking approach indeed provides the desired electric field, we have conducted a number of experiments to investigate the electric field distribution from metal wires (such as those used in CMOS technology). Fig. 4(b) shows a sample layout of clocking regions resembling those that may be used for one plane of a PLA. The small square shapes (referred to as *select wires*) represent the clocking wires that supply the electric field for the QCA cells of the select bits. The long rectangular shapes (referred

to as *logic wires*) correspond to the wires that supply the electric field for the columns/rows of the PLA logic cells. We have used different sizes for the wires as well as different separations between the wires. The minimum metal pitch is 40nm, in accordance with the 2004 ITRS for 2018 technology node [15]. A metal plane is added 20nm above the logic and select wire plane to act as the ground plane for directing electric fields [11]. It is envisioned that the QCA-based PLA would be sandwiched between the two metal planes. Such a structure has been demonstrated in [25]. We used silicon dioxide as the dielectric with relative permittivity of 3.9 and gold as the metal for the wires. (We note that only 2 layers of metal should be needed to build the clock structure.)

To study the electric field distribution, two distinct clocking signals are applied. All the select wires are driven by a constant positive voltage while the logic wires are driven by a quasi-adiabatic 4-phase clock signal [31]. This setup emulates the PLA in normal operation mode. (Note that we did not study the programming scenario as it is similar to the operations of pipelined circuits which has been studied in [11].) FEMLAB software [9] is used to obtain the electric field distribution at the plane where QCA cells reside. Since only the component of the electric field that is perpendicular affects the states of the QCA cells [11], we only examine the distribution of the vertical (i.e., the z-direction) electric field. We applied a 5 volt signal to select wires, and applied five different voltages, -5 volts, -2.5 volts, 0 volt, 2.5 volts, and 5 volts, to the quasiadiabatic signal (logic wires). The selection of 5 volts as the maximum voltage is based on the assumption that the intensity of 2 Mv/cm is sufficient to maintain the states of QCA cells. Based on conversations with chemists, this is a realistic assumption for molecular QCA cells.

The simulation results are summarized in Fig. 4(a)–Fig. 4(f). These figures depict the intensity of the vertical electric field 10nm above the logic/select wire plane. According to the intensity scale bar in Fig. 4(a), one can see that the vertical electric field above the logic wires changes from -2.5Mv/cm to 2.5Mv/cm while the vertical electric field above the select wires is unchanged. The field intensity above the select wires exceeds 2Mv/cm, which indicates that the QCA cells on top of these wires are in the active/bistable state [11]. At this state, the QCA cells cannot receive any inputs and maintain their previous logic values. This is precisely what we need in order to guarantee the correct functionality of our PLA structure. The field distributions remain essentially the same regardless of the sizes of the wires and distances between them.

V. CONCLUSIONS AND FUTURE WORK

We have presented a novel QCA-based PLA structure and showed how to detect defects and program around them by exploiting the bi-directionality property of QCA cells. We also demonstrated that our PLA design is implementation friendly in two regards. First, the clocking regions are regular and the electric field distribution is relatively simple. Therefore, the circuitry to provide the electric field can be designed and fabricated with traditional CMOS technology. This has been

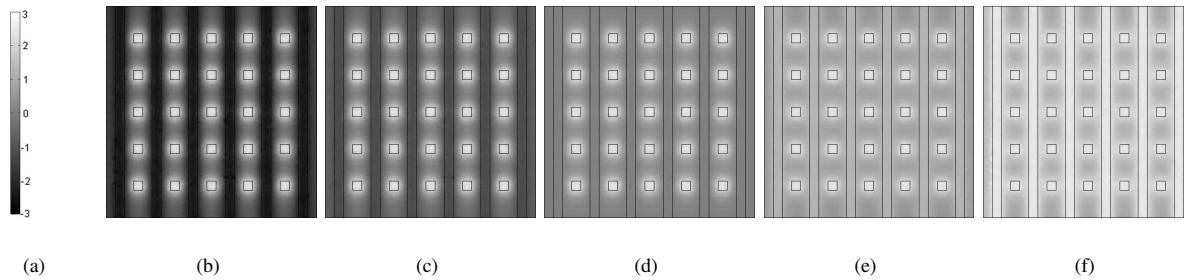


Fig. 4. Vertical electric field distribution at the plane of QCA cells. (a) shows the intensity scale bar from -3 Mv/cm to 3 Mv/cm, and (b)-(f) correspond to the five different voltage values on the logic (i.e., long) wires, -5 volts, -2.5 volts, 0 volt, 2.5 volts, and 5 volts, respectively.

validated via detailed electric field simulation. Second, the area of each PLA cell is small and requires only a single wire crossing. Wire crossings can be a major road block in the near- to mid-term molecular QCA fabrication. By having a single wire crossing, we achieve a minimal PLA cell size.

Our proposed PLA structure laid down some ground work for designing re-programmable QCA circuits. More detailed investigation for QCA-based PLA is still needed. For example, the actual layout of a PLA needs to be conducted to examine the impact of PLA cell size on its functionality. This should be studied together with the clocking wire layout. Another important aspect is to study the yield based on realistic QCA defect models. Designing reconfigurable/reprogrammable QCA-based circuits is still at its infancy. We expect that our work will initiate a much needed effort in this area.

REFERENCES

- [1] Amlani I., Orlov A. O., Snider G., and Lent C.S., "Demonstration of a functional quantum-dot cellular automata cell," *J. Vac. Sci. Technol. B*, 16, 1998, p. 3795-3799.
- [2] S. Balensiefer, L. Kregor-Stickles, and M. Oskin. "An Evaluation Framework and Instruction Set Architecture for Ion-Trap based Quantum Micro-architectures," in the *Int. Symp. on Comp. Arch.*, 2005.
- [3] P.F. Barbara, T.J. Meyer, and M.A. Ratner, "Contemporary Issues in Electron Transfer Research," *J. Phys. Chem.*, 1996, 100, p. 13148-13168.
- [4] G.H. Bernstein, A. Imre, V. Metlushko, A. Orlov, L. Zhou, L. Ji, G. Csaba, and W. Porod, "Magnetic QCA systems," *Microelectronics Journal*, 36(2005), p. 619-624.
- [5] A. Chaudhary, D.Z. Chen, X. Sharon Hu, M.T. Niemier, R. Ravichandran, and K. Whitton, "Eliminating Wire Crossings for Molecular Quantum-dot Cellular Automata Implementation," in *Proc. of Int. Conf. on Comp. Aided Des.*, p. 565-571, Nov. 6-10, 2005.
- [6] R.P. Cowburn and M.E. Welland, "Room Temperature Magnetic Quantum Cellular Automata," *Science*, Vol. 287, Issue 5457, 1466-1468, 2000.
- [7] A. DeHon, "Array-based Architectures for FET-based Nano-scale Electronics," *IEEE Nano.*, Vol. 2, No. 1 pp 23-32, March 2003.
- [8] A. DeHon and M.J. Wilson, "Nano-wire based sublithographic Programmable Logic Arrays," published in *Proc. of Int. Symp. on FPGAs*, (February 22-24, 2004), pp. 123-132.
- [9] FEMLAB 3.0 software by COMSOL, Inc., <http://www.comsol.com>.
- [10] C.R. Graunke, D.I. Wheeler, D. Tougaw, and J.D. Will, "Implementation of a Crossbar Network Using Quantum-dot Cellular Automata," *IEEE Trans. on Nano.*, Vol. 4, No. 4, July 2004.
- [11] K. Hennessy, C.S. Lent, "Clocking of molecular quantum-dot cellular automata," *J. of Vac. Sci. & Tech. B*, 19, 5(Sep-Oct 2001), 1752-55.
- [12] T. Hogg and G. Snider, "Defect-tolerant logic with Nanoscale crossbar circuits," HP Labs, <http://www.hpl.hp.com/research/idl/papers/molecularAdder/>.
- [13] W. Hu, K. Sarveswaran, M. Lieberman, and G. H. Bernstein, "High resolution electron beam lithography and DNA nano-patterning for molecular QCA," *IEEE Trans. Nano.*, 2005, 4, 312-316.
- [14] J. Huang, M. Momenzadeh, M.B. Tahoori, F. Lombardi: "Design and characterization of an and-or-inverter (AOI) gate for QCA implementation," *ACM GLVLSI Symp. 2004*, 426-429.
- [15] International Technology Roadmap for Semiconductors, 2004, <http://www.itrs.net/Common/2004Update/2004Update.htm>.
- [16] Jiao J.Y., Long G.J., Grandjean F, Beatty A.M., Fehlner T.P. "Building blocks for the molecular expression of quantum cellular automata," *J. Am. Chem. Soc.*, 2003, 125, 7522-7523.
- [17] T.H. Labean, S. Park, S.J. Ahn, and J.H. Reif, "Stepwise DNA self-assembly of fixed-size nanostructures," *Found. of Nanoscience, Self-assembled Arch., and Dev.*, pp. 179-181, 2005.
- [18] J.D. Le, Y. Pinto, N.C. Seeman, K. Musier-Forsyth, T.A. Taton, and R.A. Kiehl, "DNA-Templated Self-Assembly of Metallic Nanocomponent Arrays on a Surface," *Nano Lett.*, 4 (12), 2343-2347, 2004.
- [19] Lent C.S., Snider G.L., Bernstein G., Porod W., Orlov A., Lieberman M., Fehlner T., Niemier T., and Kogge P. *Quantum-Dot Cellular Automata*. Chapter in Electron Transport in Quantum Dots, pp. 397-433, Johathan P. Bird (ed.) Kluwer Academic Publishers, 2003.
- [20] C.S. Lent, P.D. Tougaw, "A Device Architecture for Computing with Quantum Dots", *Proc. of the IEEE*, Vol. 85, p. 541, 1997.
- [21] M. Lieven, K. Vandersypen, M. Steffen, G. Breyta, C.S. Yannoni, M.H. Sherwood, and I.L. Chuang, "Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance", *Nature*, 414, 883887 (20 December 2001).
- [22] M.T. Niemier, "The Effects of a New Technology on the Design, Organization, and Architectures of Computing Systems," *Thesis.*, 2004.
- [23] Niemier, M.T. and Kogge, P.M. "The '4-Diamond' Circuit - A Minimally Complex Nano-scale Computational Building Block in QCA," In *Proc. of the IEEE Comp. Soc. Symp. on VLSI*, pp. 3-10, February 2004.
- [24] M. Niemier and P. Kogge, "Exploring & exploiting wire-level pipelining in emerging technologies", *Int. Sym. of Comp. Arch.*, 2001, p.166-177.
- [25] Qi H., Sharma S., Li Z.H., Snider G.L., Orlov A.O., Lent C.S., Fehlner T.P., "Molecular quantum cellular automata cells," *J. Am. Chem. Soc.* 2003, 125, 15250-15259.
- [26] P.W.K. Rothmund, N. Papadakis, and E. Winfree, Algorithmic self-assembly of DNA Sierpinski triangles, *PLoS Biology*, 2004, 2041-2053.
- [27] D.B. Strukov and K.K. Likharev, "CMOL FPGA: a reconfigurable architecture for hybrid digital circuits with two-terminal nanodevices," *Nanotechnology*, 16(2005) 888-900.
- [28] M.B. Tahoori, J. Huang, M. Momenzadeh and F. Lombardi, "Defects and Faults in QCA at the nano-scale", *VLSI Test Sym.*, 2004.
- [29] J. Timler and C.S. Lent, "Power gain and dissipation in quantum-dot cellular automata", *J. of App. Phys.*, 91, 2002, p.823-831.
- [30] G.M. Tom and H. Taube, "The Mixed Valence State Based on μ -Cyanogen-bis(pentaammineruthenium)," *J. of Am. Chem. Soc.*, Vol. 97 p.5310-5311, 1975.
- [31] G. Toth and C. S. Lent, "Quasiadiabatic switching for metal-island quantum-dot cellular automata", *J. Appl. Phys.*, 85, 2977, 1999.
- [32] P.D. Tougaw, C.S. Lent, "Logical Devices Implemented Using Quantum Cellular Automata", *J. of App. Phys.*, Vol. 75, 1994, p. 1818.
- [33] K. Walus, T. Dysart, G.A. Jullien and R.A. Budiman, "QCA Designer: A Rapid Design and Simulation Tool for Quantum-dot Cellular Automata", *IEEE Trans. on Nano.*, 3(1), 2004, pp. 26-31.
- [34] Yuliang Wang and Marya Lieberman, "Thermodynamic behavior of molecular-scale quantum-dot automata (QCA) wires and logic devices," *IEEE Trans. on Nano.*, 2003.
- [35] D. Whang, S. Jin, Y. Wu, and C.M. Lieber, "Large-scale hierarchical organization of nanowire arrays for integrated nanosystems," *Nanoletters*, Vol. 3, No. 9, pp. 1255-1259, September, 2003.
- [36] C.P. Williams and S.H. Clearwater, *Explorations in Quantum Computing*, Springer-Verlang New York, Inc., New York, NY, 1997.