Logic Devices for Partitioned Quantum-Dot Cells

Daniel P. Kelly

Department of Computer Engineering and Computer Science, University of Missouri-Columbia, Columbia, Missouri 65211

John C. Lusth

Department of Computer Science, Boise State University, Boise, Idaho 83725

Abstract

Quantum-dot cellular automata are a method of computation using a small number of electrons. This paper explores the use of quantum-dot cellular automata as logic devices and proposes a way to increase the reliability of these devices. By manipulating their architecture, the energy difference between the ground state and the first excited state can be increased. A larger difference allows for higher operating temperatures at the same reliability level or increased reliability at lower temperatures. The architectural changes explored in this paper include limiting the number of quantum dots accessible by an electron within a cell and the addition of a secondary layer of quantum-dot cells.

Keywords: computer architecture, quantum-dot cellular automata, QCA

1. Introduction

The computer age has developed with one goal at the forefront: make computers run faster and more efficiently in less space. Quantum-dot cellular automata (QCA) are one of the current leading approaches on how to achieve this goal.

The idea of using quantum-dots for computing was first described in detail by Tougaw and Lent [1]. Logic devices such as AND, OR, and NOT can be constructed by grouping quantum-dot cells in certain configurations [2]. These primitive devices can then be combined hierarchically to form a QCA circuit.

In general, QCA circuits compute via energy relaxation. The input to the circuit is represented as a bit-string and a set of cells, denoted *input* cells, and is fixed to specific polarizations representing the bit-string. The device then is allowed to relax into the lowest energy, or *ground*, state. The polarizations of another set of cells, denoted the *output* cells, reflect the bit-string representation of the output.

If the temperature of an automaton is too high, its electrons can relocate from the ground state configuration (where it is assumed that the polarizations of the output cells are correct) to an excited state (where it is assumed the output cell polarizations are incorrect). By increasing the energy difference (herein Δ) between the ground state and the first excited state, an automaton can be made more reliable at a given temperature or made to operate with the same reliability at a higher temperature.

Lusth, Hanna, and Díaz-Vélez [3] have proposed two methods for increasing Δ in a two-cell automaton for computing IDENTITY: partitioning the cells and adding an additional layer of cells. This paper investigates applying these methods to increase the reliability of automata more complex than the two-cell IDENTITY automaton, namely the primitive QCA logic devices.

2. Background

The basic unit of QCA is the quantum cell. A standard quantum cell consists of four quantum dots positioned at the vertices of a square. Each cell also contains two electrons. These electrons can quantum-mechanically "tunnel" from one quantum dot to another, but they cannot escape the cell. This tunneling allows an automaton to relax to its ground state, a state of minimum energy. As shown in

Figure 1, a cell has two preferred states where electron separation is maximized, suggesting an encoding scheme for binary information [1,4].

Figure 1. The schematic of a cell, with the two bi-stable states. The circles represent quantum dots. The circles filled in with black contain electrons. By convention, the left cell represents a logical zero, whereas the right cell represents a logical one.

2.1 Logical vs. non-logical states

There are other possible configurations of electrons within a cell, but these states are considered *non-logical* [3], since only the two antipodal configurations shown in Figure 1 have logical interpretations. Figure 2 illustrates the more important non-logical states.

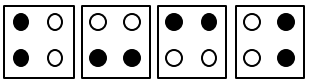

Figure 2. Non-logical cells. None of these four cells represent a binary value.

Other non-logical states, not shown, exhibit the simultaneous occupation of a dot by two electrons. Such states are assumed to be prohibitively expensive, energy-wise, and therefore, not considered in this paper.

Often, the energies of configurations with non-logical cells lie between the energies of the desired ground state and excited states with purely logical cells. By prohibiting or promoting the non-logical states in these cases, Δ increases since a higher excited state becomes the first excited state. With increasing Δ, the automaton becomes more reliable, as can be seen from the *viability relation* [5], $\left[\frac{kT}{\Delta}\right]\ln(n) << 1$ where k is Boltzman's constant, T is temperature, and n is the number of cells in the automaton. If the viability relation holds, the automaton will compute correctly. The smaller the left-hand side of the relation, the less likely an electron will be found in the "wrong" quantum dot.

2.2 QCA geometry

The *standard* geometry for a QCA is a nearest neighbor dot spacing of 20 nm and a center-to-center cell spacing of 60 nm [6]. As detailed in [3], under the standard geometry, the cells of both the ground state and first excited state of an automaton for computing IDENTITY are logical. By increasing the dot spacing to 40 nm (while keeping the cell-spacing constant), the two logical states spread apart, with non-logical states falling between the two logical configurations. By prohibiting some of these non-logical states and promoting the others to higher energies, the logical, but incorrect, state becomes the first excited state and thus Δ is increased. A summary of how the non-logical states can be removed from consideration is found in the next section. The dimensions under which these effects can be seen (a dot-spacing of 40 nm and a cell-spacing of 60 nm) will herein be referred to as the *alternate* dimensions.

2.3 Eliminating non-logical states

Partitioning the cells into two equal sections can eliminate some non-logical states and will increase Δ if the eliminated states lie between the logical states. A QCA cell can either be partitioned horizontally or vertically, as shown in Figure 3.

Figure 3. Partitioned cells. A horizontally partitioned cell on the left and a vertically partitioned cell on the right.

It is assumed that an electron cannot escape its partition of the cell; therefore, it only has two choices of location.

For example, partitioning the cell horizontally prohibits the second and third

configurations in Figure 2. However, a cell can only be partitioned one way, eliminating only two of the four non-logical states. Placing a second layer, especially with opposite partition geometry, directly on top (or below), can promote the remaining non-logical states to higher energies. The layered and partitioned system for simple IDENTITY yields a 20-fold improvement in Δ over a single layered system using the standard dimensions [3].

2.4 QCA Logic Devices

A line of QCA cells, denoted a binary wire, can be considered a logical device which computes the IDENTITY function. In its ground state, every cell in a line of cells has the same polarization as the first, or input, cell, as shown in Figure 4.

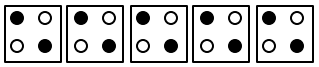

Figure 4. A line of quantum-dot cells or binary wire.

Therefore, the last, or output, cell will have the same polarization as the input cell. Purely linear wires have been studied in the context of increasing Δ in [7], but there are two useful, non-linear, forms of wires. They are a "Fan-Out", which allows the propagation of a value to two or more other points within the array, as shown in Figure 5, and a "Corner in a Wire", which changes the direction of the line of cells, as shown in Figure 6.

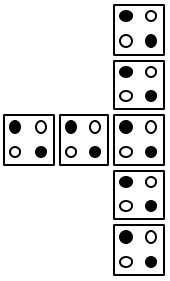

Figure 5. A Fan-Out. Both output cells on the top and bottom have the same polarization as the input cell on the left.

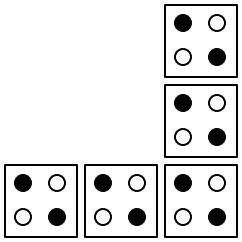

Figure 6. A Corner in a Wire. The output cell on top has the same polarization as the input cell on the left.

Arrangements of cells can also be used as logic devices for Boolean Algebraic calculations [2]. The most common operations used in the Boolean algebra are AND, OR, and NOT gates. NOT gates, or inverters, are realized as QCA as shown in Figure 7.

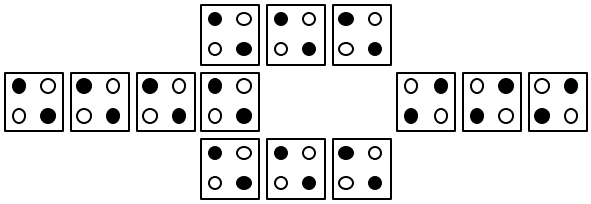

Figure 7. A QCA Inverter. A signal comes in from the left, splits into two wires, and is inverted at the point of convergence.

An AND gate is shown in Figure 8. The top cell is fixed (as indicated by its thicker border) in the "0" position. Only when inputs A and B are polarized in the "1" position, does the device cell and the output cell take on the "1" orientation. An OR gate is constructed by fixing the top cell to a "1" position.

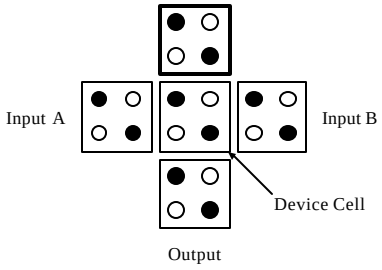

Figure 8. An AND gate. The polarizations of the device and output cells are always the same as the majority of the inputs and the fixed cell (indicated by the thicker boarder).

2.5 Modeling QCA

Calculating the ground states of QCA, for the purposes of this paper, will be computed classically, as suggested by Akazawa, Amemiya, and Shibata [8]. The classical computation of the energy of point charges is obtained by summing over all pairs of charges: $E = \sum_{allpairs} \frac{q_1 q_2}{4pe \cdot r}$, where q_1 and q_2 are the charges in the pair, ε is the permittivity of the space between them, and r is the distance between them. For the calculations in this paper, ε is taken to be permittivity of free space. Compensating charges, for maintaining cell neutrality, are handled in the idealized fashion found in [6]. This approach assumes an effective positive charge with magnitude ½ *e* is placed on each dot of a cell. Thus an occupied dot is considered to have a charge of – ½, while and unoccupied dot has a charge of + ½.

3. Improved Logic Devices

The use of partitioning and layering dramatically increases Δ for the QCA primitive logic devices. Table 1 summarizes the improvements in Δ that can be achieved by partitioning and layering. Each device is examined in detail in the following subsections.

3.1 Improved Corner Turns and Fan-Outs

To achieve gains in Δ similar to the 20-fold gains for simple IDENTITY, the cells in a single layer binary wire must be partitioned in the direction of information flow. A similar style of partition must be followed for corner turns and fan-outs. For example, an un-partitioned, single-layer, three-cell corner turn with standard dimensions has a Δ of 1.481 meV. The alternate dimensions yield a nearly six-fold increase in Δ. If the input and output wires are partitioned in their direction of information flow and the cell in the absolute corner is partitioned in the same way as the input wire, a nearly eleven-fold increase in Δ results (Figure 9). The addition of a secondary layer partitioned perpendicular to the primary yields approximately a nineteen-fold increase. For exact numbers, refer to Table 1.

	D (meV)				
Device	*Original Dimensions*	*Alternate Dimensions*	*Partitioned*	*Partitioned and Layered*	*Max Percent Improvement*
Corner Turn	1.481	8.330	15.56	27.83	1779
Fan-Out	1.087	7.983	16.02	21.00	1831
NOT	0.751	9.531	15.46	20.82	2672
AND	1.143	4.550	16.96	21.21	1755

Table 1. Data for Partitioned Logic Devices in sections 3.1 through 3.3.

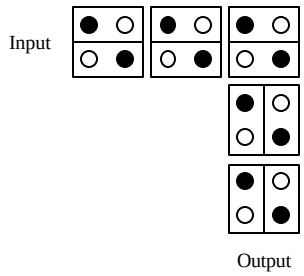

Figure 9. Partitioned corner in a wire.

A fan-out works in very much the same way as a corner, except there are two output wires instead of one. Δ is maximized when the cell at the vertex is partitioned along with the majority of the adjoining wire, as shown in Figure 10.

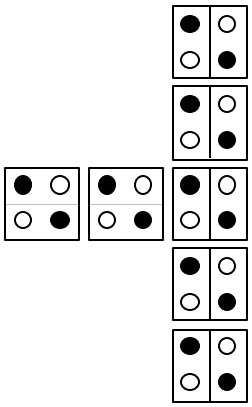

Figure 10. A partitioned fan-out.

For example, an un-partitioned, single layer four cell fan-out with standard dimensions has a Δ of 1.087 meV. The alternate dimensions yield a seven-fold increase in Δ. Partitioning the fan-out as described above yields a fifteen-fold increase in Δ. The addition of a secondary layer partitioned perpendicular to the primary yields a nineteen-fold increase.

3.2 Improved AND Gates

An AND gate, under the standard dimensions, has a Δ of 1.143 meV. Changing to the alternate dimensions yields a four-fold increase in Δ. Partitioning the device and output cells vertically, while the left and right input cells are partitioned horizontally (Figure 11), yields a fifteen-fold increase.

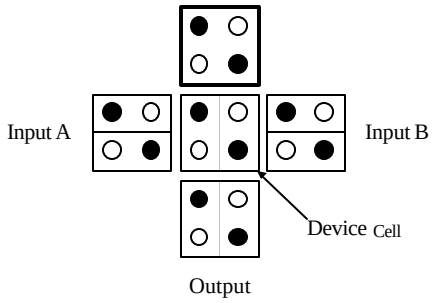

Figure 11. A partitioned AND Gate. The top cell (with the thicker border) is fixed and thus does not need to be partitioned.

To further increase the Δ, a second layer inversely partitioned to the primary layer could be attached to the majority gate. If all the cells on the primary layer are partitioned as described above, the addition of a secondary layer partitioned perpendicular to the primary yields a nineteen-fold increase in Δ.

3.3 Improved Inverter

An inverter works very much in the same way as a binary wire. All of the cells must be partitioned in the direction of information flow, except for two. For example, a single layer inverter with the standard dimensions has a Δ of 0.751 meV. Changing to the alternate dimensions yields a thirteen-fold increase in Δ. If all of the cells are partitioned in the direction of the information flow, except for cells A and B, as shown in Figure 12, there is a twenty-one-fold increase in Δ. With the addition of a second layer partitioned perpendicular to the primary, the result is a twenty-eight-fold increase.

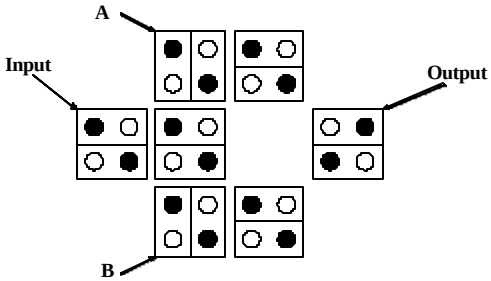

Figure 12. A simplified version of a partitioned inverter. Notice cells A & B are partitioned perpendicular to the direction of the information flow.

4. Conclusion

By partitioning the cells and the addition of a second layer, the reliability of most of the primitive QCA logic devices can be greatly improved.

- The partitioning of an AND gate yielded a 14.8-fold increase in Δ. The addition of a secondary layer yielded an 18.6–fold increase.
- The partitioning of an inverter yielded a 20.6-fold increase in Δ. The addition of a secondary layer yielded a 27.7–fold increase.
- The partitioning of a corner turn yielded a 10.5-fold increase in Δ. The addition of a secondary layer yielded an 18.8–fold increase.
- The partitioning of a fan-out yielded a 14.7-fold increase in Δ. The addition of a secondary layer yielded a 19.3–fold increase.

By the viability relation, the highest operating temperature increases with an increasing Δ. Even though *n* doubles when a secondary layer is added to the system, the fact that the viability relation factors in the natural logarithm of *n* greatly reduces the affect of the number of cells on viability. With the improvements in Δ gained via the alternate dimensions and judicious removal of non-logical states, QCA have become a much more practical approach to create a faster, denser circuit.

5. References

[1] C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernstein, "Quantum cellular automata", *Nanotechnology*, vol. 4 (1), pp. 49-57, January 1993

[2] P. D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata", *Journal of Applied Physics*, 75, 1818-1825, February 1994

[3] J. C. Lusth, C. B. Hanna, and J. C. Díaz-Vélez, "Eliminating non-logical states from linear quantum-dot-cellular automata", *The Microelectronics Journal*, to appear

[4] G. L. Snider, A. O. Orlov, I. Amlani, X. Zou, G. H. Berstein, C. S. Lent, J. L. Merz, and W. Porod, "Quantum-dot cellular automata: Review and recent experiments", *Journal of Applied Physics*, 1999;85:4283-4285 (invited paper)

[5] C. S. Lent and P. D. Tougaw, "A Device Architecture for Computing with Quantum Dots", *Proceedings of the IEEE*, vol. 85 (4), pp. 541-557, April 1997

[6] C. S. Lent and P. D. Tougaw, "Lines of interacting quantum-dot-cells: a binary wire", *Journal of Applied Physics*, 74, 6227-6233, November 1993

[7] J. C. Lusth and W. Knowlton, "Quantum-dot cellular automata and an approach to improving their performance", to be submitted to *IEEE Transactions on Computers*

[8] M. Akazawa, Y. Amemiya, and N. Shibatta, "Annealing method for operating quantum-cellular automaton systems", *Journal of Applied Physics*, vol. 82, 5176-5184, 1997