DESIGN AND SIMULATION OF FAULT-TOLERANT QUANTUM-DOT CELLULAR AUTOMATA (QCA) NOT GATES

A Thesis by

Mary Jean Beard

Bachelor of Science, California Institute of Technology, 2003

Submitted to the Department of Electrical and Computer Engineering
and the faculty of the Graduate School of
Wichita State University in partial fulfillment of
the requirements for the degree of
Master of Science

July 2006

DESIGN AND SIMULATION OF FAULT-TOLERANT QUANTUM-DOT CELLULAR AUTOMATA (QCA) NOT GATES

I have examined the final copy of this Thesis for form and content and recommend that it be accepted in partial fulfillment of the requirement for the degree of Master of Science, with a major in Electrical Engineering.

_____
Fred J. Meyer, Committee Chair

We have read this thesis
and recommend its acceptance:

_____
Coskun Cetinkaya, Committee Member

_____
Elizabeth Behrman, Committee Member

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Fred Meyer, for his guidance and support throughout my studies at Wichita State University. I would also like to thank the members of my committee, Dr. Coskun Cetinkaya and Dr. Elizabeth Behrman, for their time, effort, and support of this project.

I would not have been able to complete my studies without the love and understanding of my fiancé Garrett. I would like to thank my parents, Charles and Ann, and my sisters, Jennifer and Stephanie, for their help and ideas. I can't even begin to thank all of my friends who have provided their own insight and ideas towards my work.

ABSTRACT

This paper details the design and simulation of a fault-tolerant Quantum-dot Cellular Automata (QCA) NOT gate. A version of the standard NOT gate can be constructed to take advantage to the ability to easily integrate redundant structures into a QCA design. The fault-tolerant characteristics of this inverter are analyzed with QCADesigner v2.0.3 (Windows version) simulation software. These characteristics are then compared with the characteristics of two other non-redundant styles of NOT gates. The redundant version of the gate is more robust than the standard style for the inverter. However, another simple inverter style seems to be even more than this fault-tolerant design. Both versions of the gate will need to be studied further in the future to determine which design is most practical.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Since the development of the integrated circuit, the microelectronics industry has constantly worked to achieve smaller feature size and higher computational density on electronic devices. This shrinkage of chip size allows for higher speeds within the devices and smaller electronic products that are much more powerful than their predecessors.

Today, many PC processors are fabricated using a 90nm process, with some processors being fabricated at the 65nm level. The development of smaller-scale processors is hindered by many barriers. The lithographic techniques used to create the chips begin to break down at small scales because of the inability to focus light in such tiny detailed patterns. The doping process becomes irregular due to the statistical improbabilities of maintaining constant doping levels at such small concentrations. Once the devices have been constructed problems still exist such as electron tunneling and thermal noise.

Currently research is being conducted into alternative computational paradigms that would allow for devices to be constructed at even smaller scales without the limitations faced by the traditional CMOS-based architectures. One promising new solution is Quantum-dot Cellular Automata (QCA).

This thesis is concerned with the design and simulation of fault-tolerant QCA NOT gates. Chapter 2 details the basics of QCA and QCA design. Chapter 3 discusses the design of a fault-tolerant NOT Gate and describes the methodology used in the research for this document. The robustness of three different QCA inverters is tested in Chapter 4. Chapter 5 is a discussion of the results and the conclusions reached during this research.

CHAPTER 2

BACKGROUND & LITERATURE REVIEW

2.1 QCA Basic Concepts

QCA is not quantum computing in its strictest sense, but rather a quantum implementation of classical computing. QCA is being explored as a method for stretching the limits of classical computation, and enabling a proven method of computation to be improved at the physical limitations of that method.

2.1.1 Quantum Cells

The fundamental unit of a QCA circuit is the quantum cell, as shown below in Figure 2-1. Quantum cells typically contain four quantum dots, placed near the corners of the cell, where free electrons can reside. The QCA cells have two free electrons and two fixed protons to maintain cell charge neutrality. The electrons are free to tunnel among the quantum dots; however, a high inter-cell potential barrier ensures that the electrons do not tunnel between cells.

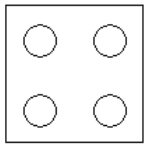

Figure 2-1: Quantum Cell

Quantum cells have two distinct stable polarizations, as shown in Figure 2-2. These states allow the cells to represent binary data. By convention, a cell with +1 polarization is used to represent a Binary 1, while a cell with -1 polarization represents a Binary 0.

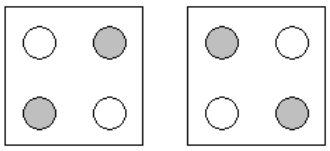

Figure 2-2: Quantum Cells with Polarity +1 (left), Polarity -1 (right)

2.1.2 Binary Wires

Binary wires are the simplest QCA structures and consist of a series of quantum cells in close proximity to each other. The cells interact through Coulombic interactions with each other. In this manner, if the leftmost cell has its polarity fixed at +1, and the other cells have no other force influencing their polarities, all cells in the wire will take on the same polarization. It is important to note that no current flows during this process. Unlike CMOS technology, QCA relies on quantum configurations rather than voltage levels to indicate state, and Coulombic interactions rather than current flow for information propagation.

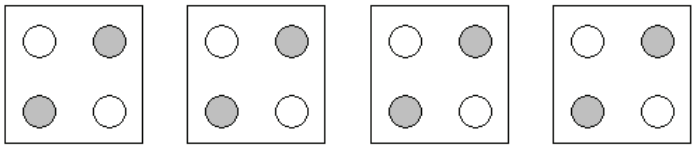

Figure 2-3: QCA Binary Wire

Binary wires can also be constructed with the dots in each cell oriented at a 45 degree angle from the standard cell. This allows binary wires to cross in the same plane or layer without interacting with each other. This configuration causes an inversion to occur between each cell. Care must be taken to use the correct number of cells to prevent accidental signal inversion from occurring.

2.1.3 Logic Gates

There are two logic gates that make up the fundamental set of logic in QCA: the Majority Voter and Inverter. All other logical functions can be computed from some combination of these two gates.

The Majority Voter takes three inputs and outputs the value that occurs most frequently. The output of this gate is defined by MV(A,B,C) = AB + AC + BC. The Majority Voter can also be used to create AND and OR gates. If one input is held at 1, the Majority Voter functions as a standard 2-input OR gate. If one input is held at 0, the Majority Voter functions as a 2-input AND gate. Figure 2-3 depicts the standard Majority Voter gate.

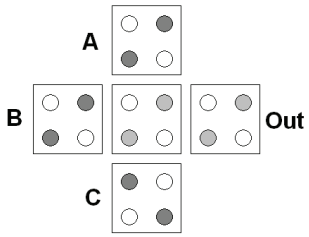

Figure 2-4: QCA Majority Voter Gate

The NOT gate has a single input and output. It simply returns the opposite of the value that was put in.

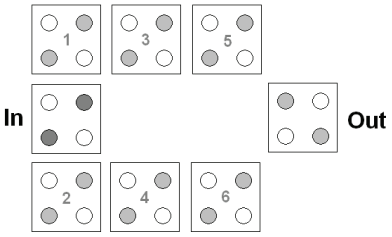

Figure 2-5: QCA NOT Gate

These two logic gates, together with a hard-wired 1 or 0 signal can be used to create all logic functions used in traditional computation.

2.1.4 Clocking

QCA circuits require a multi-stage clocking mechanism. This clocking provides two major functions: synchronization of signals and restoration of energy levels within the system.

In QCA a four-phase clocking mechanism is used. The sequence of states in this scheme is null state, switching state, locked state, and switching state. In the null state the quantum cells contain no binary data. In the switching state, the cells are free to undergo Coulombic interactions with their neighboring cells and take on the appropriate binary value. Sometimes the two switching states are differentiated as a 'switching' state and a 'release' state. In the first case, the cells are changing between a low entropy state to a high entropy state. In the other, the process is reversed. In the locked state, the electrons are prevented from tunneling within the quantum cell, so the cell remains in a fixed state.

The clock zones can be generated by a series of wires that run parallel to the plane on which the QCA cells lie. These wires can be used to generate electric fields that restrict the tunneling of electrons within the cells, creating the null and switched states. Removal of the electric field would allow the QCA devices to enter into one of the switching states.

The use of the clock as an energy restoring feature is vital to the success of any QCA circuits of useful size. Each cell dissipates energy as it interacts with neighboring cells. The clock provides a method to restore each cell to its original energy level. In a binary wire with a cell size of 18nm and intercellular spacing of 2nm, signal degradation may be observed after 5 cells.

In this thesis, it is assumed that all the cells of a given QCA gate are in the same clocking zone. Further more, it is assumed that the binary wires that carry the input and output signals are in different clock zones than the gate itself. The input wire is in the

zone before the gate, and the output wire is in the zone after the gate. These wires are not shown in the figures depicting the gates.

2.1.5 Faults and Fault Tolerance

Two major categories of faults can occur during the assembly of a QCA circuit. First, faults may occur when quantum cells are displaced from their intended locations. These displacement faults may cause the cell to be outside the radius of effect of its neighbors, so that it is no longer contributing to the interactions among the cells. A typical radius of effect for a quantum cell is 65nm. The interactions between cells are due to the electrostatic quadrupole-quadrupole interactions between adjacent cells due to the two free electrons and two fixed protons in each cell. These forces fall off as the 5^{th} power of distance, so the radius of effect will always remain relatively small. A cell that is displaced may have a polarity opposite what it should. Sometimes displaced cells have no impact on the effectiveness of a QCA circuit, and sometimes they can cause a circuit to cease functioning as expected.

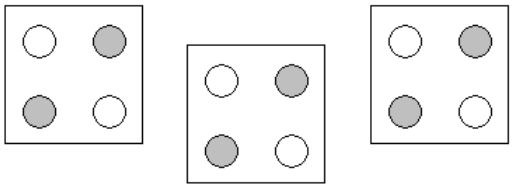

Figure 2-6: Center Cell Displaced from Intended Location in Binary Wire

A second type of fault occurs when the quantum cell itself is defective. Defective cells will not interact in the same way as ideal cells. The most obvious type of defective cell could be missing quantum dots or free electrons. In this case, the cell would be considered to be missing from the circuit, as it would have no influence on its neighbors. If the gap between ideal cells is large enough, the circuit will cease to function.

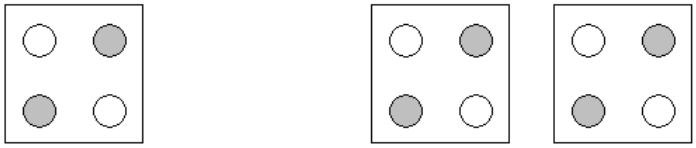

Figure 2-7: A Defective Cell in the Binary Wire

Systems that are robust enough to function correctly in the presence of faults are very important. In QCA systems the small-scale nature of the system provides enormous potential for faults to occur. Even small manufacturing defects correspond to great relative offsets. Cells must be aligned on nanometer scales in order to work correctly. Currently the technology does not exist to reliably align cells within such narrow tolerances. One way to avoid this problem is to design logic gates that are robust enough to continue to function in the presence of some faults.

Fijany and Toomarian designed and tested a fault-tolerant version of the Majority Voter gate. This gate uses an array of quantum cells (Figure 2-7) to provide redundancy. This cascaded array of Majority Voters allows some of the faults caused by positioning or defective cells to be ignored. It will always be possible that some set of defects will cause a gate to cease functioning or behave incorrectly. The hope is to design a gate that will continue to work under the widest set of potential defects.

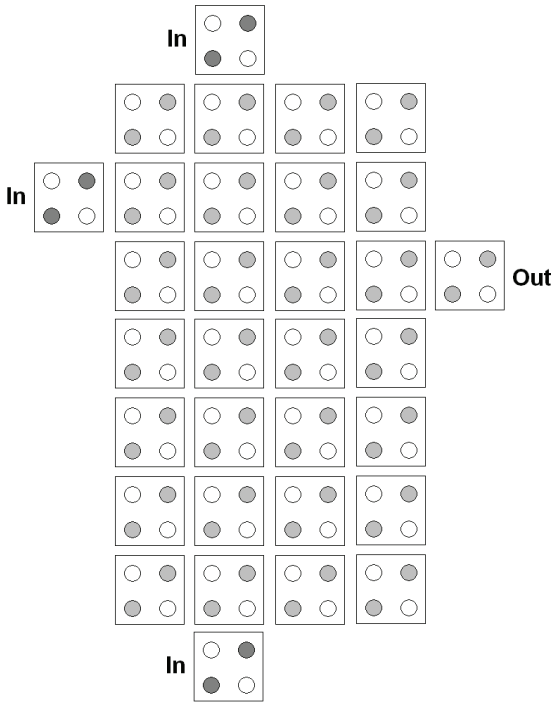

Figure 2-8: Fault-Tolerant QCA Majority Voter Block Gate

2.2 Advantages and Importance of QCA-Based Design

QCA offers several distinct advantages over the traditional transistor-based design. This schema inherently allows for very small feature size and thus high computational density. Because the intercellular interactions are governed by the relatively weak Coulombic repulsion forces, QCA devices operate better in smaller scales where the electrostatic forces are no longer dominated by thermal energies.

It has been projected that QCA circuits can achieve densities of 10^{12} devices/cm^2 and THz switching frequencies. Because current does not flow through QCA-based circuits, these designs can operate at very low power levels. This low power cost is vital to being able to achieve the device densities mentioned above.

QCA design supports massively parallel computational architectures, which can allow for more efficient information processing. In a QCA circuit that uses clocking to restore energy to the system, there is no “penalty” for branching out into parallel circuits

as there is with traditional CMOS architectures. In CMOS, current is divided between parallel branches of a circuit. This can quickly degrade the circuit if some sort of buffering mechanism is not used. In QCA, there is also energy loss, but it is the same energy loss that exists in a non-branching binary wire. As long as a mechanism is in place to restore this energy loss, a QCA circuit can be split into as many parallel branches as desired without negative effects from the branching. Clocking can be used to keep parallel branches synchronized with each other.

2.3 Difficulties of QCA-Based Design

Many obstacles must be overcome before QCA-based circuits are available as a viable alternative to transistor-based technologies.

First, quantum cells must be small, on the order of 20nm, to be efficient. Currently the technology does not exist to reliably manufacture quantum cells of this size and assemble them into particular structures. Fortunately much time and effort is being spent on these scale-related issues. The same technologies that allow transistors to be constructed at smaller and smaller scales will easily transfer to new technologies, such as QCA.

Secondly, as with any technology on this scale, it is difficult to create interfaces between the computational circuits themselves and I/O devices such as monitors and keyboards that would allow the user to interact with the computer. Again, this limitation is faced by other technologies. It is possible that this will need to be overcome in a step-wise fashion, with a traditional CMOS circuit determining the state of an output quantum cell, and relaying that information to I/O devices.

As in CMOS architectures, QCA structures also exhibit propagation delays. A long QCA wire will have a delay that is proportional to $N^{1.22}$, where N is the number of

cells in the wire. This delay can be attributed to the finite amount of time that it takes for the electrons in a cell to tunnel to their new position.

CHAPTER 3

PROPOSAL AND METHODOLOGY

3.1 Proposed Fault-Tolerant NOT Gate

The design of a fault-tolerant inverter is crucial to the success of QCA architecture. A fault-tolerant inverter should be robust enough to continue to operate correctly in the event that one or more of the cells in the array are misaligned. The simplest way to do this is to introduce redundancy into the design. Redundancy can be difficult to implement because it requires the input signal be split among redundant blocks and the results from all the blocks need to be synthesized back to a single output. Fortunately, QCA architecture inherently supports both. QCA signals do not degrade as they fan out. The QCA cell additionally behaves as a majority voter, making it simple to 00combine results into a single output.

I propose a redundant inverter, as shown below in Figure 3-1. This block inverter design is based on Fijany and Toomarian's block Majority Voter design. The design allows several paths of information travel between input and output. The cells in the gate behave as Majority Voters, synthesizing the states of its neighbor cells into a single output. This design allows some faults to be cancelled out by other cells that are in the correct state. This design will work for a limited number of faults.

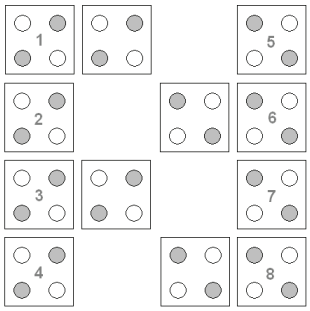

Figure 3-1: Proposed Fault-Tolerant Inverter

This design has four possible input positions (1, 2, 3, 4) and four possible output positions (5, 6, 7, 8). Simulations were run to determine the maximum output polarization for each of the input/output location combinations. The results are summarized below in Table 3-1.

		Output Position			
		5	6	7	8
Input Pos	1	0.928	0.994	0.980	0.987
	2	0.928	0.994	0.980	0.987
	3	0.928	0.994	0.980	0.987
	4	0.928	0.994	0.980	0.987

Table 3-1: Maximum Output Polarization for Proposed Inverter Design

The simulations indicate that the input position to this inverter has no effect on the output results. This observation is counterintuitive. I suspect that the four possible input cells are probably so close in polarization to each other that the effects of using different locations as inputs are too small to be measured.

The best output position is location 6. This site gives the strongest output signal. For symmetry, location 3 was selected as the input location to be used for analysis and simulation.

The use of larger arrays was also considered in the design of the fault-tolerant inverter. Table 3-2 shows the polarization values for an array with six possible inputs

and outputs. The polarizations at the outer-lying outputs were identical to the polarizations of the smaller array. To simplify the analysis, the smaller array was used for all simulations.

	Output Cell					
	1	2	3	4	5	6
1	0.987	0.980	0.994	0.981	0.994	0.928
2	0.987	0.980	0.994	0.981	0.994	0.928
3	0.987	0.980	0.994	0.981	0.994	0.928
4	0.987	0.980	0.994	0.981	0.994	0.928
5	0.987	0.980	0.994	0.981	0.994	0.928
6	0.987	0.980	0.994	0.981	0.994	0.928

Table 3-2: Maximum Output Polarization for Alternative Inverter Design

3.2 Simulation Details

To determine the tolerance of displacement faults in each design, each cell was moved independently to determine the maximum range of displacements for which the output was still correct. For these simulations, a valid value had to have a polarization magnitude of at least 0.400 (out of 1.000). It is assumed that values smaller than this will not be able to sufficiently "drive" an adjacent cell to the appropriate polarization.

The simulations were run using QCADesigner v2.0.3 (Windows version). The Bistable Simulation Engine was used for all simulations. These basic simulations can be run on a PC with no additional equipment. More advanced simulations will require the ability to automate the generation of different circuits and the simulation process.

QCADesigner's Bistable Simulation Engine uses an iterative process to determine the steady state of the system. Each cell is assumed to have only two possible states that correspond to polarization values of +1 and -1. The energy of each state can be calculated by computing the electrostatic energy between each cell and its neighbor cells. Equation 3-1 shows the calculation of energy between an electron in a quantum dot in cell *i* and one in cell *j*. ε_0 is the permittivity of free space and ε_r is the

relative permittivity of the material of the quantum cell. The energy of the cell can be calculated by summing over all dots in each cell.

$$E_{i,j} = \frac{1}{4\pi\varepsilon_0\varepsilon_r}\frac{q_i q_j}{|r_i - r_j|} \qquad \textbf{(Equation 3-1)}$$

The kink energy between two adjacent cells is defined as the difference in electrostatic energy between the two polarization states. This kink energy between cells *i* and *j*, $E_{i,j}^k$, is determined by keeping cell *i* in its fixed state and calculating the energy between the cells with cell *j* being in each of its two states. The kink energy is thus the difference between these two energies.

For a two-state system, such as we have here, the Hamiltonian shown in Equation 3-2 can be constructed to describe the state of the system, where P_j is the polarization value of cell *j*, and γ_i is the tunneling energy of electrons within cell *i*.

$$H_i = \sum_j \begin{bmatrix} -\frac{1}{2}P_j E_{i,j}^k & -\gamma_i \\ -\gamma_i & \frac{1}{2}P_j E_{i,j}^k \end{bmatrix} \qquad \textbf{(Equation 3-2)}$$

The time-independent Schrödinger equation (Equation 3-3) can be used to determine the states of the cells described by this Hamiltonian. The solution to the Schrödinger equation can be reduced to an expression to describe the polarization of each cell, as shown in Equation 3-4.

$$H_i\psi_i = E_i\psi_i \qquad \textbf{(Equation 3-3)}$$

$$P_i = \frac{\frac{E_{i,j}^k}{2\gamma}\sum_j P_j}{\sqrt{1+\left(\frac{E_{i,j}^k}{2\gamma}\sum_j P_j\right)}} \qquad \textbf{(Equation 3-4)}$$

For all simulations the default values for the simulator are used, as seen in Figure 3-2. The Number of Samples parameter is used to fix the precision of the simulation. Each sample represents a data point on the resulting output waveform. This engine uses an iterative approach to determine the output values. For a given set of inputs, the entire system is allowed to converge to a stable state. The Convergence Tolerance defines at what value the system is assumed to converge. For some situations the system may not reach a steady state or may take a long time to settle into that state. In these situations, the Maximum Iterations per Sample is used to ensure that the simulations can continue. The Radius of Effect is used in calculating which cells contribute to the state of a neighbor cell. In multi-layer QCA circuits, the Radius of Effect determines which layers will affect the design. The Relative Permittivity is used for the calculation of the kink energy. This default value is for GaAs/AlGaAs, and is assumed to be sufficient for most calculations. QCADesigner uses a cosine signal to generate the clock signal. The Clock High and Clock Low parameters are used to set the cutoff positive and negative cutoff values for this cosine. The Clock Shift allows the clock signal to be shifted positive or negative. The Clock Amplitude factor will affect how quickly the clock signal transitions between low and high states. The Layer Separation parameter is used to define how much space exists between circuit layers. In these simulations, all circuits are entirely contained in a single layer, so the Layer Separation value has no effect.

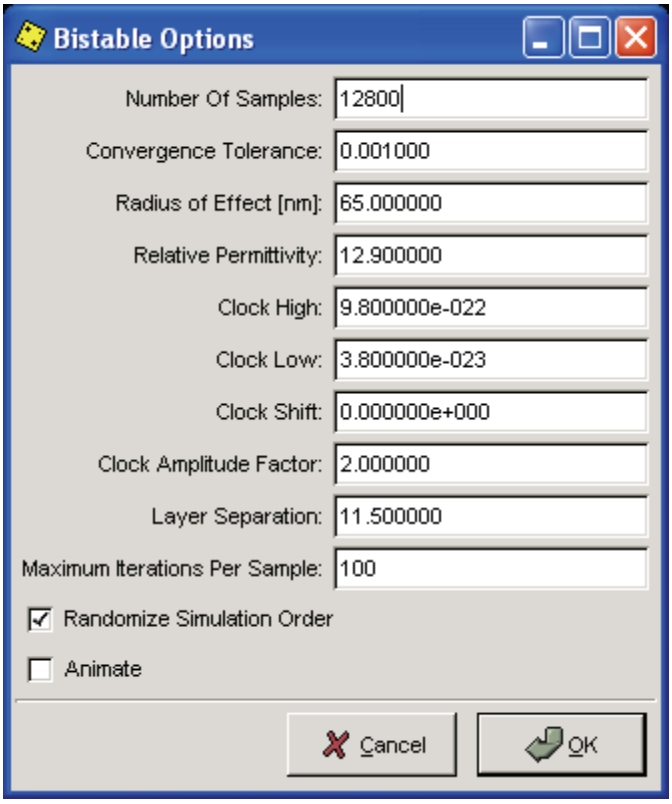

Figure 3-2: QCADesigner Bistable Simulation Engine Options

Each cell has a length of 18nm and a quantum dot diameter of 5nm. The spacing between each cell is 2nm. The horizontal and vertical spacing between the dots in a cell is 9mn. These values are summarized below in Table 3-2.

Parameter	Value
Cell Length	18nm
Cell Spacing	2nm
Dot Diameter	5nm
Dot Spacing	9nm

Table 3-3: QCA Simulation Parameters

CHAPTER 4

ANALYSYIS OF DIFFERENT INVERTER DESIGNS

4.1 A Standard Inverter

When QCA literature references an inverter or NOT gate, it usually appears in the form depicted in Figure 4-1. This gate is sometimes shown with binary wires leading into and out of the input and output cells. These cells do not contribute to the signal inversion, so they are not modeled in these simulations.

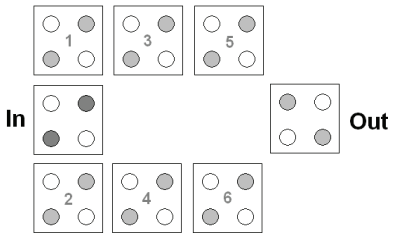

Figure 4-1: Standard QCA Inverter

Simulations revealed that, as expected, this design has a fairly low tolerance for displacement faults. The maximum range of each cell in nanometers is shown in Table 4-1, and shown visually in Figure 4-2. Some of the cells can be moved infinitely far in a given direction from their standard location. These are indicated as “inf” in the table and figure below. Cells whose movement is restricted by adjacent cells show maximum displacement as “-“ in the table below.

	Up	Down	Left	Right
1	4nm	-	4nm	-
2	-	4nm	5nm	-
3	inf	14nm	-	-
4	14nm	inf	-	-
5	inf	16nm	-	inf
6	16nm	inf	-	inf
In	-	-	7nm	4nm
Out	9nm	9nm	11nm	6nm

Table 4-1: Maximum Range for Cells in Standard Inverter

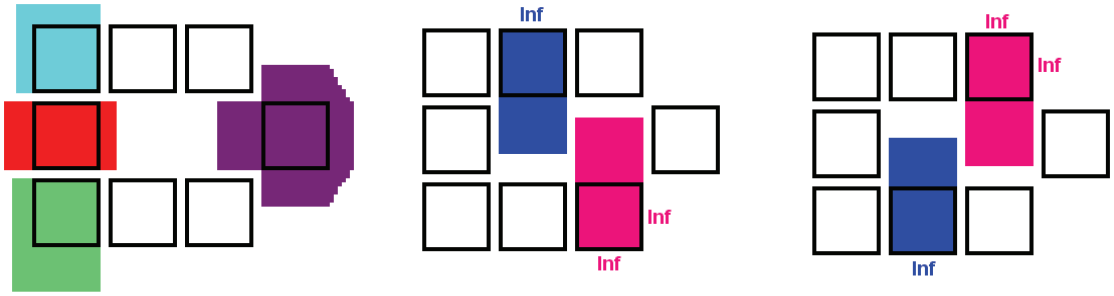

Figure 4-2: Maximum Range for Cells in Standard Inverter

4.2 A Simple Inverter

The simplest inverter consists of five cells as shown below in Figure 4-3. For the purposes of analysis and discussion, the cells will be identified as Cell 1, Cell 2, Cell 3, Input Cell, and Output Cell. This NOT Gate relies on the output branch (Cell 3 and Output Cell) being positioned in such a way that it is offset by half a cell width from the input branch (Input Cell, Cell 1, Cell 2) of the gate.

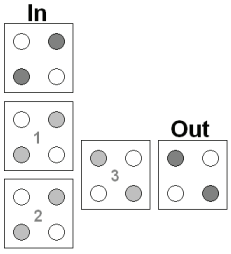

Figure 4-3: Simple NOT Gate

This inverter model proved to be more robust in simulation than anticipated. The simulation results do not seem to agree with manual simulations. Of greatest concern is the case where Cell 2 is missing entirely. QCADesigner simulations show that the signal is inverted and the output remains well within the required tolerances.

Figure 4-4 shows the results of the QCADesigner simulations. This diagram indicates the maximum range for each of the cells when moved independently.

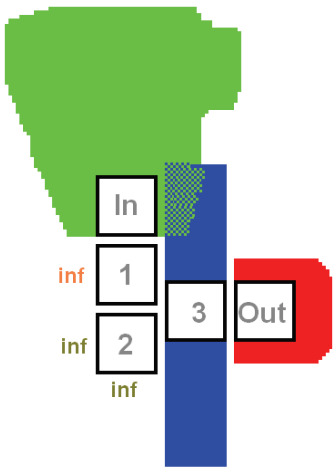

Figure 4-4: Maximum Range for All Cells in the Simple Inverter

The simulations indicate that Cell 1 and Cell 2 can be in any position. In fact, these cells may be missing entirely from the design. If one of the two cells is in the appropriate location, the other cell no longer matters to the design.

The Input Cell has the largest range in which it may lie. The green region indicates the input range. The Input Cell has the strongest polarization (magnitude 1), so it stands to reason that it will have influence over larger distances. In the simulations, the polarization of each successive cell is decreased. With the vertical position fixed, the Input Cell can be offset horizontally from its intended coordinates by up to 8nm. The cell can be located up to 48nm up from its designated coordinates. The asymmetry in the green region can be attributed to the influences of Cell 3 and the Output Cell.

Because of the constraint requiring cells to be at least 2nm apart, Cell 3 is limited to moving up and down. It can be displaced upward by up to 16nm and downwards by up to 19nm. The influence of the Input Cell limits the upward movement of this cell. If Cell 3 is missing from the design, the inverter fails to work. The output polarization is too low in this case. The simulations require that another cell be adjacent to the output cell to drive the cell to the appropriate state.

The Output Cell has a very narrow range of motion. It must be located within 6nm up or down and within 10nm right of its intended position.

4.3 A Fault-Tolerant Inverter

The fault-tolerant design being analyzed here is the inverter that was designed in Chapter 3. This design relies on the Majority Voter-like behavior of the QCA cell and the fact that there are redundant paths for information to travel through the gate. Figure 4-5 defines the numbering scheme for the cells. Table 4-2 summarizes the range of motion for this inverter design, while Figure 4-5 represents these values graphically. Some cells have no limitation to their movement in a given direction. These cases are indicated with a value of “inf”. Other cells are restricted in movement by neighboring cells, their maximum movement is indicated by ”-“.

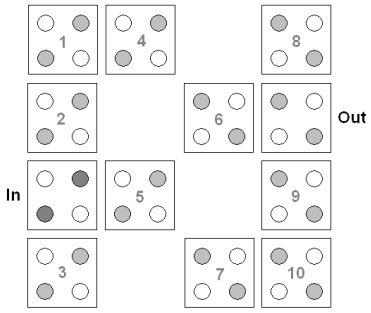

Figure 4-5: Fault-Tolerant Inverter

	Up	Down	Left	Right
1	inf	-	inf	-
2	-	-	inf	17nm
3	-	inf	inf	inf
4	inf	15nm	-	inf

5	6nm	6nm	-	11nm
6	7nm	6nm	inf	-
7	15nm	inf	inf	-
8	inf	-	inf	inf
9	-	-	15nm	inf
10	-	inf	-	inf
In	-	-	44nm	-
Out	-	-	-	8nm

Table 4-2: Maximum Range for Cells in Fault-Tolerant Inverter

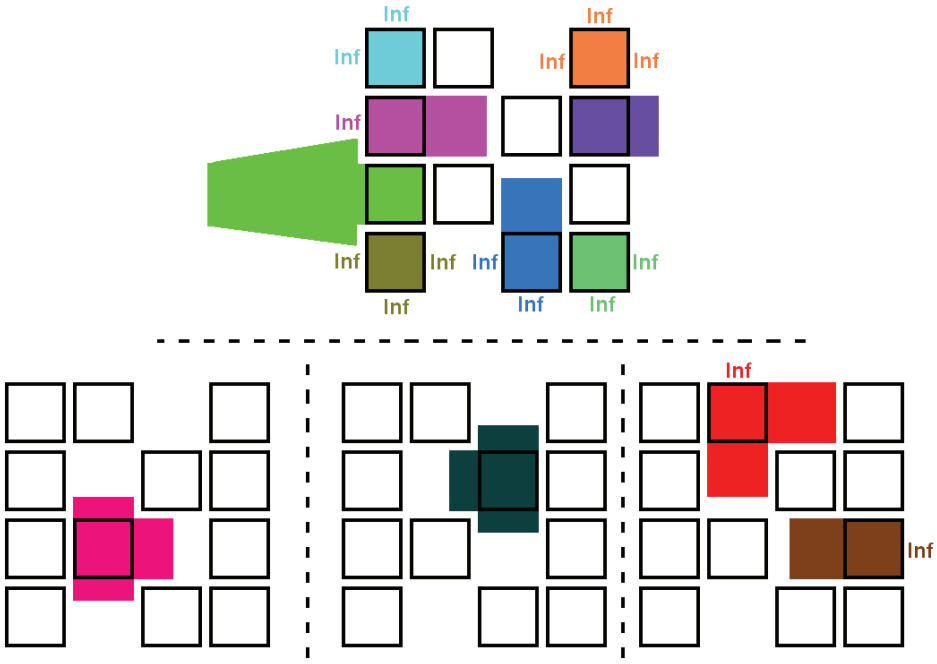

Figure 4-6: Maximum Range for Cells in Fault-Tolerant Inverter

Analysis of the inverter shows that this design is robust in the presence of moderate displacement faults. The cells that are most vulnerable to displacement are located in the middle of the inverter. Cells 4 through 7 cause the inversion effect on the output. If these cells are not properly aligned vertically, the NOT Gate ceases to function.

Moving the Output Cell more than 8nm to the right results in an inversion of the signal. The same inversion effect should be observed on the Input Cell as well.

However, this inversion is not seen in QCADesigner simulations. This should be examined further.

CHAPTER 5

MANUAL CALCULATIONS

Manual calculations can be preformed to verify the validity of simulation results. For simple systems, it can be assumed that an input cell with a fixed input polarization will force its near neighbors to assume a state with the least possible electrostatic energy. The electrostatic energy between quantum dots *i* and *j* is given by

$$E_{i,j} = \frac{1}{4\pi\epsilon_0\epsilon_r} \frac{q_i q_j}{|r_i - r_j|}$$

where ε_0 is the permittivity of free space, ε_r is the relative permittivity of the material system, q_i and q_j and are the charges of dots *i* and *j*, and $|r_i - r_j|$ is the distance between the two dots. For a system with multiple dots, this energy is summed over all *i* and *j*.

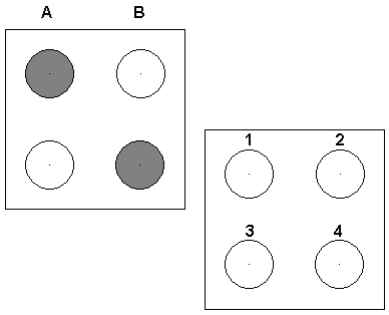

Figure 5-1: Simple Inversion with Input Polarity -1

Non-Inverting			
	distance		
	x	y	total
A1	20nm	10nm	22.36068nm
A4	29nm	19nm	34.66987nm
B1	11nm	1nm	11.04536nm
B4	20nm	10nm	22.36068nm

Inverting			
	distance		
	x	y	total
A2	29nm	10nm	30.67572nm
A3	20nm	19nm	27.58623nm
B2	20nm	1nm	20.02498nm
B3	11nm	10nm	14.86607nm

Table 5-1: Distance Comparisons of Cells with Different States

The table above shows the distance between the fixed charges on the input cell and each of the dots in the other cell. In the energy equation above, the only variable term is the distance between dots. All other terms are constant. Therefore, to compare the electrostatic energies of the two states, we must compare the sums of the inverses of each of the distances. $E_{inverting}$ - $E_{non\text{-}inverting}$ is the kink energy of the system, as described in section 3-2.

$$E_{non\text{-}inverting} \sim= 1/A1 + 1/A4 + 1/B1 + 1/B4 = 0.208822$$

$$E_{inverting} \sim= 1/A2 + 1/A3 + 1/B2 + 1/B3 = 0.186054$$

Since the inverting state has the lowest energy level, that is the equilibrium state for the output cell.

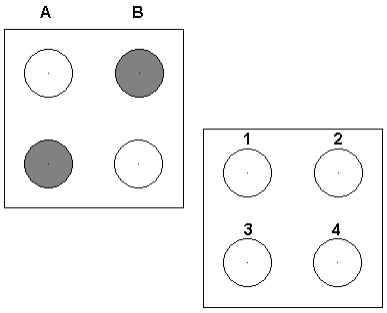

Figure 5-2: Simple Inversion with Input Polarity +1

Inverting			
	distance		
	x	y	total
A1	20nm	1nm	20.02498nm
A4	29nm	10nm	30.67572nm
B1	11nm	10nm	14.86607nm
B4	20nm	19nm	27.58623nm

Non-Inverting			
	distance		
	x	y	total
A2	29nm	1nm	29.01724nm
A3	20nm	10nm	22.36068nm
B2	20nm	10nm	22.36068nm
B3	11nm	19nm	21.9545nm

Table 5-2: Distance Comparisons of Cells with Different States

$$E_{inverting} \sim= 1/A1 + 1/A4 + 1/B1 + 1/B4 = 0.186054$$

$$E_{non\text{-}inverting} \sim= 1/A2 + 1/A3 + 1/B2 + 1/B3 = 0.169454$$

In this case, the inversion will be expected to occur.

The results still hold in the case where there is an additional cell on the input branch of the circuit.

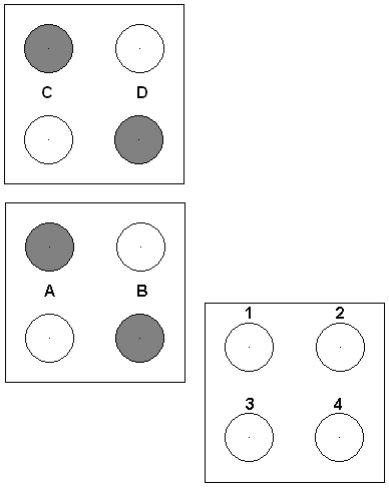

Figure 5-3: Two-Input Cell Inversion with Polarity -1

Non-Inverting			
	distance		
	x	y	total
A1	20nm	10nm	22.36068nm
A4	29nm	19nm	34.66987nm
B1	11nm	1nm	11.04536nm
B4	20nm	10nm	22.36068nm
C1	20nm	30nm	36.05551nm
C4	29nm	39nm	48.60041nm
D1	11nm	21nm	23.70654nm
D4	20nm	30nm	36.05551nm

Inverting			
	distance		
	x	y	total
A2	29nm	10nm	30.67572nm
A3	20nm	19nm	27.58623nm
B2	20nm	1nm	20.02498nm
B3	11nm	10nm	14.86607nm
C2	29nm	30nm	41.72529nm
C3	20nm	39nm	43.82921nm
D2	20nm	21nm	29.00000nm
D3	11nm	30nm	31.95309nm

Table 5-3: Distance Comparisons of Cells with Different States

$E_{non\text{-}inverting}$ ~= 0.32705

$E_{inverting}$ ~= 0.298615

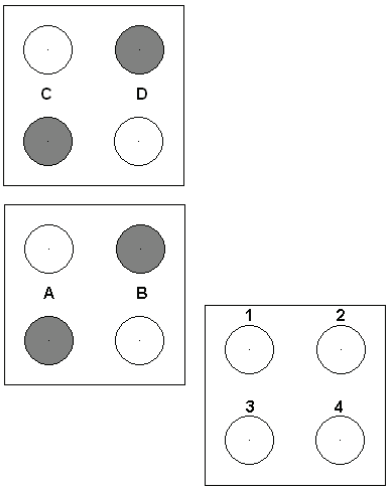

Figure 5-4: Two-Input Cell Inversion with Polarity +1

Inverting			
	distance		
	x	y	total
A1	20nm	1nm	20.02498nm
A4	29nm	10nm	30.67572nm
B1	11nm	10nm	14.86607nm
B4	20nm	19nm	27.58623nm
C1	20nm	21nm	29.00000nm
C4	29nm	30nm	41.72529nm
D1	11nm	30nm	31.95309nm
D4	20nm	39nm	43.82921nm

Non-Inverting			
	distance		
	x	y	total
A2	29nm	1nm	29.01724nm
A3	20nm	10nm	22.36068nm
B2	20nm	10nm	22.36068nm
B3	11nm	19nm	21.95450nm
C2	29nm	21nm	35.80503nm
C3	20nm	30nm	36.05551nm
D2	20nm	30nm	36.05551nm
D3	11nm	39nm	40.52160nm

Table 5-4: Distance Comparisons of Cells with Different States

$$E_{inverting} \sim= 0.298615$$

$$E_{non\text{-}inverting} \sim= 0.277531$$

These simulations assume that the electrons are located in the exact center of the quantum dots. However, further analysis shows that if the electrons are located at any of the extremes of the cells, the overall results are still the same. The +1 input polarization always suggests that it should produce a non-inverted output. Likewise, -1 input polarization always generates an inverted output.

It can also be assumed that the QCA cells will not be perfectly planar, and as such, there will be some freedom of motion for the electrons out of the plane of the cell. Calculations of the effect of this displacement indicate that it does not significantly affect the results. For these calculations it is assumed that the electrons can be displaced in the normal direction by ±2nm,

$$E_{non\text{-}inverting} \sim= 0.184711$$

$$E_{inverting} \sim= 0.203222$$

Inverting				
	distance			
	x	y	z	total
A1	20nm	10nm	0nm	22.36068
A4	29nm	19nm	4nm	34.89986
B1	11nm	1nm	4nm	11.74734
B4	20nm	10nm	0nm	22.36068

Non-Inverting				
	distance			
	x	y	z	total
A2	29nm	10nm	0nm	30.67572
A3	20nm	19nm	4nm	27.87472
B2	20nm	1nm	4nm	20.42058
B3	11nm	10nm	0nm	14.86607

Table 5-5: Calculations of Electron Displacement in Cell with +1 Polarization

CHAPTER 6

RESULTS AND CONCLUSIONS

The proposed fault-tolerant inverter proved to be better adapted for displacement faults than the standard NOT Gate. The Input Cell has a range more than 6 times (44nm instead of 7nm) farther in the fault-tolerant design, but the range of the Output Cell only increased from 6nm to 8nm. However, the range of the interior cells in the fault-tolerant design actually decreased. One feature of the redundancy in the design is that many of the exterior cells have complete freedom of motion when each fault is analyzed independently.

The simple inverter proved to be quite a surprise. I had expected that it would be extremely vulnerable to displacements, but the simulations indicate that it is much more forgiving of this type of fault than any other design I tested. The biggest drawback for this design is that it requires a manufacturing process that can accurately place cells at half-cell intervals. The main premise of this thesis is that such precise placement is not possible at this time. Thus, this design may be worth looking into more in the future when the technology is available.

A comparison of the smallest maximum displacement for a cell in each of the inverter types is shown below in Table 6-1. The displacement values of the cells are compared excluding and including the input and output cells. Both the Fault-tolerant design presented in this thesis and the simple inverter were less susceptible to displacement faults.

	Fault-Tolerant	Standard	Simple
Without I/O	6nm	4nm	8nm
Including I/O	6nm	4nm	6nm

Table 6-1: Smallest Maximum Displacement

Several problems were encountered with the QCADesigner simulator. The first problem was with signal degradation in the simulations. The QCA clocking mechanism should mostly compensate for signal loss in a circuit. The lack of compensation for this required the cells in each design be closely spaced. Simulations that allow for the cells to be more widely spaced, and thus have a greater degree of freedom in their movements, would be required before the proposed design could be definitively called an improvement.

The second problem with the simulator is that I do not believe that some of the results are accurate. For example, the simulations indicate that the simple inverter will function properly even if the lower left cell is missing completely, as shown in Figure 6-1. However, I do not believe that the output is guaranteed to be inverted, as discussed in Chapter 5.

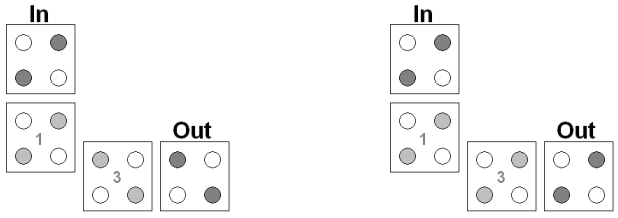

Figure 6-1: Faulty Simple Inverter

QCADesigner also gives different results depending on how many adjacent cells there are to a given output. The most noticeable instance of this was when I ran simulations on a fault-tolerant inverter that had the input and output cells extending from the sides of the array. The results of the simulations to find the strongest input and

output positions indicated that the output should be in one of the corners. The design of the inverter was changed to give more accurate results.

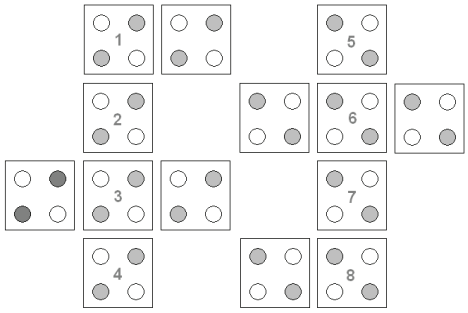

Figure 6-2: Fault-Tolerant Inverter Yielding Poor Simulations

These simulations indicate that it is possible to use a more robust form of the QCA inverter using the same techniques that Fijany and Toomarian used to create their robust Majority Voter. I do not feel like the results are conclusive at this point. More work must be done to improve the simulation process. More detailed analysis of all three inverters with multiple positional faults will also be required.

Further investigation into QCA inverters will be required. Communication with the QCA Design community indicates that simulating inverters in QCADesigner can lead to results that are difficult to replicate. Analysis of these designs with a different simulation tool would be an appropriate first step to validating these results.

REFERENCES

REFERENCES

Amlani, I., Orlov, A.O., Snider, G.L., Lent, C.S., "Demonstration of a Six-Dot Quantum Cellular Automata System." *Applied Physics Letters* 72(17): 2179-2181, April 1998.

Finjay, A., Toomarian, B.N., "New Design for Quantum Dots Cellular Automata to Obtain Fault Tolerant Logic Gates." *Journal of Nanoparticle Research* 3: 27-37, 2001.

Heath, J.R., Kuekes, P.J., Snider, G.S., Williams, R.S., "A Defect-Tolerant Computer Architecture: Opportunities for Nanotechnology." *Science* 280(5370): 1716-1721 June 1998.

Kummamuru, R.K., Timler, J., Toth, G., Lent, C.S., Ramasubramaniam, R., Orlov, A.O., Bernsteing, G.H., Snider, G.L., "Power Gain in a Quantum-Dot Cellular Automata Latch." *Applied Physics Letters* 81(7): 1332-1334, August 2002.

Lent, C.S., Isaksen, B., "Clocked Molecular Quantum-Dot Cellular Automata." *IEEE Transactions on Electron Devices* 50(9): 1890-1896, September 2003.

Lent, C.S., Tougaw, P.D., Porod, W., Bernstein, G.H., "Quantum Cellular Automata." *Nanotechnology* 4: 49-57, 1993.

Niemier, M.T., Rodrigues, A.F., Kogge P.M. "A Potentially Implementable FPGA for Quantum Dot Cellular Automata." Presented at the 1st Workshop on Non-Silicon Computation (NSC-1), Boston MA, 2002.

Orlov, A.O., Amlanie, I., Toth, G., Lent, C.S., Bernstein, G.H., Snider, G.L., "Experimental Demonstration of a Binary Wire for Quantum-Dot Cellular Automata." *Applied Physics Letters* 74(19): 2875-2877, May 1999.

Orlov, A.O., Kummamuru, R.K., Ramasubramaniam, R., Toth, G., Lent, C.S., Bernstein, G.H., Snider, G.L., "Experimental Demonstration of a Latch in Clocked Quantum-dot Cellular Automata." *Applied Physics Letters* 78(11): 1625-1627, March 2001.

Qi, H.; Sharma, S.; Li, Z.; Snider, G.L.; Orlov, A.O.; Lent, C.S.; Fehlner,T.P. "Molecular Quantum Cellular Automata Cells. Electric Field Driven Switching of a Silicon Surface Bound Array of Vertically Oriented Two-Dot Molecular Quantum Cellular Automata." *Journal of the American Chemical Society* 125: 15250-15259, November 2003.

Smith, C.G., "Nanotechnology: Computation Without Current." *Science* 284(5412): 274, April 1999.

Tahoori, M.B., Huang, J., Momenzadeh, M., Lombardi, F., "Testing of Quantum Cellular Automata." *IEEE Transactions on Nanotechnology* 3(4): 432-442, December 2004.

Tahoori, M.B., Momenzadeh, M., Huang, J., Lombardi, F., "Defects and Fault Characterization in Quantum Cellular Automata." Presented at NanoTechnology Conference and Trade Show, 2004.

Timler, J; Lent, C.S. "Power Gain and Dissipation in Quantum-dot Cellular Automata." *Journal of Applied Physics* 91(2): 823-831, January 2002.

Toth, G., Lent, C.S., “Quasi-Adiabatic Switching for Metal-Island Quantum-Dot Cellular Automata”

Tougaw, P.D., Lent, C.S., “Logical Devices Implemented Using Quantum Cellular Automata.” *Journal of Applied Physics* 75(3), 1818-1825, February 1994.