ABSTRACT

Quantum-dot Cellular Automata as an Implementation for Random Number Generation

Heath McCabe

Director: Enrique Blair, Ph.D.

Quantum-dot Cellular Automata (QCA) provides a viable low-power alternative to conventional implementations of classical computing machines. QCA cells with no biasing voltage will yield a "1" or a "0" with a 50% chance of being "1" and 50% chance of being "0" upon measurement. Applying a bias voltage to a QCA cell allows this probability to be tuned such that the probability of measuring a "1" could range anywhere from 0 to 1. Many applications benefit from equal probabilities of measuring "0" or "1," but some applications such as stochastic computing require having an adjustable probability of measurement outcomes. Performing a series of measurements can be used to serially create a random number of any desired size. Thus, tuning the probability of a QCA cell can be used as an implementation for random number generation. Furthermore, this system is suitable for applications in which zero outcome bias is desired, or a specific and dynamically-tunable bias is desired. We discuss the quantum mechanics of random number generation using a QCA cell, as well as different physical implementations for a QCA random number generator.

APPROVED BY DIRECTOR OF HONORS THESIS:

_____

Dr. Erik Blair, Department of Electrical and Computer Engineering

APPROVED BY THE HONORS PROGRAM:

_____

Dr. Elizabeth Corey, Director

DATE: _____

QUANTUM-DOT CELLULAR AUTOMATA AS AN IMPLEMENTATION FOR RANDOM NUMBER GENERATION

A Thesis Submitted to the Faculty of

Baylor University

In Partial Fulfillment of the Requirements for the

Honors Program

By

HEATH MCCABE

Waco, Texas

May 2019

TABLE OF CONTENTS

ACKNOWLEDGMENTS

I owe a very special gratitude to my Thesis Director, Dr. Erik Blair. I appreciate the wisdom Dr. Blair provided and the time spent helping me bring this project together.

Thank you to Dr. Scott Koziol and Dr. Howard Lee for serving as Thesis Committee members.

I am also grateful to my friends and family, who provided me with immeasurable support during my time at Baylor, and throughout my time spent working on this Honors Thesis.

CHAPTER ONE

Introduction

The eternal objective for electrical engineering research is to make smaller, faster, more efficient, and more powerful electronics. To that end, one important benchmark is Moore's Law: the projection that the number of transistors in a given area of a processor will double roughly every 2 years. At this projected rate of increase, eventually the devices will become too small for the underlying physics to continue behaving appropriately. Additionally, as devices become smaller, their power consumption and heat output increase. There is a certain point at which the heat output of the device will damage it to the point of self-destruction, a side effect which must be avoided. Two key limiting factors, the size of the devices causing unwanted quantum mechanical oddities and the heat dissipation of the devices, are key motivators for QCA logic devices. Quantum-dot Cellular Automata (QCA) is proposed as a next step in computing device technology as well as a new approach for random number generation.

Motivation for QCA

Transistors are becoming very small and are approaching a limit beyond which quantum mechanics will introduce unwanted effects into device operation [1] [2]. A new solution will be needed in order to create devices that are more powerful once that size limit is reached and silicon devices cannot be shrunk any further. An explanation of the implications regarding the present topic is given by McIntyre [3].

Another key problem facing the future of transistor-based computing is heat dissipation. As a conductor gets smaller, a constant current passing through it will

create more heat; this is exactly what is occuring in silicon devices which, is a key reason why processor clock rates are leveling off. The power equation, Equation 1 below, helps to explain this phenomena by proving that power is directly related to area. Power traveling through wires tends to convert itself into thermal energy, as anyone who has unplugged a cell phone cord from the wall after it has been in use can confirm. So, in turn, smaller wires tends to create more heat output. This relationship is explored in depth in Lent's article [2].

$$\begin{aligned} P &= IV \\ &= I^2R \\ &= I^2(\rho\frac{L}{A}) \end{aligned} \tag{1}$$

QCA is one potential solution to this problem. The nature of Quantum-dot Cellular Automata QCA is that it does not rely on currents traveling through wires. This allows it to operate at very high speeds without becoming overheated. [2], [4]. QCA devices can help us continue to make our computers more efficient, more powerful, and faster [2].

Introduction to QCA Operation

QCA circuits are made up of cells that act both as wires and logic gates. An array of cells in a line acts as a wire, transmitting a bit down the line. Particular structures of cells can implement the fundamental logic operations (AND, OR, NOT). The cell-cell response function illustrated in Figure 2 shows the polarization induced on cell 1 based on driver cell 2. This demonstrates that a target cell can have a high magnitude polarization from a rather small driver polarization, which can allow for very fast clocking of QCA circuits. In Figure 3, it can be seen that for larger tunneling energy γ, the target cell's polarization has a more linear response to the driver cell's

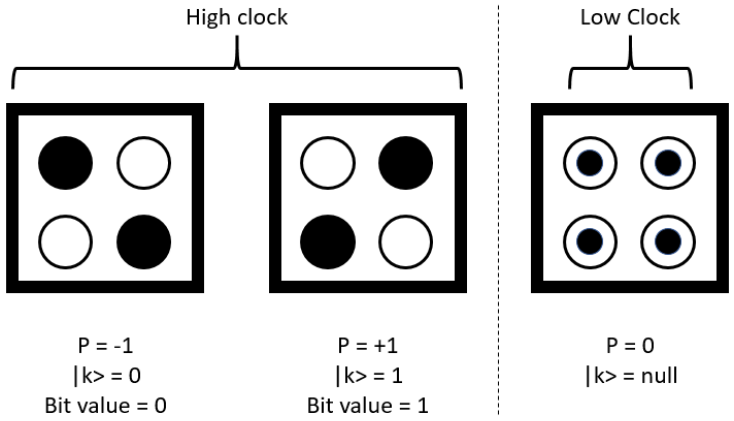


Figure 1: QCA Cell States. This figure illustrates the possible states for a 4-dot QCA cell.

polarization. It is also clear that for small tunneling energy γ only a small positive polarization is needed on the driver to make the target cell have a bit value of 1.

Basic Operation

A molecular QCA cell is made up of two molecules, each having one free electron. The cell as a whole then has two free electrons within it, which can be localized into one of two possible configurations when the cell is in the "active state". The active state is synonymous with a "high clock" state. These two possible configurations in which the electrons may localize are termed "polarizations," which are quantified by assigning a value P $\in [-1, +1]$. The computational basis states for QCA are $|0\rangle$ and $|1\rangle$ which correspond to $P = -1$ and $P = +1$, respectively, as illustrated in Figure 1. That is to say that "logic states are encoded no longer as voltages but rather by the positions of individual electrons" [2].

Clocking

Clocking helps transmit data through a circuit, and provides synchronization for complex circuits. Clocking in QCA is achieved by forcing the electrons into an

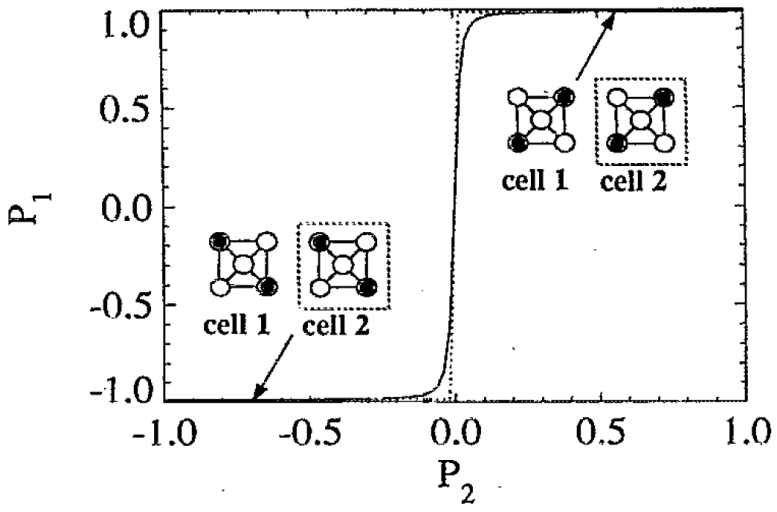


Figure 2: Cell-Cell Response Curve [5] showing how a target cell's polarization is affected by a changing driver cell polarization.

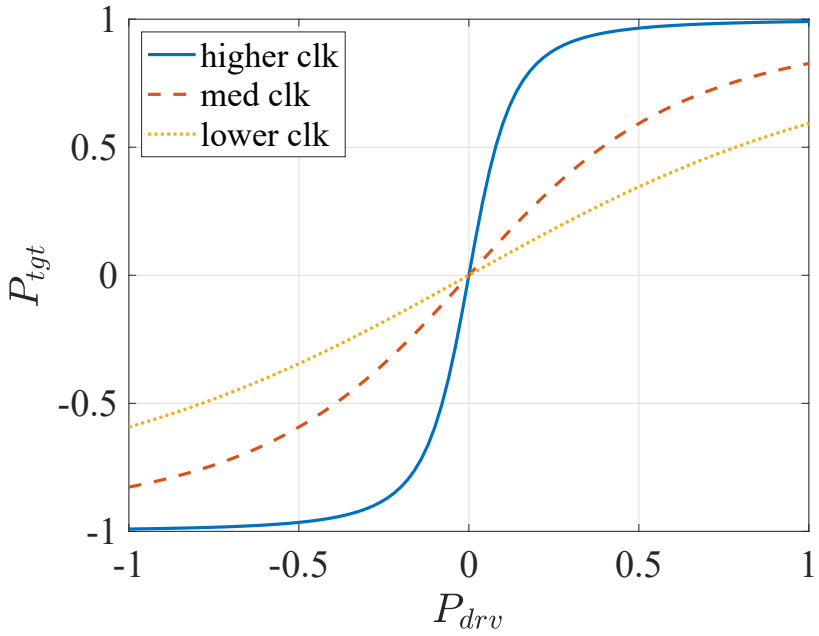


Figure 3: Cell-Cell Response Curve generated in Matlab. This plot shows the polarization of a target cell as a function of driver cell polzarization. The three curves show that a higher γ (clock) makes the target cell more sensitive to the driver polarization.

active state at specific intervals. In Figure 1, the two cells shown on the left are active and the right cell is in the null state. Thus when the clock is "high" the cells in that clock region have valid data and when the clock is "low" the cells in that region are in a null state. When the clock signal goes high, the cells in that clock zone will be pushed into the active state and assume the bit values that would be expected from the circuit design for that moment in time. Then when the clock signal goes low again, the cells in that clock zone are pulled back down into the null state and are ready to evaluate logic again the next time the clock goes high.

Figure 5 shows the output of the simulation for the majority gate shown in Figure 4a. Notice that the output enters an active state when clock 1 goes low. This is a little counterintuitive because typically, one would say that a "high clock" puts the cells in its area into an active state, but this is simply how the QCADesigner tool operates.

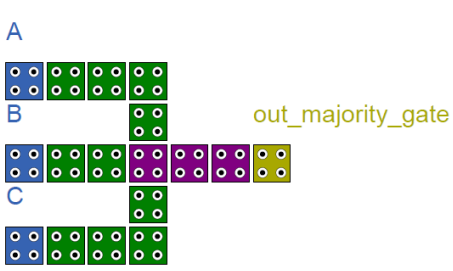
Majority Gate

The majority gate is the natural method of implementing the AND and OR operation in QCA. This gate requires **exactly** 3 inputs and produces 1 output. Figure 4a shows how a majority gate is implemented using QCA cells. Figure 5 shows the simulation results for the majority gate schematic. For clarification and for verification, the truth table for the majority gate shown in Figure 6 can be cross referenced with the simulation results.

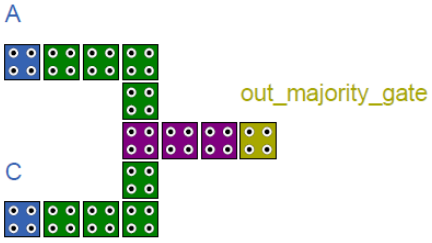
The majority gate can be made to function as an AND gate by forcing one of the inputs to be 0, because achieving a majority vote requires that both of the remaining inputs be 1. This can be verified by examining the top half of the majority gate's truth table which is shown in Figure 6. Similarly, the majority gate can be made to function as an OR gate by forcing one of the inputs to be 1, because achieving a majority vote requires only one remaining input be a 1. This can be verified

by examining the bottom half of the majority gate's truth table which is shown in Figure 6.

It is important to note that the majority gate must have 3 inputs because of the physics that underlies its operation. Coulombic interaction between the 2 free electrons of one cell and those of its neighbor enables QCA to operate, and because of the superposition of Coulombic interactions, when three cells meet as shown in the majority gate schematic in Figure 4a, the result is "computed" on the center cell (the leftmost purple cell as shown in the same figure). Two wires with the same bit value can be joined together, as in Figure 4b, to produce an output with the same bit value. However, two wires joining together in the same way, but with different bit values, will not produce any consistent result. This is important to note: there is not a natural logic operation in QCA that takes two inputs.



(a) This is a proper majority gate with 3 inputs.



(b) This is an improper schematic, as there is not a natural logic gate in QCA having two inputs.

Figure 4: Majority Gate Schematic

QCA Circuits and Simulation

QCA circuits are made up of binary wires, inverters, majority gate, and programmable AND/OR gates. This project primarily relies on circuit design and simulation via a library of Matlab code. It allows for visualizing a QCA circuit, and takes into account the complex calculations involved in expanding the Hamiltonian matrix to the full

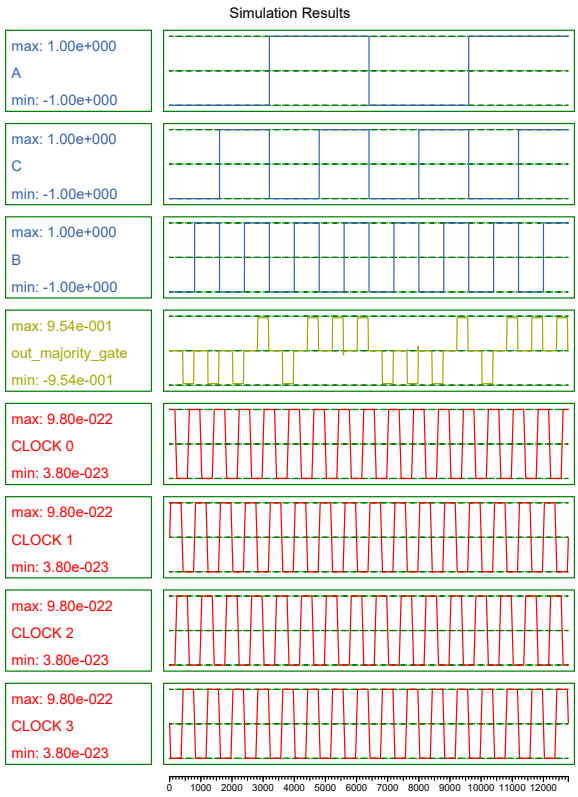


Figure 5: Majority Gate simulation result generated using QCA Designer. All eight possible inputs are shown in the top three rows. The output of the majority gate is shown in the fourth row. The four clock zones used in the schematic are shown in the bottom four rows. By using four clock zones, each area of the circuit is able to have active data in it, while the other zones are able to process different data.

Hilbert space, in order to accurately model the relationships between cells. Pictured in Figure 7 is a wire entering an inverter on the left and an output wire exiting on the right which is designed and simulated via this collection of Matlab code.

Another tool which is used for circuit design and simulation is QCADesigner. This tool allows for simple design and up to four clock phases, and a nice graphical output feature. Figure 4 shows a majority gate design using QCADesigner, with the inputs A, B, and C on clock 1, the green cells on clock 2, the purple cells on clock 3,

A	B	C	M(A,B,C)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Figure 6: Majority Gate truth table. From this, it can be seen that taking $A = 0$ causes AND gate behavior, and taking $A = 1$ causes OR gate behavior.

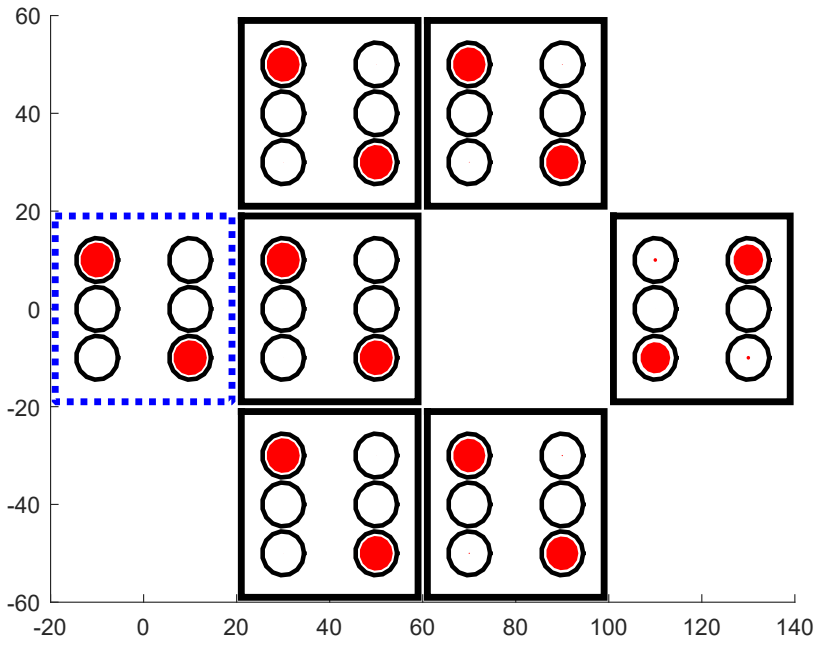


Figure 7: Inverter schematic generated via Matlab. This is a circuit of 6-dot cells, which have thre possible states: $|0\rangle, |1\rangle, |null\rangle$.

and the yellow output cell on clock 4. Figure 5 shows the simulation output for this same majority gate.

QCA Progress

QCA technologies have been simulated and implemented with promising results. Quantum mechanical calculations can simulate the operation of a QCA array and provide a proof of concept before researchers take the steps to implement circuits in hardware. As of 1994, Lent had performed mathematical simulations proving the proper functioning of a QCA full adder circuit [5]. These simulations prove the possibility of QCA full adder circuits, which inspires confidence that more complex and useful QCA circuits are forthcoming. An interesting takeaway from this same research is the concept that "we can confidently predict the results by just considering each element as a separate computing component" [5]. This means that if an idea for a new component arises, we can simply perform the necessary checks to ensure that device performs its operation without having to consider the QCA cell's operation as fundamentally different than in other contexts. This scalability saves lots of time that would have otherwise been required for simulation and testing.

Both metal-dot and molecular-dot QCA cell implementations have been explored by researchers. Molecular QCA cells offer some advantages over the previous metal-dot implementations. One major factor is that they could operate at room temperature, as opposed to the frigid temperatures previously required [2] [4] while supporting electron transfer rates (clock speeds) on the order of 10^{12} per second [6]. Due to the size differences, the molecular QCA implementation could offer much greater density. The junctures for the metal-dot QCA cell are roughly 60 nm by 60 nm [2] whereas the molecular implementation would be smaller than 1 nm [6].

Introduction to Stochastic Computing

Stochastic computing is a method of performing operations which uses bit streams having particular probabilities to represent its operand values. A stochastic

value s can hold a value $s \in [0.00, +1.00]$. Here probability refers the percent of bits in the stream which are 1's: thus the stream $\{0, 0, 0, 1\}$ represents the value $s = 0.25$, as does the stream $\{0, 0, 1, 1, 0, 0, 0, 0\}$. These stochastic values are used as the inputs to a stochastic operator, and the operator outputs another stochasatic value. See Figure 8 for a simple example use of a stochastic multiplier.

A stochasitc value can be represented by a bit string of any size, but bit string length comes with a tradeoff. Longer bit strings allow for more precise calculations while consuming more time and energy. Using a 2-bit value may work for a limited set of problems, but it limits the number of values that can be represented. A value represented as an n-bit string can represent 2^n distinct values, so clearly a longer string increases flexibility in the values that can be represented.

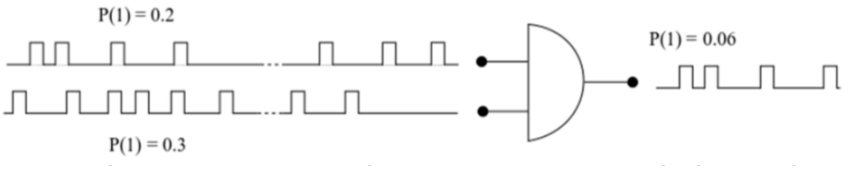


Figure 8: Stochastic multiplier example. Here, input 1 is the stochastic value $s1 = 0.2$ and input 2 is $s2 = 0.3$. As expected, the output value is the expected product, $s3 = 0.06$. For stochastic processes, longer input values lead to more precise calculations while requiring more time and energy.

It is important to consider the energy costs of generating all the random numbers required for a stochstic comupting system. It turns out that creating the input values for these systems is not trivial, and that it may even offset the energy savings stochastic computing could provide over conventional computing. Because a stochastic adder, which is a single OR gate, and a stochastic multplier, which is a single AND gate require very minimal hardware, the processing takes much less energy than conventional methods [7]. However, generating random numbers with conventional random number generators uses a lot of energy, to the point that the net energy savings is reduced or nonexistent [7]. Considering the energy of both random number

generation and stochastic processing, a stochastic multipler only consumes less energy than conventional processing methods for up to 3-bit precision. However, ignoring the energy required to generate stochastic inputs, the stochastic multiplier cell uses less energy than conventional methods for up to 12-bit precision numbers [7]. This is very promising for future work on stochastic processing, if the energy requirements for generating the random number inputs could be reduced.

Conclusion

The subsequent work focuses on the probabilistic nature of a single QCA cell and how this can be manipulated to implement a tunable hardware random number generator. A low energy method of fast random number generation will be of use in stochastic computing, as well as many other applications including cryptography and quantum communications.

CHAPTER TWO

Theory

Creating a tunable random number generator using a coupled pair of quantum dots or "DQD" is possible because there are particular parameters which can be used to affect the probabilities of measuring a $|0\rangle$ or a $|1\rangle$. These parameters are the hopping or tunneling energy γ and the bias Δ. The tunneling energy is a measure of how much energy is required to move from one state to the other, and the bias is a measure of how much one state is preferred over the other. By tuning these parameters, one alters the energetics of the DQD such that the probability of the cell changes. By doing this dynamically and measuring the cell repeatedly, tunable random number generation can be accomplished.

First, we must cover the mathematics of determining a cell's state. The charge basis of a DQD is defined as $B = \{|0\rangle, |1\rangle\}$. A DQD's state may be, in accordance with quantum mechanical theory, any complex superposition $|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle$. However, a measurement of a DQD will yield only a "0" or "1" which is reduced from the state $|\psi\rangle$. However, due to the probabalistic nature of quantum systems, 2 distinct measurements of the DQD will not necessarily return the same answer. The probability of measuring a particular state is expressed as:

$$p(k) = |c_k|^2 = c_k^* c_k \tag{2}$$

The Hamiltonian is a helpful way of characterizing the energetics of a system. The matrix contains information on the energies of each state, and the transition energy required to change states. We can use a Hamiltonian to characterize the energetics of a system of QCA dots interacting with each other. The Hamiltonian of a 4-dot cell is expressed as:

$$\hat{H} = -\gamma\hat{\sigma_x} + \frac{\Delta}{2}\hat{\sigma_z} = -\gamma \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + \frac{\Delta}{2} \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -\frac{\Delta}{2} & -\gamma \\ -\gamma & \frac{\Delta}{2} \end{bmatrix} \tag{3}$$

where γ is the tunneling energy between basis states, and Δ is the detuning, σ_x and σ_z are Pauli operators:

$$\begin{aligned} \sigma_x &= |1\rangle\langle 0| + |0\rangle\langle 1| \\ \sigma_z &= |1\rangle\langle 1| - |0\rangle\langle 0| \end{aligned} \tag{4}$$

The solutions to the time-indepedent Schrodinger equation are the eigenstates $|\phi_n\rangle$ with their corresponding eigenenergies E_n, where $n \in 1, 2$ and $E_1 < E_2$.

$$\hat{H}|\phi_n\rangle = E_n|\phi_n\rangle \tag{5}$$

E_1 is used to represent the ground state eigenenergy, and the eigenstate $|\phi_1\rangle$ represents the ground state. Likewise, the state $|\phi_2\rangle$ denotes the excited state with energy E_2. It can be shown that:

$$E_1 = -\frac{1}{2}\sqrt{4\gamma^2 + \Delta^2} \tag{6}$$

and:

$$|\phi_1\rangle = \frac{1}{\sqrt{\alpha^2 + 1}}(\alpha|0\rangle + |1\rangle), \tag{7}$$

using:

$$\alpha = \frac{\Delta + \sqrt{4\gamma^2 + \Delta^2}}{2\gamma}. \tag{8}$$

Given a system relaxed into the ground state $|\phi_1\rangle$, a position measurement will randomly yield a state $|\psi\rangle \in \{|0\rangle, |1\rangle\}$ with the probabilities in Equation 9. A graphical representation of these probabilities is shown in Figure 9 for $\Delta \in [-1, +1]$.

$$p(|0\rangle) = \frac{\alpha^2}{\alpha^2 + 1} \qquad \text{and} \qquad p(|1\rangle) = \frac{1}{\alpha^2 + 1} \tag{9}$$

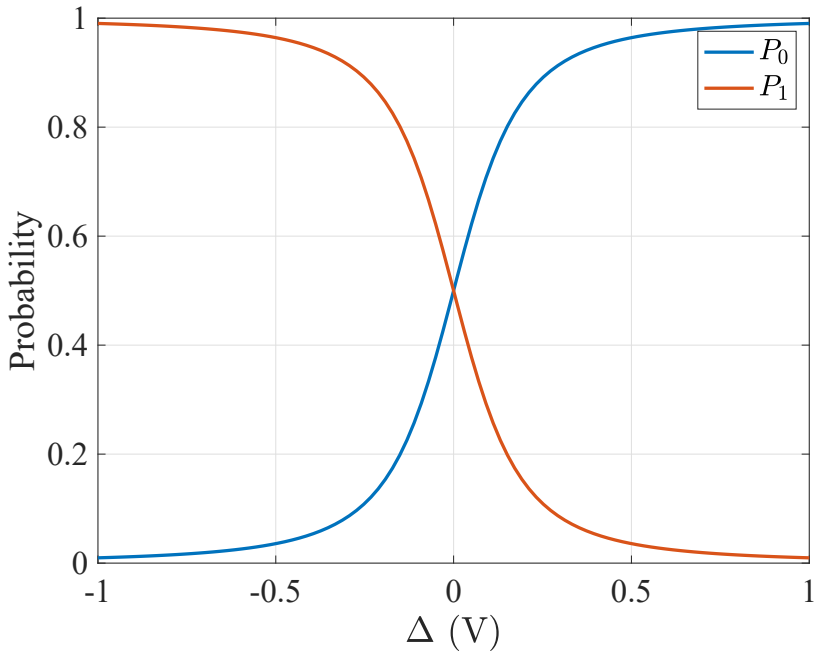


Figure 9: This curve shows how the probability of measuring each state as the bias is adjusted. This principle is what allows easy tuning of the measurement probabilities.

Importantly, these probability equations depend on Δ, a tunable driver polarization. This means that the probability of a cell, the probability that a given measurement will yield $|1\rangle$, can be dynamically modified. Therefore, by performing some number N consecutive measurements on a DQD an N-bit random number can be assembled by placing one randomly generated bit in a string after the previously measured bit.

A QCA based random number generator requires no software post processing. This means the energy used to generate a stochastic number is only the energy required to bring the cell from its ground state $|\phi_1\rangle$ to its measured state. Each measurement projects the system onto one of the basis states, either $|0\rangle$ or $|1\rangle$. The

average energy required to generate a random bit is:

$$E_{avg} = \frac{2\gamma^2}{\sqrt{4\gamma^2 + \Delta^2}}, \tag{10}$$

and the total power required to generate bits at a rate of N bits/s is:

$$\begin{aligned} P_{tot} &= E_{avg} N \\ &= \frac{2N\gamma^2}{\sqrt{4\gamma^2 + \Delta^2}}. \end{aligned} \tag{11}$$

Equation 11 was graphed for $\gamma = 0.03$ eV and for a range of $\Delta = [-1, 1]$ eV. The resulting graph is shown below in Figure 10. This figure illustrates that the worst case scenario for power consumption occurs at $\Delta = 0$. From this realization a general expression for the max power required to generate bits at a rate of N bits/s was found, and is shown in Equation 12.

$$P_{tot}(\Delta = 0) = N\gamma \tag{12}$$

For generating a string of random bits, it will likely be preferable to use repeated measurements of a single DQD to generate a random bit sequence in a serial method. Here, one quantum device is associated with each stochastic input. The alternative is a parallel method, which would require a massive network of interconnections from a large number of DQDs to a single logic gate input. Because generation of a bit string will be a serial operation, the relaxation time T_1 for the quantum system will play an important role in defining the bit rates possible for generating a stochastic bit sequence as well as setting the time scale for single measurement operations. Figure 11 show the timing scheme required for generating random bits from the DQD.

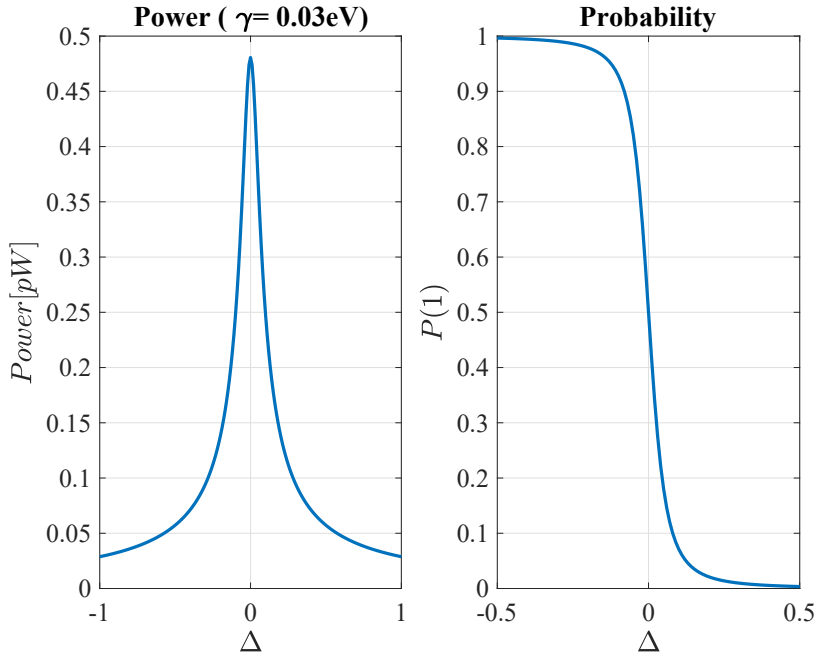


Figure 10: Average power dissipation and probability shown as a function of Δ with $\gamma = 0.03$ eV. The maximum power is consumed for Δ=0, which corresponds to $p(1)$=0.5. This maximum power for a rate $N = 10^8$ bits/s is 480.6 fW.

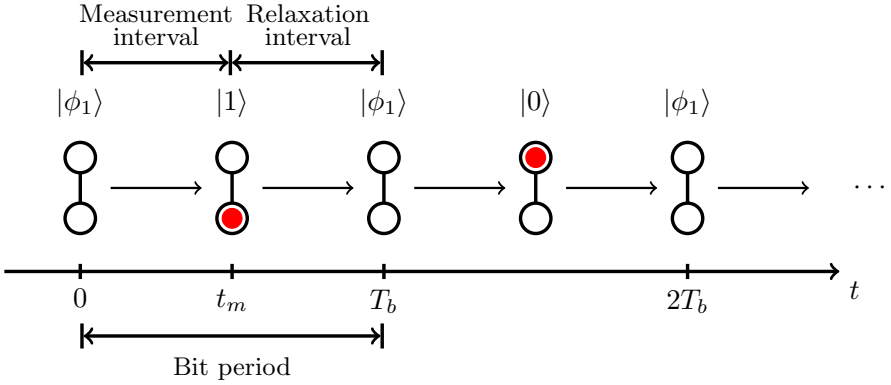


Figure 11: Showing timing of measurement then relaxing for generating bits with a QCA cell. This is the process for generating just 2 bits, but the cycle can go on for any desired number of bits.

CHAPTER THREE

Device Implementation

Quantum-dot Cellular Automata (QCA) cell implementation is an active area of analysis with lots of open questions. These questions include determining optimal qualities of a good candidate molecule, how the state of a QCA cell can be read, and how sample reading must be spaced out in time. Systems of quantum dots with localized charge states have been fabricated and tested for application in a low-power, classical computing paradigm known as Quantum-dot Cellular Automata [8] [9]. QCA Devices have been implemented using both metallic quantum dots [2] [10], and using semiconductor devices [2]. Molecular cells offer certain benefits over other options, particularly operating temperature and size. State read-out may be achieved by measuring the charge state using single-electron transistors (SET) electrometers.

Single Electron Transistor Readout

Bit read-out may be done using SET electrometers, which have demonstrated sensitivity to sub-nanometer displacements of single electrons [11]. This high level of sensitivity makes it a great tool for tiny systems such as QCA cells. An SET can be useful in taking measurements of any quantum system which has two distinct charge states [12]. Since a QCA cell has a defined basis of $|\psi\rangle \in \{|0\rangle, |1\rangle\}$, the SET is a good fit for this application. An important question to consider is whether the SET will be able to measure within an appropriate amount of time to avoid causing other problems. Determining how long a measurement should take, and how long to wait between consecutive measurements, is a challenging task and finding answers to these questions will require future research.

The Quantum Zeno effect can help inform the optimal measurement time for a quantum mechanical system. This effect states that there exists "inhibition of transitions between quantum states by frequent measurements" [13]. This is to say that if measurements are taken too frequently, the system will not have time to evolve after being collapsed during the last measurement, resulting in continuous measurement. A continuously measured state can never decay [13]. So, if you know the state of the system at all times, it will never change.

Metal-dot Cell Implementation

Capacitively coupled metal dot QCA cells yield "experimental results show excellent agreement with theory" [2]. The great accomplishment of building real-life QCA celles demonstrated that QCA computing devices are attainable. However, this implementation does have its drawbacks. For example, testing was conducted at a temperature of 15 mK [2], because higher temperatures would disturb the proper operation of the cells. Some of the issues that arise due to the small size of the cell are "quantum effects and nondeterministic behavior of small current" [14].

Metal-dot QCA devices have been fabricated and used to gather data [2]. In these experiments, the device under test consisted of four Aluminum (Al) islands having two input dots D1 and D2 and two output dots D3 and D4. Two $Al - AlO_x - Al$ tunnel junctions were contructed on a Silicon substrate using electron beam-lithography and shadow evaporation creation techniques to allow electrons to travel between the dots. This arrangement is shown below in Figure 12. In a lithographic implementation, the detuning Δ may be achieved by directly applying a voltage between dots 1 and 0: $\Delta = qV$, where q is the mobile charge, and V is the applied voltage.

Important considerations for comparing the metal-dot implementation covered in Orlov's research and possible molecular implementations are cell area, operating

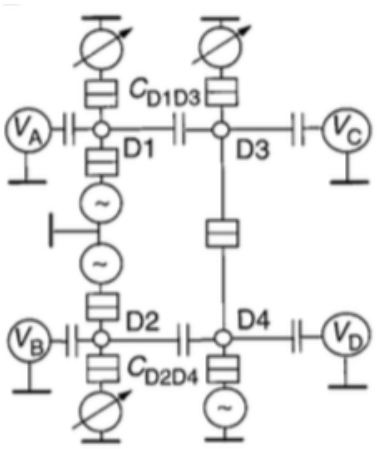


Figure 12: Metal-Dot QCA Implementation [2]

temperature, and switching speed. The area of the junctions in a metal dot cell is "about 60 nm by 60 nm," and the base temperature of the cell is 15 mK at the time of the experiments performed [2]. For a metal-dot QCA implementation a damping rate of 0.15 GHz is obtained [15] [16] using with intercell coupling $E0 = 0.62$ meV. This damping rate tells us that a metal-dot QCA has a relaxation time of 6.7 ns, which in turn gives us back a possible clocking rate of 1.5 GHz.

Molecular-dot Cell Implementation

Molecular DQDs have been conceived for applications such as molecular charge qubits [17] and room temperature, low-power classical computing devices known as quantum-dot cellular automata [8] [18]. Such molecular-dot QCA systems could support clocking rates "well beyond the GHz range" [6]. Here, a single mixed-valence molecule provides a coupled pair of dots, with redox centers functioning as quantum dots. In one system proposed for biasing a DQD using an applied electric field, [19] the detuning is

$$\Delta = -q_e \vec{E} \cdot \vec{a}. \tag{13}$$

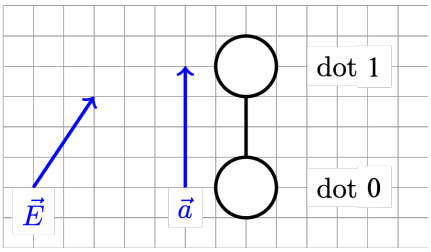


Figure 13: Visualization of DQD biasing. By applying $\hat{E}$ to the DQD, we can cause one state to be naturally preferred over the other. This is what is meant by biasing a cell.

Here $q_e > 0$ is the fundamental charge, $\vec{E}$ is the applied biasing electric field, and $\vec{a}$ is the vector of length a in a direction pointing from dot 0 to dot 1 as in Figure 13. Here, $|\vec{a}|$ is the distance between the coupled dots, and for a differocenyl acetylene (DFA) molecule this distance is a mere $a = 0.67$ nm [6]. For a DQD based on an ionic DFA, it has been calculated that relaxation times under a field-driven bias are $T_1 = 1$ ps [20]. Taking $T_s = 10$ ps, it will be possible to generate stochastic bits at a rate of 100 Gb/s. It is also important to keep sample times of the time scale $\Delta t_m < 1ps$.

Conclusion

As stated above, the molecular implementation has a notable advantage in regards to size, having a distance of just 0.62 nm rather than the 60 nm distance between dots in the metal-dot implementation. Also, the molecular implementation supports room temperature operation, while the metal-dots require cryogenic cooling. Finally, the molecular implementation supports THz range switching speeds, while metal dots do not even keep up with current semiconductor device speeds. The molecular-dot cells are clearly better on all accounts, other than the ease of implementation. Current lithographic techniques can produce metal-dot cells, while further work is required to create molecular cells.

CHAPTER FOUR

Discussion

QCA is a technology that could enable electronics to continue getting faster, smaller, and more efficient after semiconductor transistor based devices cannot continue shrinking. It can be used to implement a logically complete set, which means that it is suitable for designing all types of digital circuits. It also has the unique ability to act as a random number generator, due to the quantum mechanical behavior inherent in its operation. This random number generation implementation will use far less power than traditional methods, and it requires no post-processing in software.

A molecular QCA based random number generator can generate random bits at a rate of 100 Gb/s or higher. The maximum power required to generate bits at a rate of 100 million bits per second is just 480.6 fW. This maximum occurs when $\Delta = 0$ which corresponds to a probability $p(|1\rangle) = 0.5$. The minimum power, which occurs when $\Delta = -1$ and $\Delta = 1$ is just 28.8 fW. This device would meet the need for lower energy random numbers, which would help a variety of applications including cryptography [21], stochastic computing, and quantum communications protocols like BB84 [22].

Further Questions

There are several questions which must be further explored before these systems can be implemented to their full potential. This research is complemented by chemists whose research is instrumental in finding molecules which exhibit the correct properties to act as QCA cells. Such a molecule should have 2 redox centers, should

have 1 free electron, should be small in size, and should be easy to create. Becuase the power required to generate a bit string is directly related to hopping energy γ, which is shown in Equation 12, it will be helpful to keep this value low. Some of the more specific properties of an ideal candidate molecule are currently being explored.

An important step to take is to physically realize this system. There is incredible value in being able to gather data and information from a hardware implementation of this theory to understand how well our models truly describe the system. A reliable method for fabricating cells using such small molecules is not yet known. Some promising work has been done using DNA tiling, which may be used to create molecular QCA devices in the future [24].

Another important step to take in the future is to consider how the DQD random number generator would interface with the stochastic processor. This should be done efficiently because as more buffering and processing are included in the system, more latency is introduced. This interfacing could be done by using all QCA logic. QCA cells can implement the three basic logic operations, as explored in Chapter 1, so the digital logic gates needed for the stochastic computing could be implemented directly using QCA. In this way the need for measuring input values would be eliminated, and only the system's output value would need to be measured.

The power consumption of QCA should be further explored to determine the marginal benefit of using QCA for RNG compared to traditional methods. Analyzing the power usage of each method can help to develop a full comparison. Power analysis of QCA has been explored by Lent [25], and an equation for the power consumed due to an arbitrary bit rate is provided in Equation 11, but further exploration is needed.

BIBLIOGRAPHY

[1] E. P. Blair and S. Koziol, "Neuromorphic computation using quantum-dot cellular automata," in *Rebooting Computing (ICRC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1–4.

[2] A. Orlov, I. Amlani, G. Bernstein, C. Lent, and G. Snider, "Realization of a functional cell for quantum-dot cellular automata," *Science*, vol. 277, no. 5328, pp. 928–930, 1997.

[3] D. H. McIntyre, C. A. Manogue, and J. Tate, *Quantum mechanics: a paradigms approach*. Pearson Boston, 2012.

[4] E. P. Blair and C. S. Lent, "Quantum-dot cellular automata: an architecture for molecular computing," in *Simulation of Semiconductor Processes and Devices, 2003. SISPAD 2003. International Conference on*. IEEE, 2003, pp. 14–18.

[5] P. D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata," *Journal of Applied physics*, vol. 75, no. 3, pp. 1818–1825, 1994.

[6] E. P. Blair, "Quantum-dot cellular automata: A clocked architecture for high-speed, energy-efficient molecular computing," in *International Conference on Unconventional Computation and Natural Computation*. Springer, 2017, pp. 56–68.

[7] J. M. de Aguiar and S. P. Khatri, "Exploring the viability of stochastic computing," in *2015 33rd IEEE International Conference on Computer Design (ICCD)*, Oct 2015, pp. 391–394.

[8] C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernstein, "Quantum cellular automata," *Nanotechnology*, vol. 4, no. 1, p. 49, 1993.

[9] C. S. Lent and P. D. Tougaw, "A device architecture for computing with quantum dots," *Proceedings of the IEEE*, vol. 85, no. 4, pp. 541–557, 1997.

[10] A. O. Orlov, R. Kummamuru, R. Ramasubramaniam, C. S. Lent, G. H. Bernstein, and G. L. Snider, "Clocked quantum-dot cellular automata shift register," *Surface Science*, vol. 532, pp. 1193–1198, 2003.

[11] G. Karbasian, A. O. Orlov, A. S. Mukasyan, and G. L. Snider, "Single-electron transistors featuring silicon nitride tunnel barriers prepared by atomic layer deposition," in *Ultimate Integration on Silicon (EUROSOI-ULIS), 2016 Joint International EUROSOI Workshop and International Conference on.* IEEE, 2016, pp. 32–35.

[12] A. Shnirman and G. Schön, "Quantum measurements performed with a single-electron transistor," *Phys. Rev. B*, vol. 57, pp. 15 400–15 407, Jun 1998. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.57.15400

[13] W. M. Itano, D. J. Heinzen, J. Bollinger, and D. Wineland, "Quantum zeno effect," *Physical Review A*, vol. 41, no. 5, p. 2295, 1990.

[14] J. I. Reshi, M. T. Banday, and F. A. Khanday, "Sequential circuit design using quantum dot cellular automata (qca)," in *2015 International conference on advances in computers, communication and electronic engineering, 1 (1): 143-148, 2015.*{*ISSN: 978-93-822*}, 2015.

[15] G. Tóth and C. S. Lent, "Quantum computing with quantum-dot cellular automata," *Physical Review A*, vol. 63, no. 5, p. 052315, 2001.

[16] A. Y. Smirnov, N. J. Horing, and L. G. Mourokh, "Relaxation to a bistable state in a quantum cell," *Journal of Applied Physics*, vol. 87, no. 9, pp. 4525–4530, 2000.

[17] C. Mujica-Martinez, P. Nalbach, and M. Thorwart, "Organic π-conjugated copolymers as molecular charge qubits," *Physical review letters*, vol. 111, no. 1, p. 016802, 2013.

[18] C. S. Lent, "Bypassing the transistor paradigm," *Science*, vol. 288, no. 5471, pp. 1597–1599, 2000.

[19] E. P. Blair, "Electric-field inputs for molecular quantum-dot cellular automata circuits," *arXiv preprint arXiv:1805.04029*, 2018.

[20] E. P. Blair, S. A. Corcelli, and C. S. Lent, "Electric-field-driven electron-transfer in mixed-valence molecules," *The Journal of Chemical Physics*, vol. 145, no. 1, p. 014307, 2016.

[21] L. Kocarev, "Chaos-based cryptography: a brief overview," *IEEE Circuits and Systems Magazine*, vol. 1, no. 3, pp. 6–21, 2001.

[22] D. Gottesman, H.-K. Lo, N. Lutkenhaus, and J. Preskill, "Security of quantum key distribution with imperfect devices," in *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.* IEEE, 2004, p. 136.

[23] M. Barangi, J. S. Chang, and P. Mazumder, "Straintronics-based true random number generator for high-speed and energy-limited applications," *IEEE Transactions on Magnetics*, vol. 52, no. 1, pp. 1–9, 2016.

[24] J. Liu, Y. Geng, E. Pound, S. Gyawali, J. R. Ashton, J. Hickey, A. T. Woolley,

and J. N. Harb, "Metallization of branched dna origami for nanoelectronic circuit fabrication," *Acs Nano*, vol. 5, no. 3, pp. 2240–2247, 2011.

[25] J. Timler and C. S. Lent, "Power gain and dissipation in quantum-dot cellular automata," *journal of applied physics*, vol. 91, no. 2, pp. 823–831, 2002.