Quantum-Dot Cellular Automata

Final Project: Multiprocessor Architecture

Barcelona, June 29th, 2020

Jake Watson

Contents

1 Introduction

For 60 years, advances in computing have been driven by the relentless miniaturisation of *complementary metal oxide semiconductor (CMOS)* technology. Doubling the transistor density led to a 40% increase in performance in each generation, a trend known as Moore's law. This trend began in the 1970s and has continued despite predictions of its failure, becoming integral to the health of the semiconductor industry.

However, we may be witnessing the final generations of the trend, as transistor dimensions dip below 10 nm. At these scales, quantum mechanical effects dominate, and the operation of CMOS devices becomes unpredictable. *Quantum-dot cellular automata (QCA)* is one approach proposed to replace CMOS. The technology promises high circuit density, low power dissipation, and high speed. This report presents an overview of this technology, including its logical foundations, and its physical implementation.

1.1 CMOS Challenges

Computing advances have consisted of continued reduction in scale of the *metal-oxide semiconductor field-effect transistors (MOSFET)*. First invented in 1959, the MOSFET has been continuously reduced in scale from approximately 20 µm, to below 10 nm in the 2010s. At these scales, several challenges in operation are introduced [1].

- Leakage currents: at small scales, reductions in barrier thicknesses increase the probability of quantum tunnelling of electrons, causing leakage currents.
- Material reliability: as materials such as silicon dioxide become thinner, they increasingly suffer from mechanical and thermal stress, affecting the reliability of the device.
- Power dissipation: with increasing circuit density, the power density also increases. Ever increasing heat generation in the same area leads to enormous temperatures.
- Lithography: as transistor sizes become smaller than the wavelengths of light used to print them, the complexity of the printing process increases. Eventually, the process becomes uneconomical.

1.2 QCA Capabilities

QCA is an emerging nanotechnology promising a replacement for CMOS, relying on quantum mechanical effects to offer faster speed at reduced size and power consumption. In this report, the operation and principles of QCA are presented, followed by a simulation of the standard Boolean logic operations. Two reversible logic gates are also simulated, as well a 1-bit full-adder circuit.

2 QCA Fundamentals

2.1 Cells

The basic unit of this technology is the QCA cell, a square structure with a quantum dot at each corner. Dots can be viewed as containers for electrons. The cell contains two free electrons that move between the dots. Electrons tend to occupy opposite diagonal dots, due to their Coulombic repulsions, but due to tunnelling effects they can shift to the opposite diagonal. These two diagonals represent two bistable states of the cell. The polarization of the cell is defined as [2]

$$P = \frac{(p_1 + p_3) - (p_2 + p_4)}{(p_1 + p_2 + p_3 + p_4)}$$

where p_i represents the electric charge at dot i in the cell. The two values of the polarization can hold binary information.

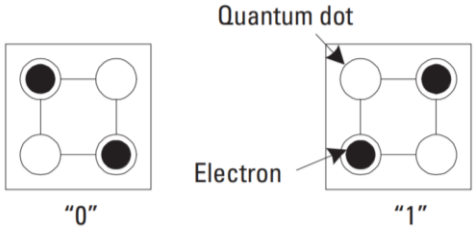


Figure 1: Polarization states of the QCA cell. [2]

2.2 Wires

When cells are placed close to each other, the electrons will tend to align for every cell, through their electrostatic interaction. For a driver cell with fixed polarization, all cells in contact with it will assume its polarization – the value will propagate. This behaviour can be used to construct a 'wire' out of a chain of adjacent cells. To handle wire crossings, we can either rotate the cells in one wire by 45°, or build multi-layer structures.

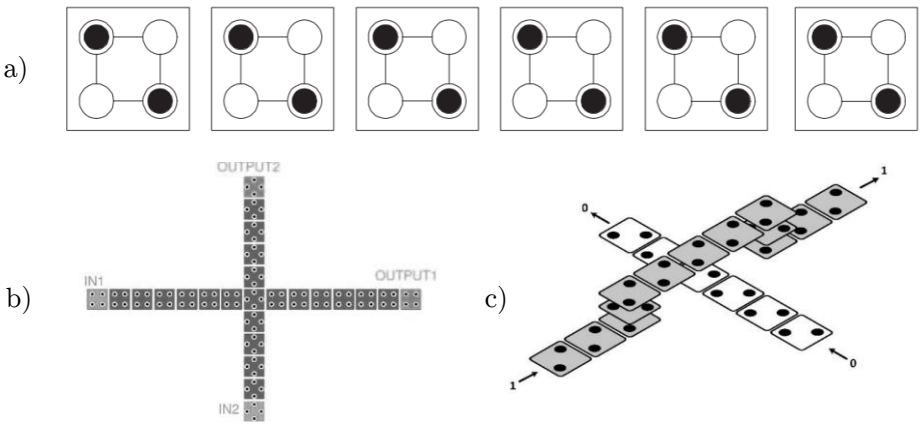


Figure 2: a) QCA wire constructed with chained cells [2] b) rotated cells crossing c) multilayer crossing.

2.3 Clocking

In order to control the flow of information in QCA circuits, a clock signal must be used. The clock operates directly on the cells, raising and lowering the potential barriers that prevent tunnelling between dots. When the barriers are high, no change of state is possible for the cells. It has been shown [3] that 4 clock signals, shifted by 90° to each other, is sufficient to regulate the circuit. The clocking phases are described below.

1. Switch: Tunnelling barriers are raised, and the QCA cells move from un-polarized to polarized state.
2. Hold: Barriers are kept high, and electrons cannot tunnel, so the cells remain polarized.
3. Release: Barriers are lowered.
4. Relax: Electrons begin to tunnel through the lowered barriers, and the cell becomes unpolarized.

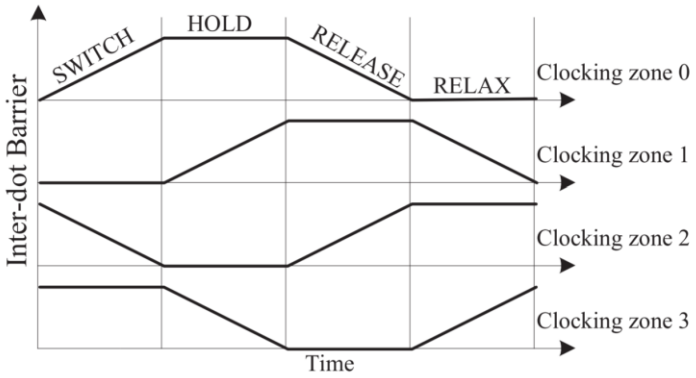


Figure 3: Shifted phases of the clocking signal. [2]

2.4 Physical Implementation

There are four approaches that have been taken to realize physical QCA.

- Metal Island: this approach implements a QCA cell as four µm-scale islands of aluminium, to act as the dots, with tunnel junctions of aluminium oxide between them. A variety of components have been fabricated, but due to the extremely low operating temperature of less than 1 Kelvin, this does not appear to be a practical approach. [4]
- Semiconductor: this approach attempts to use the existing semiconductor fabrication process. Dots are defined as metallic gates, with an electron gas below the surface. This approach has been widely studied due to the availability of existing semiconductor fabrication equipment, but this is also its limitation, as the sizes required to truly exploit QCA are too small for the fabrication technology.
- Molecular: here, the QCA cell is a single molecule, with charge building on certain areas. The molecules could be smaller than 1 nm, with room-temperature operation, enormous density

and high switching speeds possible. However, synthesizing and positioning these molecules remains extremely difficult. [5]

- Magnetic: using nano-metre scale magnets, with the dipole moment representing the binary information, a QCA cell can be constructed. This approach allows for room-temperature operation, with low power dissipation and being resistant to thermal effects, and can be fairly easily fabricated. However, it has low switching speeds, on the order of 100 MHz.

3 QCA Simulation

To illustrate the potential of QCA, several basic logic gates were simulated using a numerical simulation engine called QCADesigner (v. 2.0.3) [6]. This is a popular tool in designing QCA circuits, and supports multilayer systems.

3.1 Bistable Simulation Engine

The Bistable engine is one of several engines used by QCADesigner in modelling QCA circuits. The following description of its operation is taken almost verbatim from the documentation of QCADesigner. The engine assumes that each cell is a simple two-state system, for which the following Hamiltonian can be constructed:

$$H_i = \sum_j \begin{bmatrix} -\frac{1}{2} P_j E_{i,j}^k & -\gamma_i \\ -\gamma_i & \frac{1}{2} P_j E_{i,j}^k \end{bmatrix}$$

Where P_j is the polarization of cell *j*, E^k is the kink energy between cell *i* and *j*, γ_i is the tunnelling energy of electrons in the cell *i*. The summation is over all cells within an effective radius of cell *i*, one of the parameters of the simulation. The kink energy represents the energy cost of two cells having opposite polarization. This can be calculated from the electrostatic interaction between all charges. For each dot in cells *i, j* the kink energy is:

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

Where ϵ_0 is the permittivity of free space and ϵ_r is the relative permittivity of the material of the circuit. Kink energy decays inversely at the power of five of the distance between cells, allowing us to approximate the Hamiltonian with the effective radius. It is assume that the cells remain very close to their ground state in either polarization, so the stationary state of each cell in the circuit can be calculated using the time-independent Schrödinger equation as:

$$H_i \psi_i = E_i \psi_i$$

Where H_i is the Hamiltonian described above, ψ_i is the state vector of the cell, and E_i is the energy of the state. This eigenvalue expression evaluates to the following relation:

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

The engine iteratively computes the polarization of each cell until the circuit converges to a tolerance. This method sacrifices some accuracy for speed of computation, but given the relatively simple circuits in this project, it is sufficient.

3.2 Simulation Parameters

- Convergence Tolerance: 0.001
- Radius of Effect: 65 nm
- Relative Permittivity: 12.9
- Clock High/Low/Shift: 9.8E-23 J / 3.8E-23 J / 0 J
- Clock Amplitude Factor: 2.0
- Layer Separation: 11.50 nm

4 QCA Logic

4.1 Classical Gates

A series of logic gate designs were implemented in QCADesigner. The complete set of simple gates was successfully simulated, as well as the universal NAND gate, showing that QCA circuits can perform any Boolean operation. The different colours of the unlabelled cells represent their clock region.

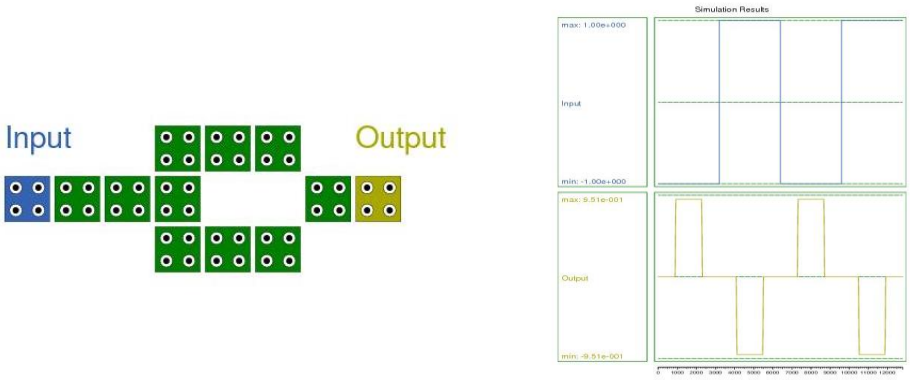


Figure 4: Schematic and output of NOT gate.

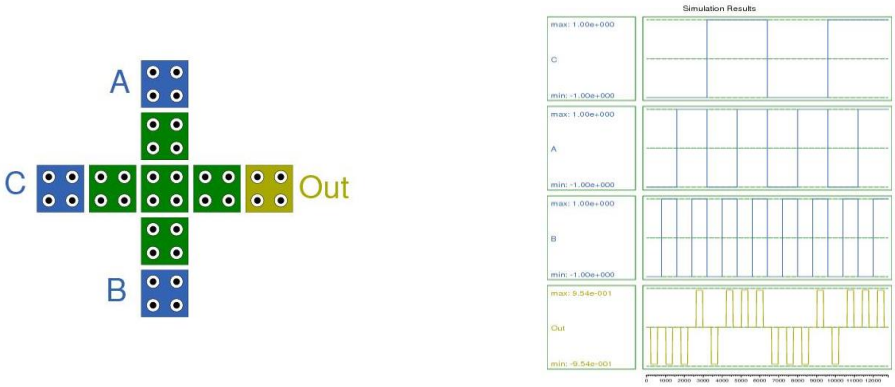


Figure 5: Schematic and output of 3-majority gate.

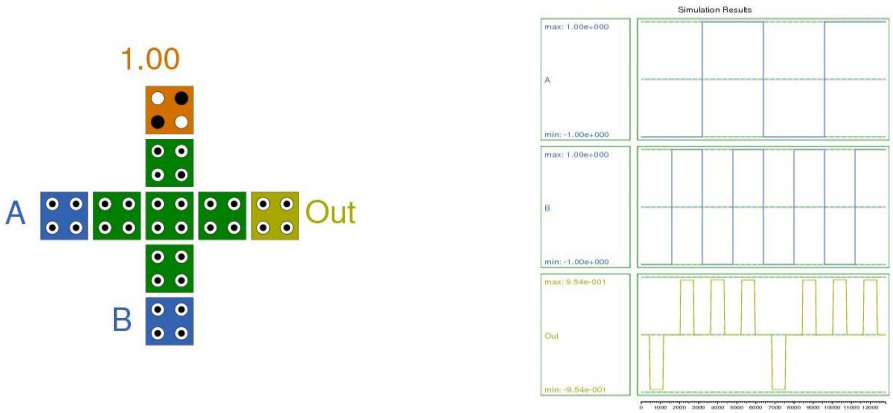


Figure 6: Schematic and output of OR gate.

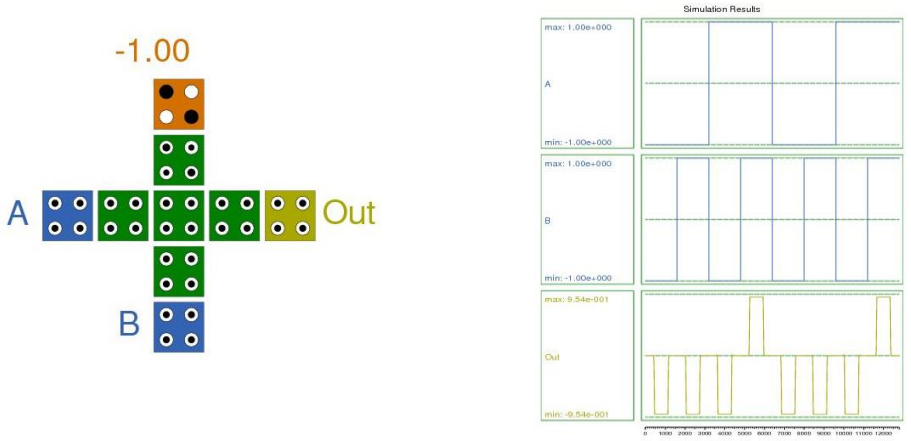


Figure 7: Schematic and output of AND gate.

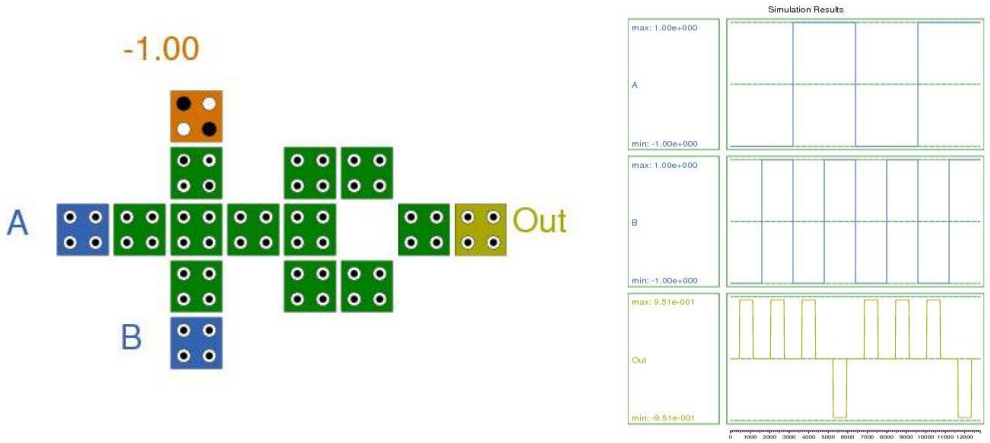


Figure 8: Schematic and output of NAND gate.

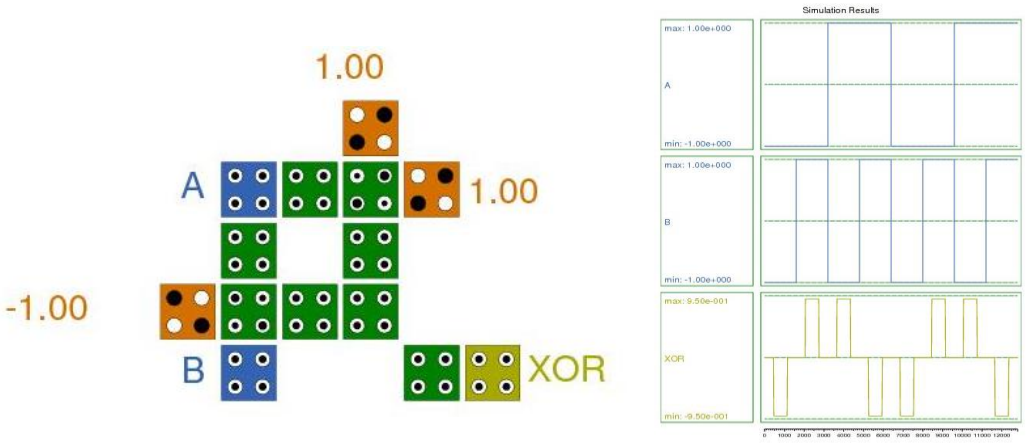


Figure 9: Schematic and output of XOR gate.

4.2 Reversible Computing

Reversible computing is an alternative computational paradigm, first conceived in 1973 by Charles Bennett of IBM. It envisages computational processes which are completely *time-reversible* – the inputs of the process can be reconstructed from the outputs. This paradigm was conceived in response to Rolf Landauer, who first described the link between information in computation, and energy cost [7].

Current modes of computation are *irreversible* – for example, consider the AND gate. Inputs of (0, 0), (0, 1), (1, 0) all map to an output of (0). It is impossible to recreate the inputs given only the output, as the mapping is *many-to-one*. We can view the gate as having consumed its inputs, or erasing them.

Landauer's principle states that the erasure of n bits of information must incur an energy loss of $nKT \cdot ln(2)$, where K is Boltzmann's constant and T is the temperature of the interaction. At room temperature, this is approximately 3E-21 J. This energy loss contributes directly to the inefficiency of the computer performing the operation, leading to higher power consumption and greater temperatures.

To avoid this loss of energy, the circuits making up the computer must be *reversible*. It must be possible to entirely reconstruct the inputs from the outputs. To this end, two reversible logic gates were simulated, as well as a 1-bit full-adder circuit.

4.2.1 Feynman Gate

The Feynman gate takes two inputs and returns two outputs. It is also referred to as a *controlled-NOT (CNOT)* gate. It maps the outputs *one-to-one*, making it logically reversible.

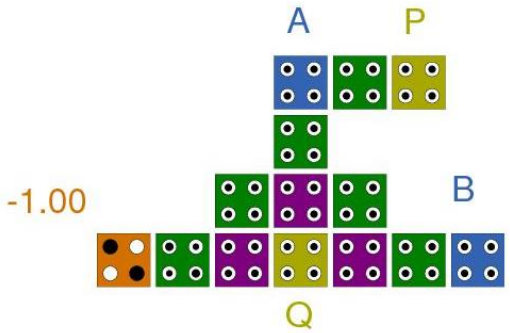
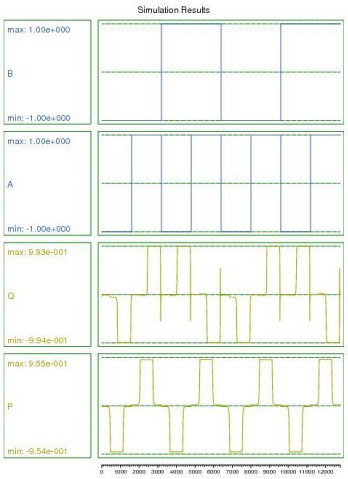


Figure 10: Schematic of Feynman gate.



Input		Output	
A	B	P	Q
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Figure 11: Output and reference truth table for Feynman gate.

4.2.2 Toffoli Gate

The Toffoli gate, also known as a *controlled-controlled-NOT (CCNOT)* gate, is a universal reversible logic gate.

The gate has three inputs and outputs. If the first two inputs are high, it flips the third output. The first two outputs are mapped directly to the first two inputs.

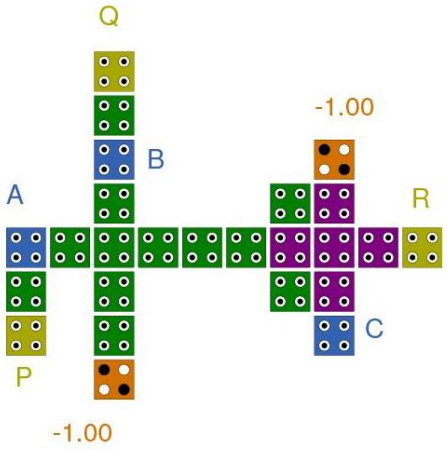
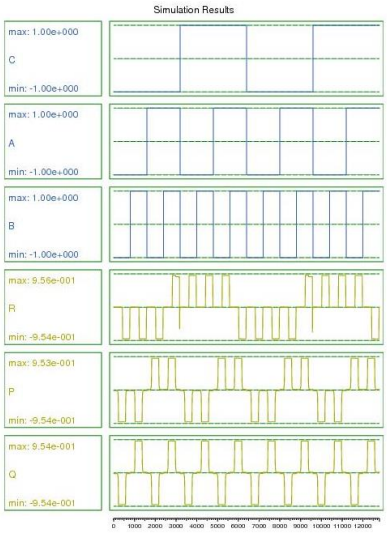


Figure 12: Schematic of Toffoli gate.



Input			Output		
A	B	C	P	Q	R
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Figure 13: Output and reference truth table for Toffoli gate.

4.2.3 1-bit Full Adder

The full adder circuit takes three inputs A and B, and a third input representing the carry-bit as C. The output carry-bit is represented as *Cout*, and the sum as *SUM*. The circuit also provides two *garbage* outputs *Gar1* and *Gar2*, to maintain reversibility. .

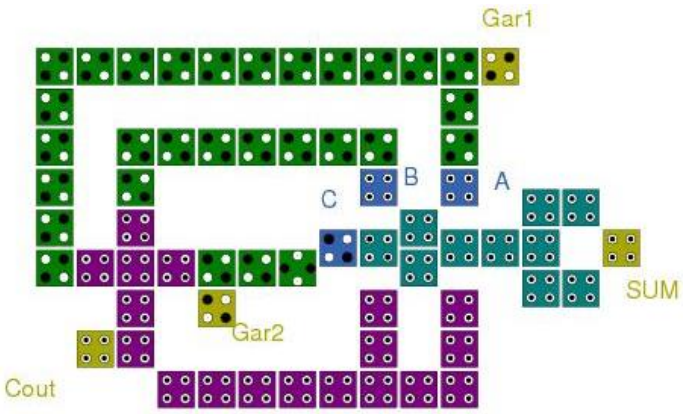
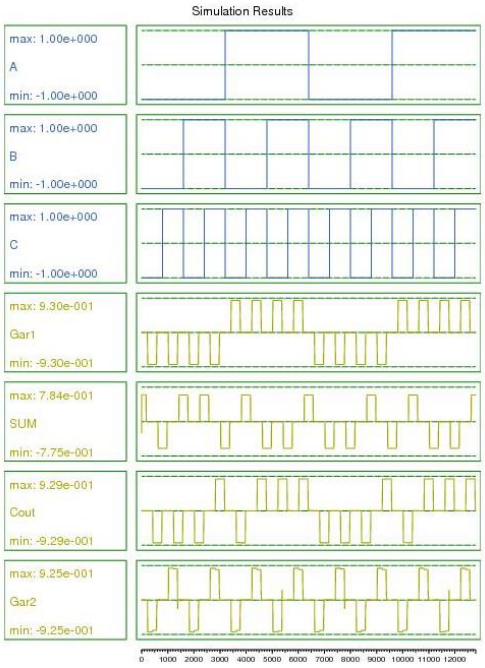


Figure 14: Schematic of reversible 1-bit full adder circuit.



Input			Output			
A	B	C	Sum	Cout	Gar1	Gar2
0	0	0	0	0	0	0
0	0	1	1	0	0	0
0	1	0	1	0	0	1
0	1	1	0	1	0	1
1	0	0	1	0	1	0
1	0	1	0	1	1	0
1	1	0	0	1	1	1
1	1	1	1	1	1	1

Figure 15: Output and reference truth table of 1-bit full adder.

5 Conclusions

This project presents the background and concepts underlying QCA, as a potential alternative to CMOS technologies in computation. Various conventional logic gates were simulated, showing that QCA can perform any Boolean operation. Additionally, two reversible logic gates, and a reversible full-adder circuit were simulated, as reversible computing in QCA would represent close to the ideal in computing. Clearly, given the rapidly-approaching end of Moore's law, QCA presents an attractive avenue of research.

6 References

[1] N. Z. Haron and S. Hamdioui, "Why is CMOS scaling coming to an END?," Delft University of Technology, Delft.

[2] L. Weigiang, S. E. E and O. Maire, Design of Semiconductor QCA Systems, Artech House, 2013.

[3] G. Toth and C. S. Lent, "Quasi-adiabatic Switching for Metal-Island Quantum-dot Cellular Automata," *Journal of Applied Physics,* vol. 85, p. 2977, 1999.

[4] I. Amlani, A. O. Orlov, G. Toth, G. H. Bernstein, C. S. Lent and G. L. Snider, "Digital Logic Gate Using Quantum-Dot Cellular Automata," *Science,* vol. 284, no. 5412, pp. 289-291, 1999.

[5] C. S. Lent, B. Isaksen and M. Lieberman, "Molecular Quantum-Dot Cellular Automata," *J.Am.Chem.Soc,* 2003.

[6] K. Walus, T. Dysart, G. Julien and A. Budiman, "QCADesigner: A Rapid Design and Simulation Tool for Quantum-Dot Cellular Automata," *IEEE Transactions on Nanotechnology,* vol. 3, no. 1, pp. 26-31, 2004.

[7] R. Landauer, "Irreversibility and Heat Generation in the Computing Process," *IBM Journal of Research and Development,* vol. 5, no. 3, 1961.

[8] P. D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata," *Journal of Applied Physics,* vol. 75, p. 1818, 1994.