

Optimizing a Quantum-dot Cellular Automata Device Design Process

G. Lance Millholland, Katherine Olejniczak, P. Douglas Tougaw, Mark. M. Budnik

Mark.Budnik@valpo.edu, Department of Electrical and Computer Engineering, Valparaiso University, USA.

Lance.Millholland@valpo.edu

Katie.Olejniczak@valpo.edu

Doug.Tougaw@valpo.edu

Poster

Modeling and Simulation OR Quantum Devices

Optimizing a Quantum-dot Cellular Automata Device Design Process

Quantum-dot Cellular Automata (QCA) is an emerging nanoscale computing paradigm that uses a novel quantum-mechanical interaction within the device to perform useful binary calculations. [1] Simulations on QCA devices suggest that they will offer orders of magnitude improvements in operational speed, density, and power consumption when compared to traditional CMOS devices. [2] The device that forms the foundation of the QCA computing paradigm is the quantum cell. [3] Each cell is composed of four quantum dots placed at the corners of a square. These four quantum dots are collectively occupied by two electrons, which are allowed to tunnel between adjacent dots within the cell (Fig. 1). Typical spacing between dots is 20 nm, and the spacing between adjacent cells is 60 nm (Fig. 2). Because of the Coulombic repulsion between each cell's two electrons, they tend to align in opposite corners of the cell. This creates two stable states for each cell, with the electrons aligned along the left-leaning diagonal or the right-leaning diagonal. These two states can be used to map a binary 0 and 1, which provides a connection between the electron arrangement at the device level and the binary value it represents. Nearby cells also interact Coulombically, so that neighboring cells tend to align in the same direction, allowing the propagation of signals along a line of cells (Fig. 3). By placing the cells in precise geometric patterns, it is also possible to implement inverters, AND gates, and OR gates. [4] QCA devices have been shown to be capable of supporting both combinational and sequential digital devices ranging from a few cells up to a full CPU (which requires several thousand cells). [5]

The design process for laying out a QCA system has, until now, involved a large amount of manual work: first sketching the device on paper, then determining the precise location of each cell and the color which corresponds to its clock phase, and finally entering this data into an ASCII file by hand. The device characteristics can then be analyzed using MATLAB. Fig. 4, 5, and 6 show the implementation, timing diagram, and QCA layout for a full adder. [6] Presently, designing QCA devices is a long and labor-intensive process that may require several days to complete a single simulation. While MATLAB expedites the analysis of the Coulombic interaction between cells, the layout of the QCA components remains a manual, intensive process.

The creation of our QCA layout tool is intended to minimize the amount of time to set up a device simulation by simplifying the layout process and automating the conversion from the layout to the text-based MATLAB input file format. A GUI allows designers to select cell positions and their appropriate timing (color) cycles (Fig. 7). Upon completion, the tool allows users to automatically generate a layout output file, analogous to a GDS II file, which in turn allows the device to be simulated in MATLAB. The tool was developed with National Instruments's LabVIEW 2012 and uses a grid of simulation cells arranged into a 75x75 matrix. The user may interact with each cell, pressing any one of its four clock cycle (color) buttons (Fig. 7). An expansion feature allows subsequent 75x75 grid "tiles" to be added to the initial 75x75 grid to allow for the development of larger QCA devices.

To demonstrate the tool operating as intended, the full adder in Fig. 4 was recreated to generate the device layout and data file. The data file was simulated in MATLAB (generating its own representation of the layout, see Fig. 8). Part of the simulation results are shown in Fig. 9. The results showed that the new tool was able to successfully reproduce the manual layout process, but in a shorter amount of time.

In summary, the tool that has been developed provides an integral link in the QCA device design process. Experience with multiple QCA devices points to a 75% reduction in the time required to design QCA devices (compared to previous manual methods), enabling the development of larger and more complex circuits.

References

- [1] Craig S. Lent, P. Douglas Tougaw, Wolfgang Porod, and Gary H. Bernstein, "Quantum Cellular Automata," *Nanotechnology* vol. 4, no. 49, pp. 49-57, 1993.
- [2] P. Douglas Tougaw and Craig S. Lent, "Dynamic Behavior of Quantum Cellular Automata," *J. Appl. Phys.*, vol. 80, no. 8, pp. 4722-4736, Oct. 1996.
- [3] Craig S. Lent, P. Douglas Tougaw, and Wolfgang Porod, "Bistable saturation in coupled quantum-dot cells," *J. Appl. Phys.*, vol. 74, no. 5, pp. 3558-3566, Sept. 1993.
- [4] P. Douglas Tougaw and Craig S. Lent, "Logical devices implemented using quantum cellular automata," *J. Appl. Phys.*, vol. 75, no. 3, pp. 1818-1825, Feb. 1994.
- [5] A. Gin, S. Williams, and D. Tougaw, "Hierarchical Design of Quantum Cellular Automata," *J. Appl. Phys.*, vol. 85, no. 7, pp. 3713-3720, April 1998.
- [6] P. Douglas Tougaw and Craig S. Lent, "Logical devices implemented using quantum cellular automata," *J. Appl. Phys.*, vol. 75, no. 3, pp. 1818-1825, Feb. 1994.

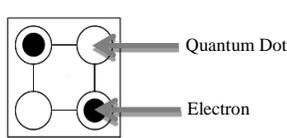


Fig. 1: A QCA cell with its quantum dot and electron positions labeled.

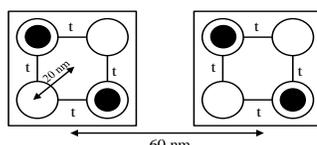


Fig. 2: Two QCA cells and their dimensions.

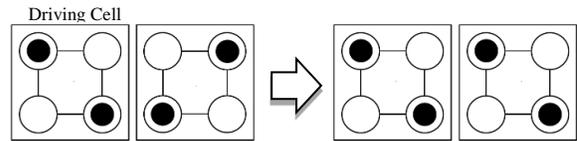


Fig. 3: An example of the relationship between a driver cell and an adjacent cell.

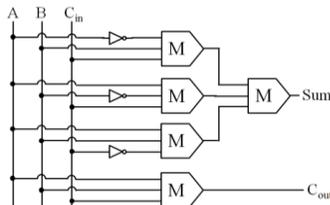


Fig. 4: A full adder circuit to be simulated.

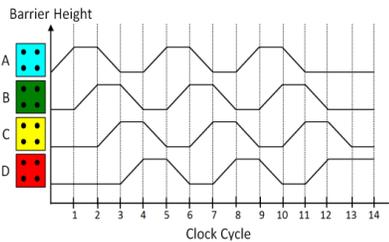


Fig. 5: Clock cycle timing diagram.

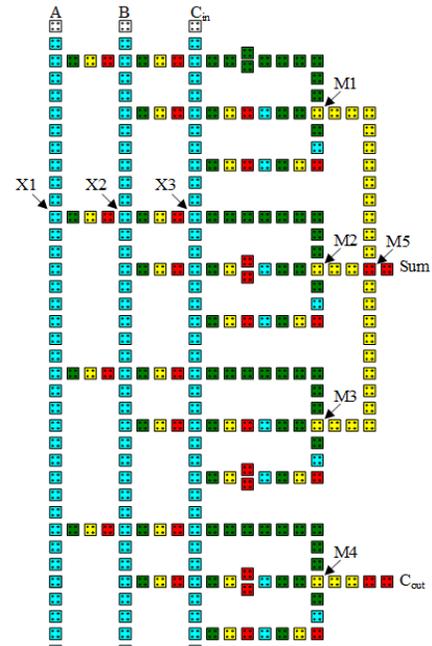


Fig. 6: QCA Layout which implements the logic function in Fig. 4.

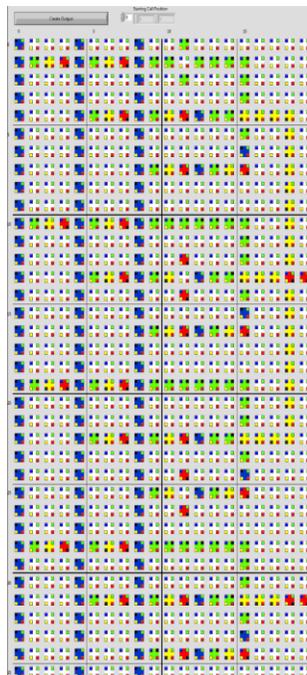


Fig. 7: The layout in Fig. 4 generated using the QCA layout tool.

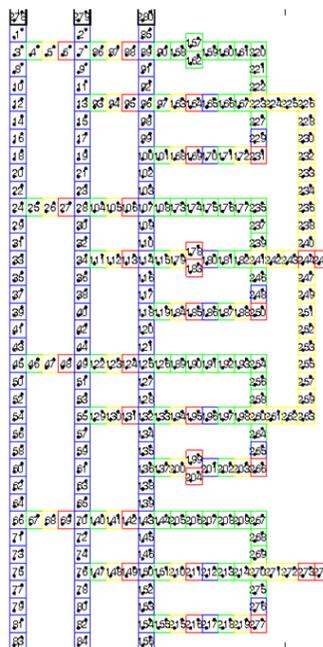


Fig. 8: MATLAB representation of the layout data file.

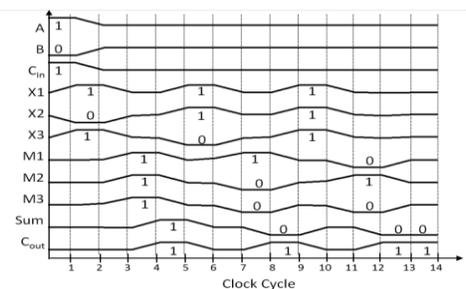


Fig. 9: Actual output from the QCA simulation of Fig. 4.