

Research Goals

- To develop a world-class tool for simulation and design with quantum-dot cellular automata (QCA).
- To advance the state-of-the-art in QCA computing by creating a standard library of QCA circuits.

Quantum-Dot Cellular Automata

Quantum-dot cellular automata is a computing paradigm using arrays of nanostructures called quantum dots (see *What is a Quantum-Dot?* below). QCA takes advantage of quantum phenomena, the same phenomena that may ultimately slow progress in scaling down CMOS circuits.

What is a Quantum-Dot?

Quantum dots are nanostructures created from standard semiconductive materials such as Si/SiO_2 . These structures can be modelled as 3-dimensional quantum wells. As a result, they exhibit energy quantization effects even at distances several hundred times larger than a material's lattice constant.

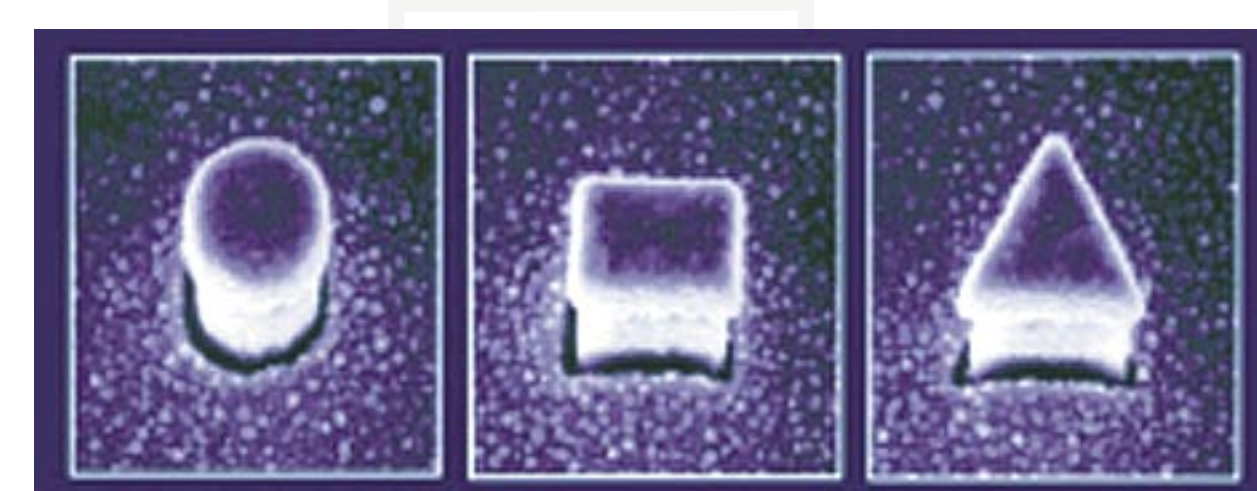


Figure 1 - Example quantum dots

A quantum dot can indeed be visualized as a well. Electrons, once trapped inside the dot, do not alone possess the energy required to escape. We can use quantum physics to our advantage because the smaller a quantum dot is physically, the higher the potential energy necessary for an electron to escape.

How are Quantum Dots Fabricated?

There are several techniques for fabricating quantum dots:

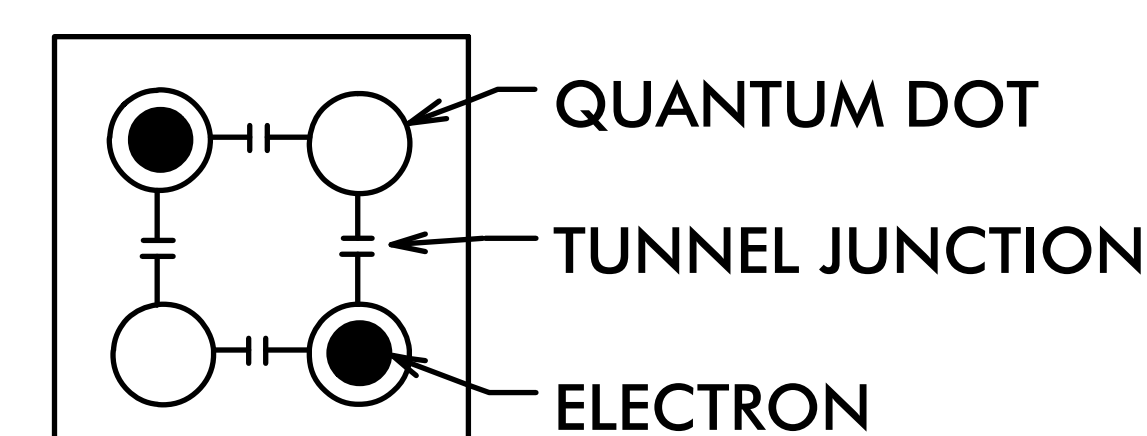
- electron-beam lithography
- self-organization
- formation of depletion bubbles.

Computing With QCA

QCA Computation Cell

Consider a system of four quantum dots arranged in a square pattern as shown.

The tunnel junctions between the dots allow any electron trapped within one dot to tunnel to another dot via quantum tunneling. If we charge the cell with two



electrons as shown they will tend to occupy antipodal dots as a result of Coulombic repulsion. Given that the electrons always occupy antipodal sites, there will be two possible states or polarizations. We can use those two polarizations to encode binary information.

QCA Logic

Coulombic forces will cause adjacent cells to interact. The state of the cells will always tend to the lowest energy. We can take advantage of this to transmit information through a "wire" by forcing the polarization of a cell at one end.

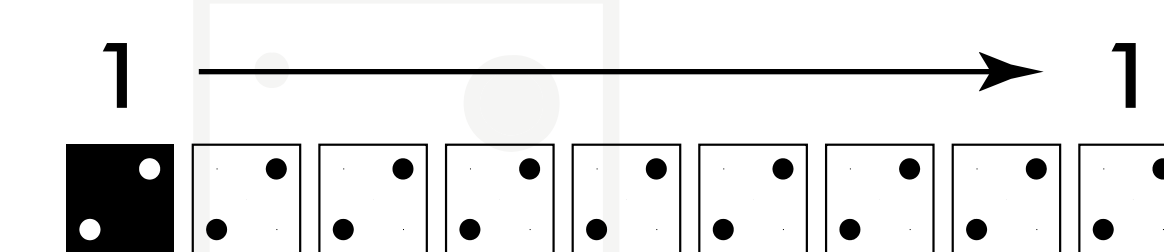


Figure 2 - QCA Wire

By properly arranging the cells any logic function can be created. Consider the arrangements below, demonstrating the QCA implementation of an inverter and a majority gate. By fixing the polarization of one input to a majority gate, we can produce common logic functions, such as the AND gate and OR gate.



Figure 3 - inverter and majority gate

One of the most powerful features of QCA is the ability to create coplanar wire crossings. This makes it possible to implement large designs on a single layer, eliminating the need for complex metal layer interconnects.

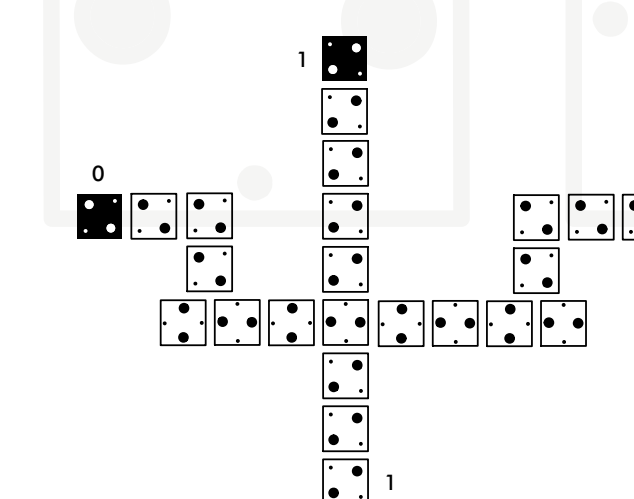


Figure 4 - QCA coplanar wire crossing

With the majority gate, inverter, and wire crossing we can create any digital system. Consider the complete full adder shown here:

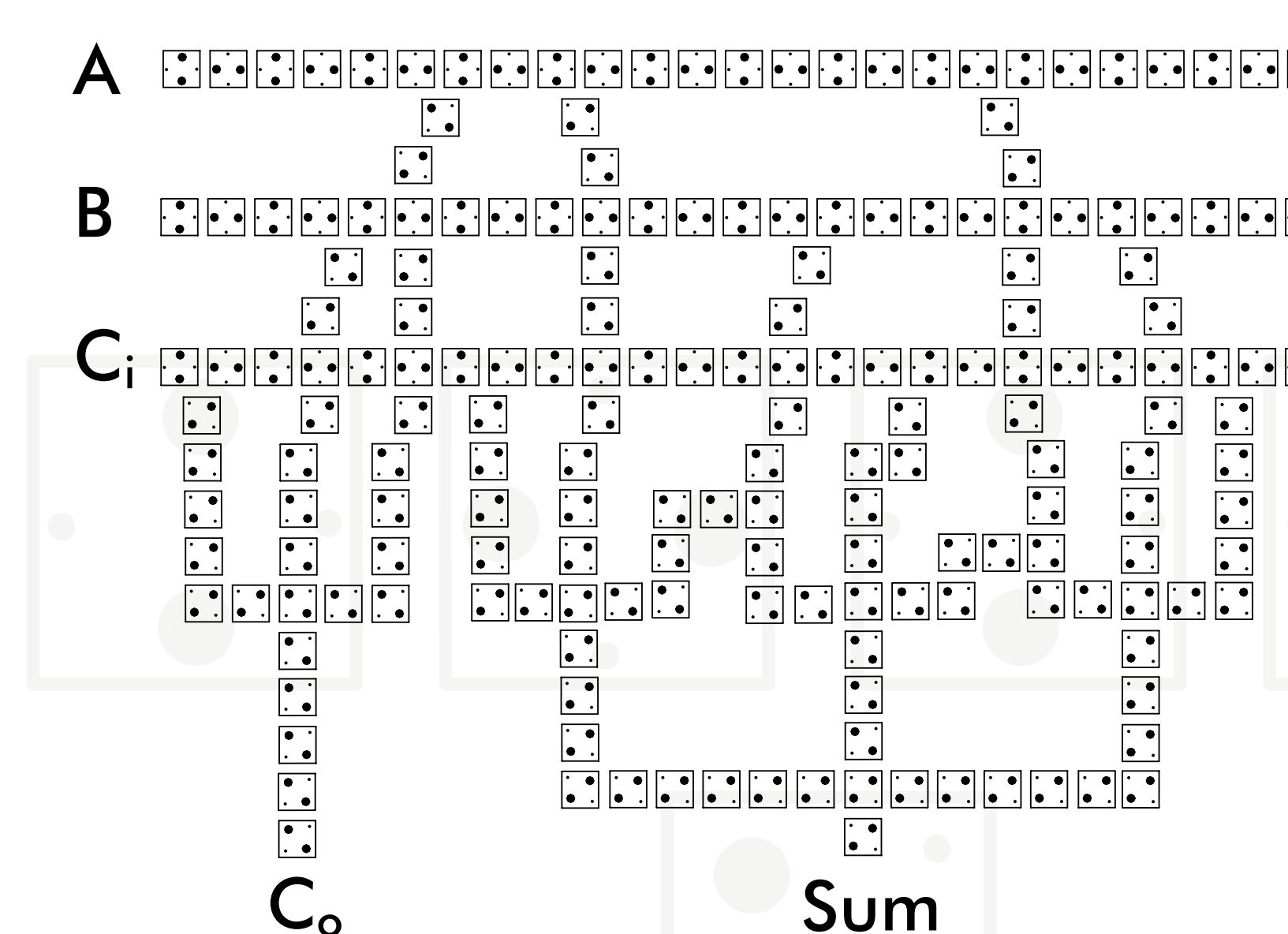


Figure 5 - QCA full adder

QCA Clocking

The clocking of QCA is accomplished by controlling the potential barriers between adjacent quantum-dots. When tunneling potential is high the electron wave functions become de-localized causing indefinite polarization. Raising the potential barrier decreases the tunneling potential, as a result the electrons begin to localize. Once the electrons localize the cell gains a definite polarization. This process, shown below, is completely adiabatic. The only energy expended is that necessary to produce the controlling potentials.

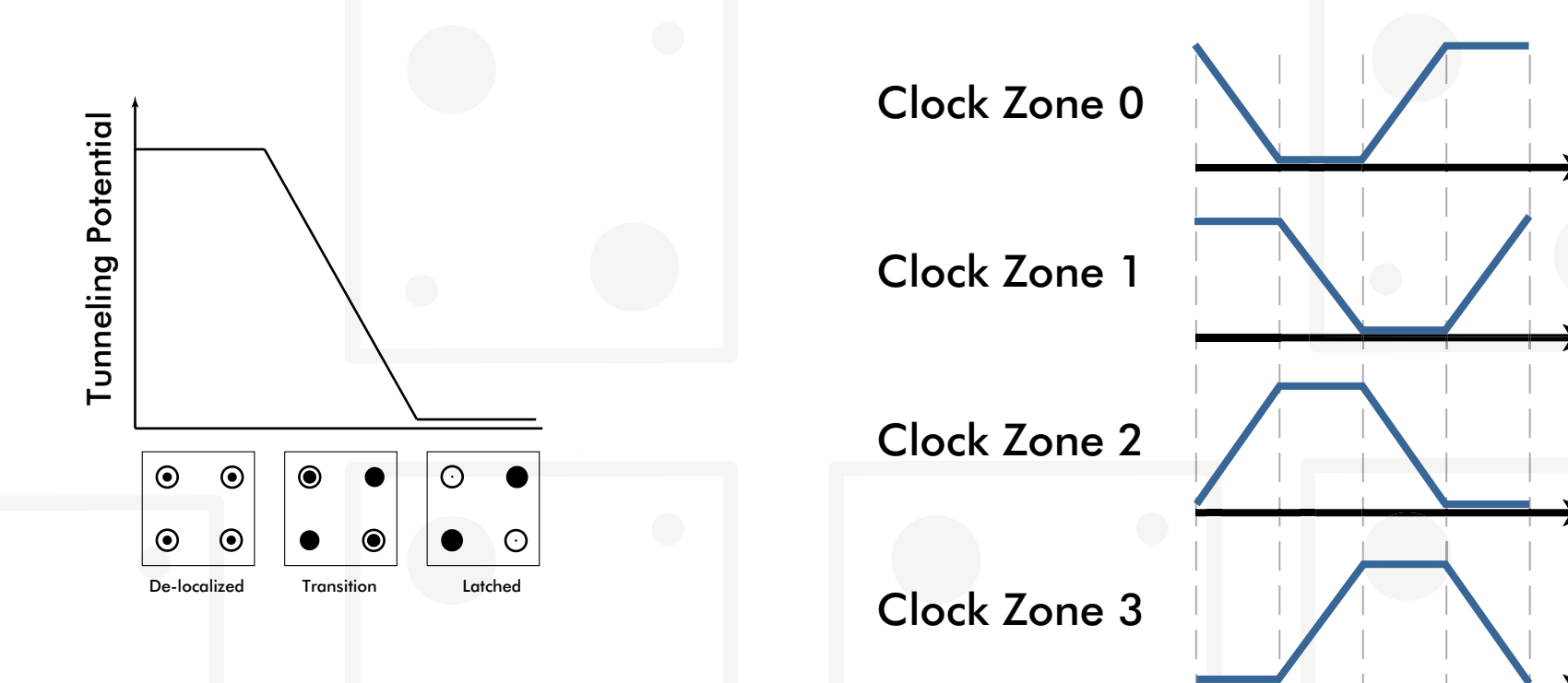


Figure 6 - QCA adiabatic switching and clocking zones

In order to overcome problems associated with thermal excitation, four clocking zones are employed, each 90° out of phase. This allows for cells to latch successively, in a controlled manner.

QCADesigner

Creating large, complicated digital designs using QCA would be nearly impossible without a design and layout tool. **QCADesigner** is an ongoing project to create such a tool. Major features already implemented in **QCADesigner** include:

- advanced cell placement
- block import and export
- sophisticated simulation model
- efficient formal design verification

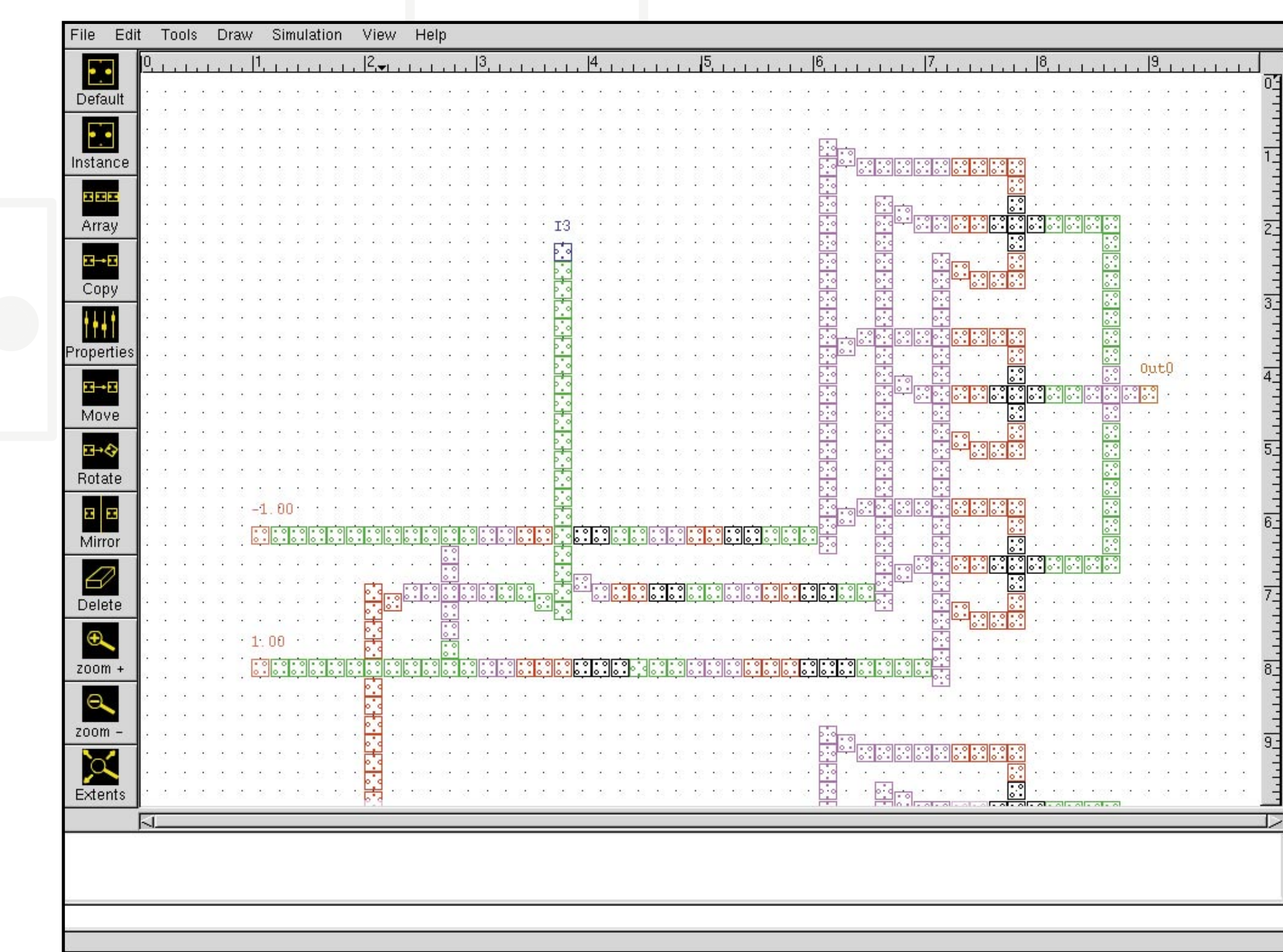


Figure 7 - 4-bit CLA Adder designed in QCADesigner

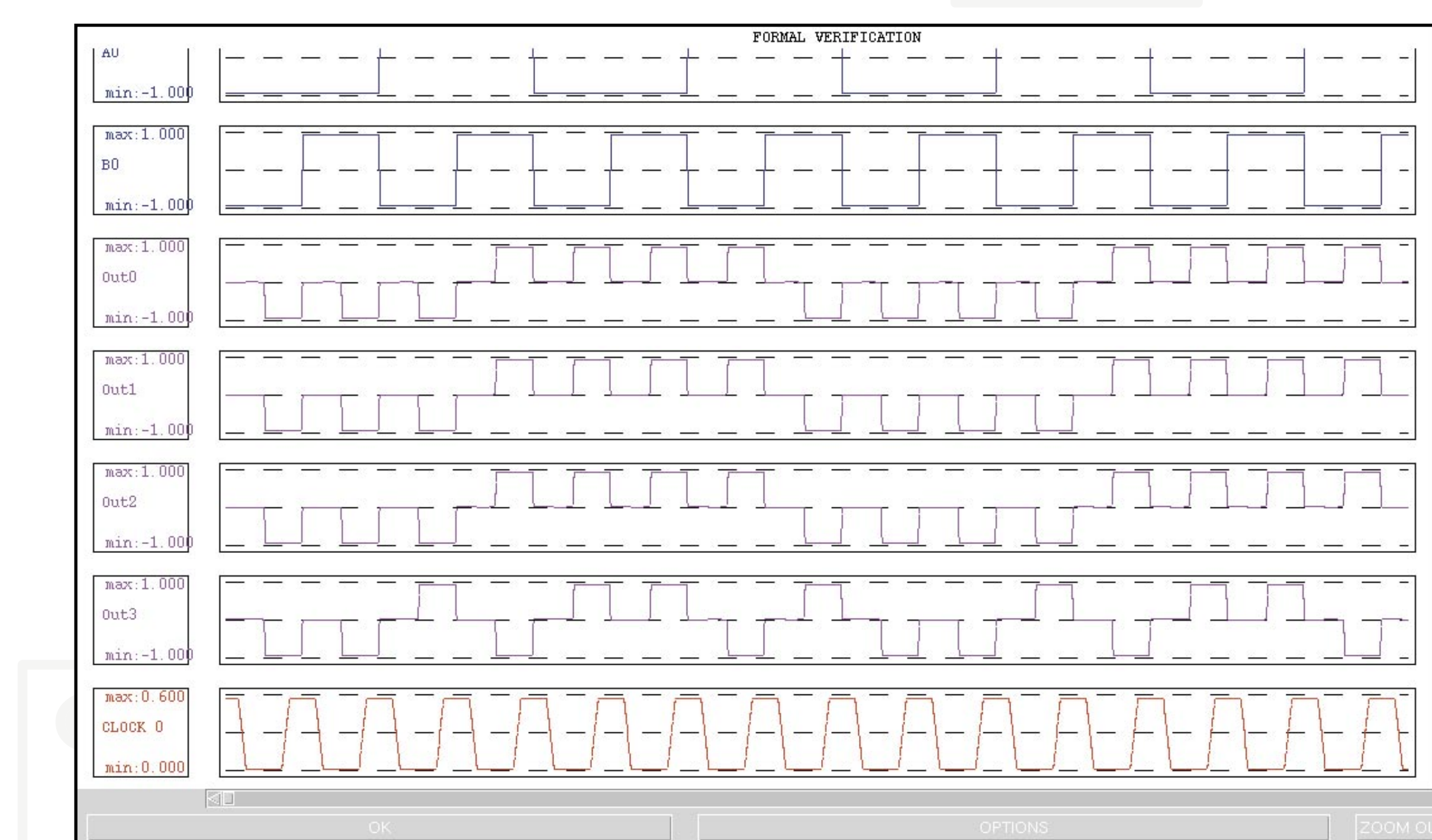


Figure 8 - Formal verification results window

QCADesigner is available for free. Source code and binaries for Solaris, Linux and Mac OS X are available at:

<http://people.atips.ca/~walus/>