Abstract
Quantum cellular automaton (QCA) is a newly-developed technology for next-generation nanoelectronics. It results in high device density, ultra-fast speed and almost zero power dissipation. Multiplier is an important unit in microprocessor design. Traditional N-bit array multiplier consists of N*N adder units and results in large circuit area. In this poster, a novel area-efficient self-cycling architecture for N-bit multiplier is proposed. It ingeniously utilizes the inherent pipeline feature of QCA clock zones to convert the array multiplication into a serial-in-serial-out process. In this way, the N-bit multiplier can be implemented with a single (instead of N*N) adder unit, which leads to significant area saving. The proposed self-cycling architecture can be extended to other N-bit QCA circuits as well. It is especially suitable for area-critical QCA circuit design where serial output can be accepted.

Keywords: Quantum-dot Cellular Automata (QCA), Multiplier, Self-cycling Architecture, QCA designer.

Introduction
Quantum Cellular Automata (QCA) is a newly-developed technology for next-generation nanoelectronics. It utilizes charge polarity instead of voltage to represent digital “0” and “1” states. The working principle is based on the electron distribution among quantum dots inside the QCA cells. In this poster, the design and simulation of a self-cycling multiplier structure are reported.

Self-cycling QCA Multiplier
1). Carry-Self-cycling multiplier design
As we observe the structure of N×N multiplier, it contains many identical multiplication-addition units. Each adder unit takes the carryout from previous unit. In this way, we may combine multiple units into a single unit, but use pipeline structure to store the outputs in series. As every unit will pass the carryout to next unit, why not let them become input again? In this way we can use one unit to finish what four slice-bit units do (in 4×4 multiplier), as shown in Figure 1. This theory is based on different clock zones can save different signal. Such as when combining four units into one, the first output S3b0 (SUM of a0×b0) will be saved in first clock zone, then S2b0 then S1b0 then S0b0 finally Cas3b0 (Carry a0×b0). Every output comes as the clock zone sequence.

By doing this a 4×4 bit QCA pipelined multiplier can become four signal units in series. As shown in Figure 2, 16 bit-slice units have been combined to 4 units. The output sequence is presented as clock zone number increasing.

2). Combined 4x4 pipelined multiplier into 4 single units in series
In Figure 2, we can see the Carry-self-cycling unit design. It is very similar to slice-bit unit but simpler. The Carry-self-cycling part is in the center. Because self-cycling part will have one clock zone delay passing to carryout (reaching the final majority gate), other wire (Cin3) should have one clock zone delay to make sure the inputs of all final majority gates arrive at the same time.

3). Self-cycling multiplier design
Since Carry out can become self-cycling, why not Sum out? As shown in Figure 4, if carry-self-cycling Sum output is used as its input, what will happen? 4 units will finally be combined into one; it will save significant area and make the design much more compact. The output sequences are showing as the output lane in Figure 4.

4). Simulation
For easier adjust the clock zone during design this self-cycling, I took 1111×1111 as an example test pattern because it results in many carries, which is good to show how the clock zones should be adjusted.

Self-cycling QCA Multiplier
1). Carry-Self-cycling multiplier design
As we observe the structure of N×N multiplier, it contains many identical multiplication-addition units. Each adder unit takes the carryout from previous unit. In this way, we may combine multiple units into a single unit, but use pipeline structure to store the outputs in series. As every unit will pass the carryout to next unit, why not let them become input again? In this way we can use one unit to finish what four slice-bit units do (in 4×4 multiplier), as shown in Figure 1. This theory is based on different clock zones can save different signal. Such as when combining four units into one, the first output S3b0 (SUM of a0×b0) will be saved in first clock zone, then S2b0 then S1b0 then S0b0 finally Cas3b0 (Carry a0×b0). Every output comes as the clock zone sequence.

By doing this a 4×4 bit QCA pipelined multiplier can become four signal units in series. As shown in Figure 2, 16 bit-slice units have been combined to 4 units. The output sequence is presented as clock zone number increasing.

2). Combined 4x4 pipelined multiplier into 4 single units in series
In Figure 2, we can see the Carry-self-cycling unit design. It is very similar to slice-bit unit but simpler. The Carry-self-cycling part is in the center. Because self-cycling part will have one clock zone delay passing to carryout (reaching the final majority gate), other wire (Cin3) should have one clock zone delay to make sure the inputs of all final majority gates arrive at the same time.

3). Self-cycling multiplier design
Since Carry out can become self-cycling, why not Sum out? As shown in Figure 4, if carry-self-cycling Sum output is used as its input, what will happen? 4 units will finally be combined into one; it will save significant area and make the design much more compact. The output sequences are showing as the output lane in Figure 4.

4). Simulation
For easier adjust the clock zone during design this self-cycling, I took 1111×1111 as an example test pattern because it results in many carries, which is good to show how the clock zones should be adjusted.

6). Comparison between pipelined and self-cycling multiplier
Table 1. comparison between pipelined and self-cycling multiplier

When compared to pipelined QCA multiplier, it is a trade-off, self-cycling multipliers are using more delay to trade for area.

References